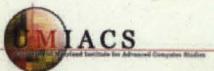


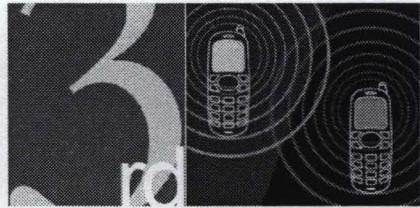


THIRD INTERNATIONAL CONFERENCE ON

MOBILE AND UBIQUITOUS MULTIMEDIA (MUM2004)

College Park, Maryland, U.S.A. October 27 - 29, 2004





THIRD INTERNATIONAL CONFERENCE ON
MOBILE AND UBIQUITOUS MULTIMEDIA (MUM2004)
College Park, Maryland, U.S.A. October 27 - 29, 2004

MUM 2004

**Proceedings of the 3rd International Conference on
Mobile and Ubiquitous Multimedia,
October 27-29, 2004
College Park, Maryland, USA**

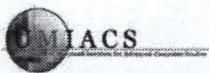
Organized by
**University of Maryland, Institute for Advanced
Computer Studies (UMIACS)**

In cooperation with

**ACM,
SIGCHI,
SIGGRAPH,
SIGMOBILE**

Edited by

David Doermann and Ramani Duraiswami



FUJITSU
FOR CONNECTIVITY AND INNOVATION

NOKIA
CONNECTING PEOPLE

Microsoft

PROCEEDINGS

of the
International Conference on
Advanced Computer Studies
ACCS '93, held at
AUB, Beirut, Lebanon

organized by
the Institute for Advanced Computer Studies
(IACS) of the University of Maryland

and published by



University of Maryland

A limited number of additional copies of these proceedings are available for \$50 from:

University of Maryland
Institute for Advanced Computer Studies
College Park, MD 20742
Phone: (301) 405-6444
Fax: (301) 314-2644
Email: mum04@umiacs.umd.edu

Table of Contents

bYOB [Build Your Own Bag], A computationally-enhanced modular textile system, <i>G. Nanda, A. Cable, V.M. Bove, Jr.</i>	1
Alternate Feature Location for Rapid Navigation Using a 3D Map on a Mobile Device, <i>M. Bessa, A. Coelho and A. Chalmers</i>	5
Exploring the Potentials of Combining Photo Annotating Tasks with Instant Messaging Fun, <i>Y. Qian and L.M.G. Feijis</i>	11
Pocket PhotoMesa: A Zoomable Image Browser for PDAs, <i>A. Khella, and B.B. Bederson</i>	19
Enabling Fast and Effortless Customisation in Accelerometer Based Gesture Interaction, <i>J. Mäntyjärvi, J. Kela, P. Korppiä, and S. Kallio</i>	25
The Road Rager – Making Use of Traffic Encounters in a Mobile Multiplayer Game, <i>L. Brunnberg</i>	33
UMAR – Ubiquitous Mobile Augmented Reality, <i>A. Henrysson and M. Ollila</i>	41
The GapiDraw Platform: High-Performance Cross-Platform Graphics on Mobile Devices, <i>J. Sanneblad and L.E. Holmquist</i>	47
Middleware Design Issues for Ubiquitous Computing, <i>T. Nakajima, K. Fujinami, E. Tokunaga, H. Ishikawa</i>	55
Plug-and-Play Application Platform: Towards Mobile Peer-to-Peer, <i>E. Harjula, M. Ylianttila, J. Ala-Krikka, J. Riekki, and J. Sauvola</i>	63
Survey of Requirements and Solutions for Ubiquitous Software, <i>E. Niemelä and J. Latvakoski</i>	71
Utilizing Context-Awareness in Office-Type Working Life, <i>M. Tähti, V-M. Rautio and L. Arhippainen</i>	79
Towards Connectivity Management Adaptability: Context Awareness in Policy Representation and End-to-end Evaluation Algorithm, <i>J-Z. Sun, J. Riekki, J. Sauvola, and M. Jurmu</i>	85
Usage Patterns of FriendZone—Mobile Location-Based Community Services, <i>A. Burak and T. Sharon</i>	93

Fast Watermark Detection Scheme for Camera-equipped Cellular Phone, <i>T. Nakamura, A. Katayama, M. Yamamuro, N. Sonehara</i>	101
New High-speed Frame Detection Method: Side Trace Algorithm (STA) for i-appli on Cellular Phones to Detect Watermarks, <i>A. Katayama, T. Nakamura, M. Yamamuro, N. Sonehara</i>	109
Automatic Video Production of Lectures Using an Intelligent and Aware Environment, <i>M. Bianchi</i>	117
A Novel Video Coding Scheme for Mobile Devices, <i>Y. Wang, H. Li and C.W. Chen</i>	125
Utilising Context Ontology in Mobile Device Application Personalisation, <i>P. Koripää, J. Häkkilä, J. Kela, S. Ronkainen, and I. Känsälä</i>	133
Design and Evaluation of Producer: A Mobile Authoring Tool for Personal Experience Computing, <i>C.-M. Teng, C.-I. Wu, Y.-C. Chen, H.-H. Chu, and J.Y.-J. Hsu</i>	141
Mobile Kärpät – A Case Study in Wireless Personal Area Networking, <i>T. Ojala, J. Korhonen, T. Sutinen, P. Parhi and L. Aalto</i>	149
Configuring Gestures as Expressive Interactions to Navigate Multimedia Recordings from Visits on Multiple Projections, <i>G. Jacucci and J. Kela, and J. Plomp</i>	157
IP Network for Emergency Service, <i>K.K.A. Zahid, L. Jun, K. Kazaura, and M. Matsumoto</i>	165
Mobile Multimedia Service's Development – Value Chain Perspective, <i>J. Karvonen and J. Warsta</i>	171
User Experiences on Combining Location Sensitive Mobile Phone Applications and Multimedia Messaging, <i>J. Häkkilä, and J. Mäntyjärvi</i>	179
Digital Rights Management & Protecting the Digital Media Value Chain, <i>M.L. Smith</i>	187
Bandwidth Optimization by Reliable-Path Determination in Mobile Ad-Hoc Networks, <i>S. Raghavan, S. Akkiraju and S. Sridhar</i>	193
atMOS: Self Expression Movie Generating System for 3G Mobile Communication, <i>S. Tokuhisa, T. Kotabe and M. Inakage</i>	199

The Challenges of Wireless and Mobile Technologies – The RFID Encourages the Mobile Phone Development, <i>T. Hori and M. Matsumoto</i>	207
Context-Aware Middleware for Mobile Multimedia Applications, <i>O. Davidyuk, J. Riekki, V.-M. Rautio and J. Sun</i>	213
Middleware Support for Implementing Context-Aware Multimodal User Interfaces, <i>P. Repo and J. Riekki</i>	221
Efficient Method for Multiple Compressed Audio Streams Spatialization, <i>A. B. Touimi, M. Emerit and J.-M. Pernaux</i>	229
Digital Photo Similarity Analysis in Frequency Domain and Photo Album Compression, <i>Y. Lu, T.-T. Wong and P.-A. Heng</i>	237
Multiple Embedding Using Robust Watermarks for Wireless Medical Images, <i>D. Osborne, D. Abbott, M. Sorell and D. Rogers</i>	245
A Mediation Framework for Multimedia Delivery, <i>R.K. Ege, L. Yang, Q. Kharma, O. Ezenwoye</i>	251
Task Computing, <i>Z. Song, R. Masouka, Y. Labrou</i>	257
Author Index	259

Preface

Welcome to MUM 2004 in College Park, Maryland, USA

The goal of the MUM2004 conference is to provide an international forum for presenting recent research results on mobile and ubiquitous multimedia, and to bring together experts from both academia and industry for an exchange of ideas and discussion on future challenges. The program includes keynote presentations and tutorials by leading experts and contributed papers describing recent progress in mobile and ubiquitous multimedia. The papers presented in the conference are published in this conference proceeding, which will be offered to international publishers.

This year, a one day workshop and tutorial theme will center Personal and Institutional Security. We have guest speakers from government, industry and academia, focused on how mobile devices are being used to provide monitoring, security services, and emergency and disaster coordination. The theme follows through into the 2 day conference which will expand to other uses of mobile multimedia in ubiquitous environments.

Best Paper and Best Student Paper awards will be presented for outstanding contributions.

Conference topics

Submissions have been solicited on, but were not limited to, the following topics on mobile and ubiquitous multimedia:

- Architectures, protocols, and algorithms to cope with mobility, roaming, limited bandwidth, or intermittent connectivity
- Case studies, field trials and evaluations of new applications and services
- HCI, interaction design and techniques, user-centered studies
- Intelligent, aware, proactive, and attentive environments, perception and modeling of the environment
- Middleware and distributed computing support for mobile and ubiquitous multimedia
- Mobile computer graphics, games and entertainment
- Novel adaptive/context-aware/mobile/ubiquitous/wireless multimedia applications and systems
- Streaming mobile multimedia
- Media analysis on mobile devices

We acknowledge and thank our sponsors ACM, University of Maryland Institute for Advanced Computer Studies, Nokia, Microsoft and Fujitsu, for making this international conference possible.

We thank the Keynote Speakers, Profs. Brad Myers, and V.S. Subrahmanian

The organization of this meeting was handled in part by the staff of the University of Maryland Institute for Advanced Computer Studies (UMIACS); in particular, we thank Yang Wang for his support of the WWW site, Christopher McCarthy for his graphic design expertise and Denise Best for all the behind the scenes work to make this event a success.

David Doermann
General Chair

Ramani Duraiswami
Program Chair

MUM 2004 Program Committee

MUM 2004 we thank our distinguished experts in the field of Mobile and Ubiquitous Multimedia for their reviews and comments.

- Ashok, Agrawala, University of Maryland, USA
- Agre, Jonathan, Fujitsu Laboratories of America
- Bederson, Ben, University of Maryland, USA
- Billinghurst, Mark, HIT Laboratory, New Zealand
- Chalmers, Alan, University of Bristol, UK
- Doermann, David, University of Maryland, USA
- Duraiswami, Ramani, University of Maryland, USA
- Ebert, David, Purdue University, USA
- Gross, Tom, Bauhaus-University, Weimar, Germany
- Haritaoglu, Ismail, IBM, USA
- Herman, Marty, NIST, USA
- Hull, Jon, Ricoh CRS, USA
- Mitsuji, Matsumoto, Waseda University, Japan
- Kalaiah, Aravind, Nokia Research Center
- Kuutti, Kari, University of Oulu, Finland
- Niemela, Eila, VTT Electronics, Finland
- Ojala, Timo, University of Oulu, Finland
- Ollila, Mark, Linkoping University, Sweden
- Pulli, Kari, Nokia Mobile Phones, Finland
- Peng, Lim Ee, Nanyang Technological University, Republic of Singapore
- Rantzer, Martin, Swedish Defense Research Ag
- Rauterberg, Matthias, Technical University Eindhoven, UK
- Silven, Olli, University of Oulu, Finland
- Veijalainen, Jari, University of Jyvaskyla, Finland

Additional Reviewers

- Elina Koivisto, Nokia Research Center
- Jouka Mattila, Nokia Research Center
- Lusheng Ji, Fujitsu Laboratories of America
- Zhexuan Song, Fujitsu Laboratories of America
- Lusheng Ji, Fujitsu Laboratories of America
- Jani Korhonen, University of Oulu
- Markus Aittola, University of Oulu

Organizers:

- Denise Best, University of Maryland, USA
- Chris McCarthy, University of Maryland, USA
- Yang Wang, University of Maryland, USA

KEYNOTE SPEAKER



V.S. SUBRAHMANIAN
UNIVERSITY OF MARYLAND

STORY: EXTRACTING STORIES FROM HETEROGENEOUS INFORMATION SOURCES AND DELIVERING THEM TO HETEROGENEOUS DEVICES

ABSTRACT

We consider the problem of accessing multiple heterogeneous information sources (e.g. web sources, relational DBMSs, flat files, etc.) and constructing a succinct story about an entity (e.g. a person or a place or an event) in a way that meets the user's interests, and deliver the resulting story to the user across heterogeneous, potentially mobile devices. As such, a story must be short (its length is determined by the user), and it must include facts whose value may vary from one user to another. We have developed a suite of algorithms to do this, and we are conducting experiments on the quality of the stories as well as the computation time involved. We have developed a prototype implementation of the STORY system to create and deliver stories about Pompeii where user's could have multiple hardware devices on which the story is delivered - currently, we can deliver stories in two languages to laptops and PDAs such as the PocketPC. (Joint work with M. Albanese, C. Cesarano, M. Fayzullin and A. Picariello).

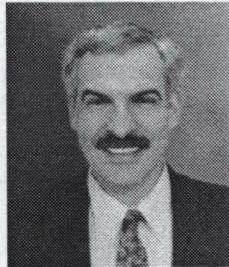
Biography: V.S. SUBRAHMANIAN is Professor of Computer Science and Interim Director of the University of Maryland Institute for Advanced Computer Studies. He received the NSF Young Investigator Award in 1993 and the Distinguished Young Scientist Award from the MD Academy of Science in 1997. He is known for his work on integrating heterogeneous information sources, for developing the foundations of multimedia database systems, for work on probabilistic reasoning, and for building cooperative agent systems on top of legacy databases and codebases. He has edited two books, one on nonmonotonic reasoning (MIT Press) and one on multimedia databases (Springer). He has co-authored an advanced database textbook (Morgan Kaufman, 1997), a multimedia database textbook, and a monograph on heterogeneous software agents.

Prof. Subrahmanian has given invited talks at numerous national and international conferences - in addition, he has served on numerous conference and funding panels, as well as on the program committees of numerous conferences. He has also chaired several conferences.

Prof. Subrahmanian is or has previously been on the editorial board of IEEE Transactions on Knowledge and Data Engineering, Artificial Intelligence Communications, Multimedia Tools and Applications, Journal of Logic Programming, Annals of Mathematics and Artificial Intelligence, Distributed and Parallel Database Journal, and Theory and Practice of Logic Programming.

Prof. Subrahmanian has served on DARPA's (Defense Advanced Research Projects Agency) Executive Advisory Council on Advanced Logistics and as an ad-hoc member of the Air Force Science Advisory Board (2001).

KEYNOTE SPEAKER



BRAD MYERS
CARNEGIE MELLON UNIVERSITY

MOBILE DEVICES FOR CONTROL OF UBIQUITOUS MULTIMEDIA

ABSTRACT

Computers today are ubiquitous, with every new automobile and home and office appliance likely to have one or more embedded computer's. Many people now carry a computer with them at all times, embedded in their mobile phones or personal digital assistant (PDA). Recent and developing wireless technologies, such as Wi-Fi, Bluetooth, and 3G, are enabling appliances and their embedded computers to be in close, interactive communication with the mobile devices that people are carrying. Many appliances have multimedia capabilities, such as stereos, VCRs, DVD players, speakers, televisions, regular computers, etc. If all these could communicate with the user's mobile device, what would be interaction be like? We are exploring, as part of the Pebbles research project, the many ways that mobile devices such as PDAs and cell phones can be used to augment the many "fixed" computers in the user's vicinity, rather than just serving as a replacement for them. This brings up many interesting research questions, such as how to provide a user interface that spans multiple devices which might be in use at the same time? How will users and the system decide which functions should be presented in what manner on what device? Can the user's mobile device be effectively used as a "Personal Universal Controller" to provide an easy-to-use and familiar interface to all of the complex multimedia appliances available to the user? This talk will provide our preliminary observations on these issues, and will include demonstrations of some of our systems that we are using to investigate them.

Biography: Brad A. Myers is a Professor in the Human-Computer Interaction Institute in the School of Computer Science at Carnegie Mellon University, where he is the principal investigator for various research projects including: the Pebbles Hand-Held Computer Project, Natural Programming, User Interface Software, and Demonstrational Interfaces. He is the author or editor over 250 publications, including the books "Creating User Interfaces by Demonstration" and "Languages for Developing User Interfaces," and he is on the editorial board of five journals. He has been a consultant on user interface design

and implementation to over 50 companies, and regularly teaches courses on user interface design and software. In 2004, he was elected to the CHI Academy, an honor bestowed on the principal leaders of the field, whose efforts have shaped the discipline and led the research in human-computer interaction. Myers received a PhD in computer science at the University of Toronto where he developed the Peridot UIMS. He received the MS and BSc degrees from the Massachusetts Institute of Technology during which time he was a research intern at Xerox PARC. From 1980 until 1983, he worked at PERQ Systems Corporation. His research interests include user interface development systems, user interfaces, hand-held computers, programming by example, programming languages for kids, visual programming environments. He belongs to SIGCHI, ACM, IEEE Computer Society, IEEE, EFF, and Computer Professionals for Social Responsibility.

Address: Human Computer Interaction Institute, Carnegie Mellon University, Pittsburgh, PA 15213-3891. bam@cs.cmu.edu, <http://www.cs.cmu.edu/~bam>.

bYOB [Build Your Own Bag]:

A computationally-enhanced modular textile system

Gauri Nanda

Media Lab

Massachusetts Institute of
Technology
Bldg E15-357
20 Ames St
Cambridge, MA 02139
nanda@media.mit.edu

Adrian Cable

Media Lab

Massachusetts Institute of
Technology
Bldg E15-357
20 Ames St
Cambridge, MA 02139
acable@mit.edu

V. Michael Bove Jr.

Media Lab

Massachusetts Institute of
Technology
Bldg E15-368B
20 Ames St
Cambridge, MA 02139
vmb@media.mit.edu

ABSTRACT

We present bYOB (Build Your Own Bag), a flexible, computationally enhanced modular textile system from which to construct smart fabric objects. bYOB was motivated by a desire to transform everyday surfaces into ambient displays for information and to make building with fabric as easy as playing with Lego blocks. In the realm of personal architecture, bYOB is an interactive material that encourages users to explore and experiment by creating new objects to seamlessly integrate into their lives. The physical configuration of the object mediates its computational behavior. Therefore, an object built out of the system of modular elements understands its geometry and responds appropriately without any end-user programming. Our current prototype is a bag built out of the system that understands it is a bag when the handle is attached to the mesh of modules, responds by illuminating its fabric and inner contents when the sun goes down (Fig 1), communicates the presence of objects placed in the bag, and interacts with the user via speech. We describe how bYOB contributes to and differs from existing work in modular based systems and fabric interfaces. We discuss our development process in respect to physical, electronic, and conceptual design. We also describe salient features and future applications enabled by this new construction kit.

Keywords

Ambient Interface, Tangible Interface, Customization, Enhanced Situational Awareness, Fabric Building Blocks, Modular Systems, Network Detection.

1. INTRODUCTION

bYOB's significance is that anyone can make an object at any time and anywhere whether they are a designer with a new vision, an office worker in need of additional light, or a student who needs useful information while in transit but does not want to carry several additional electronic devices. Therefore, the modules allow users to express themselves and create interactions with their environment. Rather than needing to carry several electronic devices to access these applications, bYOB exploits everyday surfaces as a natural medium by which to communicate information.

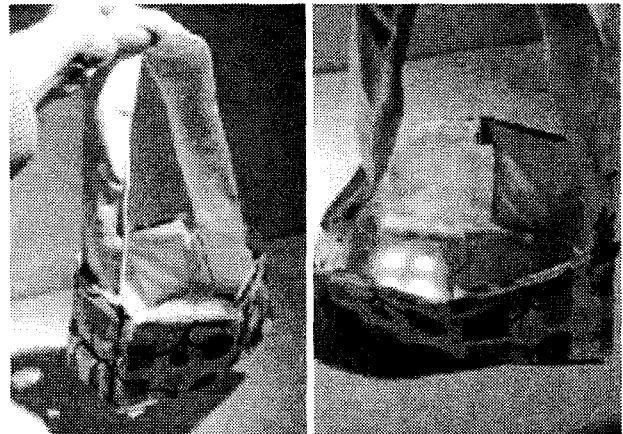


Figure 1: Current prototype demonstrating light sensing capability.

On a conceptual level, bYOB seeks to challenge the conventional ways in which technology, its producers, and the role of the intended 'beneficiary' or user, are viewed; the common notion that the technology dictates how we use it obscures its potential as an interactive outlet. By using a modular textile system, the user is not divorced from the creative process. The human computer interaction that is afforded is one that is aware of the changing needs and capabilities of people and environments, yet is dually cognizant of a wider creative evolution.

When designing applications for bYOB, we were inspired by the lifestyles of today's students and professionals. Our target audience can be characterized in part by their mobility — constantly moving between home, work, school and personal commitments. As a responsive tangible interface, bYOB can be customized to fit their unique needs. For example, a set of bYOB modules may be physically configured into a window shade or lighting fixture in the office and later be reconfigured into a bag or scarf to aid the user in transit.

1.1 Current Applications

Environmental Awareness. bYOB modules are equipped with sensors to respond to changes in, for example, light level and temperature. Our current prototype illuminates the contents of the bag when ambient illumination drops below a desirable level.

Object Detection. Using radio frequency identification, bYOB modules can detect whether or not important objects (e.g. cell phones, wallets) are nearby and alert the user, through the fabric's ambient light display or using speech, if the items are missing. For example, a bYOB-constructed bag will light up to inform the user if he or she tries to leave home without keys.

Topology Mapping. The modules are dynamically aware of their configuration and understand geometries other than those of bags. A wall hanging may be configured and act as a light source or a curtain may be constructed to respond appropriately to environmental changes (Fig 2).

1.2 Applications Under Development

Wireless Capability. Using the Bluetooth protocol, bYOB modules will be able to download information from the Internet (e.g. weather forecasts) to the modules. Additionally, the user will have the ability to 'log in' to one or more remote bYOB bags from a computer and search for the location of items.

Network Detection. Mobile lifestyles often lead to increased use of mobile phones, PDAs (personal digital assistants) and laptop computers. However, the wireless networks used with these items are not immediately visible. Through the ambient display, bYOB modules can make information regarding the existence and strength of networks readily available.

Seamless Integration Into Our Lives. We envision that when you hang your bag up for the day, it will automatically recharge its battery and synchronize itself with other objects in the bYOB network.

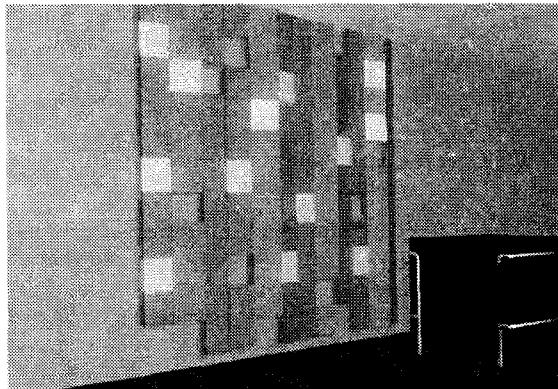


Figure 2: bYOB can detect environmental changes actuated through ambient light display.

2. RELATED WORK

There are several examples of modular based computational interfaces developed for various applications that have informed our research. One such system, Triangles [2], used for non-linear storytelling but available for other applications, most closely parallels bYOB in its ability to allow users to handle and manipulate digital information with construction pieces. While both Triangles and bYOB serve as displays for information, the modules of bYOB are used to transform familiar physical surfaces, and the technology is specifically designed for use as a flexible fabric interface.

Wifisense, a wearable scanner for wireless networks, is a self-contained handbag that conveys WiFi availability through a pattern of light and sound. [5] Unlike Wifisense, bYOB's design approach is of embedding systemic behavior inside modules of fabric to provide a novel approach for building objects. Additionally, bYOB creations can serve multiple purposes over a wide range of applications, not limited to the interface of a bag.

E-broidery or conductive fabric was examined as a possible prior work to build on. E-broidery is attractive in its ability to replace standard circuitry with flexible, washable substrates. After examination into the current state of conductive threads however, we discovered that the integrity of the threads is in question as they may be apt to break when subjected to stress. Additionally, we learned that they must endure a lengthy process of welding and sewing. An alternate design used in bYOB achieves similar freedom of movement without laborious manufacturing processes, an important requirement since many bYOB modules must be fabricated. Nonetheless, E-broidery remains a subject of study and further research is being done to investigate whether or not it is possible to replace some of the wiring with conductive fabric traces [4].

bYOB's contribution marries notions of ambient communication with tangible media to allow for the creation of personal architecture. It encourages rapid exploration in the manipulation of fabric and is a universal approach because it assumes no formal design training and little computational experience. The 'digital language' expressed by the object is made unambiguous because the user is in control of the physical form and light pattern display.

3. PHYSICAL DESIGN

The current bYOB prototype consists of squares and equilateral triangles approximately 4"x4" and no more than 1/8" thick. The simple geometries of squares and triangles were chosen because they are easily recognizable and can be effortlessly manipulated into two and three-dimensional shapes. Additionally, the dimensions were chosen to fit comfortably when grasped by the human hand.

The task of constructing a connection to fit securely and easily with neighboring modules as well as transmit data and power posed a significant challenge in the development process. Several approaches for the modules' physical connectors were examined including snap-hinges, screws, zippers, and magnetic/metal snaps. At first, we choose commercial metal snap connectors acquired from a fabric store because of their unambiguous design and their performance as conductors. The snaps were disappointing because their aggregate weight compromised the feel of the fabric. Our current prototype employs conductive hook-and-loop material that is lightweight and has impressively low resistance, allowing data to be exchanged between modules. The simplicity of snapping pieces together with hook-and-loop allows the user to, at any time, add or subtract modules from an object to fulfill a situational change or geometric design. Additional time was spent ensuring that the male/female snaps on each module were part of an intuitive design where the user would not have to understand the technical complexities of lining up power, ground, and data pins (Fig 3).

Lightweight foam padding inside each module provides cushioning against the electronics. We are currently experimenting with different types of synthetic and natural fabrics for the outside of bYOB module shells. The material must be weather proof and have the ability to dissipate light from the LEDs. However, the fabric we are looking at is merely a suggestion of what the object might look like. Using the innovative Dual Lock material by 3M stitched to the outside of the modules as well as to the fabric, users have the ability to constantly change the fabric coverings of their objects.

4. TECHNICAL DESIGN

The system is comprised of three different types of modules: passives, actives, and parents. Active and parent modules contain an array of sensors, light actuating elements and a microprocessor. Passive modules, while physically identical on the exterior to the active modules, have the sole purpose of contributing structural support and physical form and therefore contain no light actuation or computational capabilities.

The parent module differs from the actives in that it controls the algorithm to equip the rest of the modules with appropriate applications. The shape of the parent module is dependent on the object created. For example, if the user builds a bag, the handle is designated the parent and alerts its children that they are part of a bag and should respond correspondingly (with object and network detection applications).

To ensure that bYOB is not cost-prohibitive as an end-user technology, emphasis was placed on using inexpensive electronics and on power conservation. The current design of each active module houses a 1-inch square printed circuit board covered in epoxy resin to level height and imbued with modest processing to locally manage communication, actuator response, and its own unique identity. The PIC16F876 microcontroller was chosen because of its low current consumption and because it works well with the I2C bus, the network by which a mesh of modules communicate. I2C was chosen because of its minimal cost and its plug and play ability. [3] For actuation, each active module uses RGB LEDs (light emitting diodes) to illuminate the fabric that covers the board. With several illuminated active modules, an ambient light pattern emerges.

For greatest efficiency and to conserve space, it was decided that data and power would be distributed through the network of modules. The conductive hook-and-loop material that acts as a mechanical connection between pieces also transmits power and data through the object. Four connections on each side provide power, ground, and the data and clock connections required by the I2C bus. As discovered in the research of Computational Building Blocks, a building kit for geometric modeling, using self-powered modules would greatly increase the cost and maintenance of each piece within the construction kit. [1] Instead, power is currently drawn from a centralized rechargeable battery piece. During development of the modules, emphasis was made on minimizing power consumption: the microprocessor in each module draws less than 1 mA (milli-ampere), and high-brightness low-current LEDs were chosen to maximize light output while keeping current draw low.

The current bYOB module implementation accommodates transducers and sensors for light and temperature and actuators for speech allowing interaction with the user in a natural way. Modules can be outfitted with additional sensors to respond to pressure and orientation.

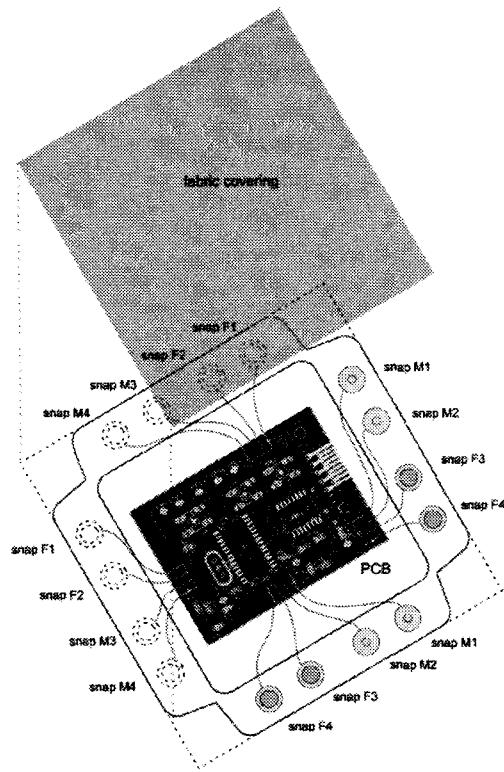


Figure 3: Module with PCB and connector layout

Given the negligible size of the circuit board, the use of fabric-based connectors and flexible cable, and the cushioning provided by the foam-fabric combination, modules effectively mask their technical belly, preventing them from coming into contact with the user's body. In subsequent designs we expect to see a decrease in the size and weight of each module's inner contents thus increasing usability.

Additionally, by using waterproof fabric and electronics covered in epoxy we ensure that the modules are durable and washable.

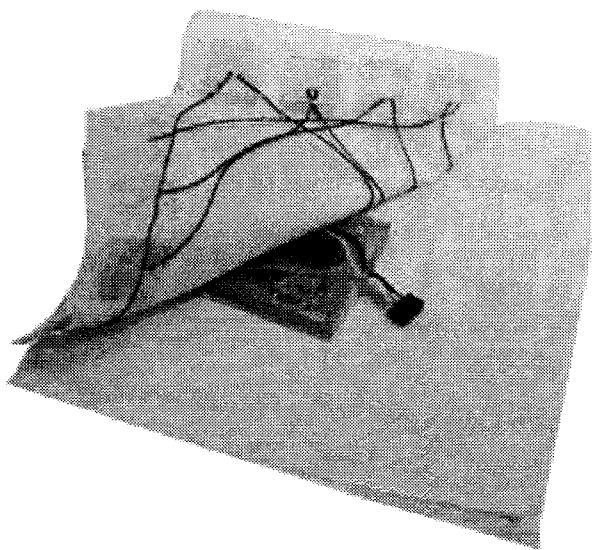


Figure 5: BYOB encourages user design

5. EVALUATION

An important part of our development process is feedback from potential users of bYOB. We conducted a survey with local college students to find out what information and objects constructed out of bYOB would be most useful and what the strengths and weaknesses of the project were.

Our respondents showed enthusiasm for this application, particularly its ability to light up at night and alert users of missing items. There was concern about the durability of the system. If they were to invest in bYOB as a product, they would want to ensure that it would last a long time. Taking this feedback into account, we are developing physical connections that have negligible wear over time and looking into materials that are weather-resistant.

Furthermore, while almost all of our respondents considered themselves creative and said they would enjoy building objects with bYOB, they voiced concern about the time commitment bYOB might require. In response, we are refining the design to ensure that physical connections are minimal and work effortlessly. As the project progresses, we will continue to gather user feedback and observational data from a wider demographic.

6. FUTURE WORK AND CONCLUSIONS

Through further development, we hope to better understand bYOB's challenges and limitations. It is likely that we will confront the challenge of making the modules scalable to accommodate shapes and sizes different from triangles and squares of uniform size. Because it is a textile that can be used for interiors, bags and accessories, it is crucial that the system is aesthetically appealing to users, allows freedom in design, and durable enough to withstand frequent use.

The user-interface is constantly being refined so that bYOB couples seamlessly with the kinds of interaction we are already

accustomed to. As part of a distributed sensor network, we envision several bYOB objects that synchronize with each other and facilitate visual data exchange. A bYOB curtain would respond to changes in weather like the onset of rain and dually inform a bYOB bag that it should contain an umbrella.

7. ACKNOWLEDGMENTS

We thank Professor Hiroshi Ishii and Amanda Parkes for their support. This project has been funded by the Things That Think and Digital Life research consortia at the Media Laboratory.

8. ADDITIONAL AUTHORS

Additional authors: Moneta Ho (MIT Comparative Media Studies, email: monetaho@mit.edu) and Han Hoang (MIT Design and Computation, email: hoang@mit.edu).

9. REFERENCES

- [1] Anderson, D. Frankel, J., Marks, J., Agarwala, A., Beardsley, P., Hodgins, J., Leigh, D., Ryall, K., Sullivan, E. and Yedidia, J., Tangible Interaction + Graphical Manipulation: A New Approach to 3D Modeling. UIST 1999.
- [2] Gorbet, M., Orth, M. and Ishii, H., Triangles: Tangible Interface for Manipulation and Exploration of Digital Information Topography, in Proceedings of Conference on Human Factors in Computing Systems, CHI '98, Los Angeles, April 1998.
- [3] Irazabal, J., Blozis, S., I2C Bus Manual (Philips Semiconductors, March 24 2003)
- [4] Post, E.R., Orth, M., Russo, P.R., Gershenfield, N., E:broidery: Design and fabrication of textile-based computing, IBM Systems Journal, VOL 39. NOS 3&4, 2000.
- [5] Wifisense project, <http://www.wifisense.com>, 2003.

Alternate Feature Location for Rapid Navigation using a 3D Map on a Mobile Device

Maximino Bessa
University of Trás-os-Montes e
Alto Douro
Quinta de Prados
5000-911 Vila Real, Portugal
maxbessa@utad.pt

António Coelho
University of Trás-os-Montes e
Alto Douro
Quinta de Prados
5000-911 Vila Real, Portugal
acoelho@utad.pt

Alan Chalmers
University of Bristol
Computer Science
Bristol, England
alan@cs.bris.ac.uk

ABSTRACT

Finding one's way around an unfamiliar city can be a major challenge. While maps can provide a very good abstract representation of our world, and a simple and efficient way to navigate within that world, they are of little use when, for example, the absence of road signs prevents us from locating where we are on the map. Mobile devices offer the potential for providing relevant 3D information to enable us to locate ourselves, rapidly navigate around an unfamiliar environment and explore it interactively. However, mobile devices are constrained by resources such as bandwidth, storage and small displays. In this paper we investigate which is the most important visual information for position location within an unfamiliar urban environment and show how we can use this knowledge to provide a perceptually high quality 3D virtual environment on existing mobile devices.

Keywords

Mobile devices, 3D maps, visual perception, inattentional blindness

1. INTRODUCTION

Two dimensional maps are an age old method for helping us navigate through unfamiliar environments. With modern GPS assistance it is even possible to (approximately) locate our position on the map. However, a map is an abstract representation of the environment and as such, it simply may not be possible to orientate the map correctly in the absence of key information, such as road names. The next generation of maps should provide a more realistic representation of our world, a high quality 3D representation, and they should be where we need them most, on our Smart phones, PDAs and Laptops.

There are many issues that still need to be addressed if such high quality 3D maps are to be made readily available.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MUM 2004, October 27-29, 2004 College Park, Maryland, USA.
Copyright 200X ACM X-XXXXXX-XX-X/XX/XX ...\$5.00.

able at interactive rates on such mobile devices. Although the next generation of mobile devices may partially resolve some of the resource issues that exist today, such as bandwidth, storage, and small displays, this is likely to further increase demand for even more realistic and complex 3D applications. Visual perception is one approach that can be used to overcome this problem. By knowing exactly what users are looking at on the mobile device it may be possible to render only this part of the image at high quality while the rest of the scene could be rendered at a lower quality for a fraction of the computation cost, without the user being aware of this quality difference.

How we perceive an environment depends on who we are and the task that we are currently performing in that environment [12]. Visual attention is the process by which we humans select a portion of the available visual information for localization, identification and understanding of objects in the environment. It allows our visual system to process visual input preferentially by shifting attention about an image, giving more attention to salient locations and less attention to unimportant regions. Although our eyes are good, they are not perfect, and so when attention is not focused onto items in a scene they can literally go unnoticed. This is known as inattentional blindness [8].

This paper presents a series of experiments which help identify key features of a scene that users will use to orientate themselves in that environment. Knowledge of these key, salient features will enable them in future to be provided to a user at high quality while the remainder of the scene can be rendered in a much lower quality, saving significant bandwidth and computing power, without the user being perceptually aware of this difference in quality within the image.

2. RELATED WORK

In recent years, knowledge of the human visual system has been increasingly used to improve the quality of the displayed image, for example [3][4] [9][10] [11]. Other research has investigated how complex detail in the models can be reduced without any reduction in the viewer's perception of the models, for example Luebke and Hallen [6]. While for visual navigation systems, Maciel and Shirley used texture mapped primitives to represent clusters of objects to maintain high and approximately constant frame rates [7]. In addition, saliency models have been developed to simu-

late where people focus their involuntary attention in images [5] and these have been used to reduce overall computation costs [13].

Cater et al. [1] [2] showed that when a user was performing a task in an environment, it was possible to selectively render the task related objects at the highest quality and the remainder of the scene at a significantly lower quality without the user being aware of this difference in quality. Such selectively rendering was at a much lower computational cost than rendering the full scene at the highest quality, and yet still provided the user with a perceptually high quality experience. We shall use such an approach to provide perceptually high quality maps on mobile devices, but first we need to determine what are the key features that should be rendered at the highest quality.

3. EXPERIMENT

In a mobile device, due to limited processor power and bandwidth, we can't render realistic images in real time to produce an high quality 3D interactive map. Nevertheless we can render the more important objects of a scene in higher quality with selective rendering. To gather information about the key features that would be important in a scene, a case study was carried out to determine which elements of an urban scenario are more important to people when they were trying to orientate themselves. The task we chose was for the subjects to identify the correct spot from where a specific photograph had been taken. This case-study had two phases. In each phase 8 subjects were tested by being given 3 photographs in order to identify 3 different locations, figures 1-3. The photographs had a VGA resolution and a dimension of 10cm x 7 cm, and were all from the city center of Vila Real, Portugal. Note that the subjects used in first phase were different from the second phase. In the first phase the subjects were given the original photograph and asked to identify the exact spot and direction from where the photography was taken. While completing the task, each subject was asked to mark the objects in the urban scenario that helped him/her to locate the spot in order of priority.

The subjects were driven to the locations where the photographs had been taken from a different directions to that of the photographs. The time it took for the subject to identify the spot was measured and the subject was asked to specify the number of objects that had been used to identify the place where the photograph was taken. In the second phase the same procedure was followed with one difference; the photograph that was given to the subject had been altered. Some of the key features identified in the first phase were changed or removed in the photograph with Photoshop, or, in the case of the red and blue flag in figure 3, the actual item had been removed (in this case by the shop) when phase 2 of the experiment was conducted.

Again a group of 8 (different) subjects were given the three modified photographs and asked to identify the exact location from where the photograph had been taken.

The details of the task were well explained to the subjects prior to them undertaking the experiment. They were also shown an example picture to ensure that they clearly understood the instructions.



Figure 1: Location 1 with key features identified by one subject.



Figure 2: Location two with key features identified by one subject.



Figure 3: Location three with key features identified by one subject.



Figure 4: Modified photograph of location 1 with new key features identified.



Figure 5: Modified photograph of location 2 with new key features identified.



Figure 6: Modified photograph of location 3 with new key features identified.

4. RESULTS

The time spent, and the number of objects used to identify the correct spot by each subject in the two phases are shown in the next two tables.

Table 1: Results of the first phase, the average time spent in seconds, the standard deviation, and the average number of items that subjects used to identify each photograph

First phase	Time Average (in seconds)	Standard Deviation	Average No. of Items used
1st photo	70.75	26.18	2.88
2nd photo	70.63	25.12	3.13
3rd photo	62.13	10.83	3.13

Table 2: Results of the second phase, the average time spent in seconds, the standard deviation, and the average of items that subjects used to identify each photograph

Second phase	Time Average (in seconds)	Standard Deviation	Average No. of Items used
1st photo	70.13	24.39	2.75
2nd photo	60.75	22.39	3.25
3rd photo	84.25	18.57	3.75

As we can see, the average time spent trying to reach the right spot is very similar in phases one and two. This result was not expected as we thought that by removing the key features identified in the first phase, the subjects would require significantly more time to find the right spot. However, with the key features missing, the subjects simply used other features within the environment to help with their orientation. To be notice, however, is that in the first phase, all the subjects reached the right spot with an error of about a 1 meter, while in the second phase, this error increased to approximately 3 meters.

The elements in the photographs were classified in one of the following categories:

- Urban furniture - Lamps, seats, bins,etc.
- Buildings - All the buildings characteristics, such as doors, windows, balconies, geometry, volume and even the whole building.
- Publicity - all types of advertisements.
- Other - Cars, trees, temporary elements.

The elements used by the subjects to find the right spot are shown the figures 7 and 8.

As the results show, when most of the key features identified in first phase were changed or removed, the subjects identified other key features. For example in the second photograph, Figure 2, all the subjects identified a total of 9 elements of urban furniture, Figure 7. When the urban furniture was removed, Figure 5, the number of urban furniture items identified drops to 1 and at the same time the number of building elements increases from 12 to 20. We can also see that the total number of elements identified to find

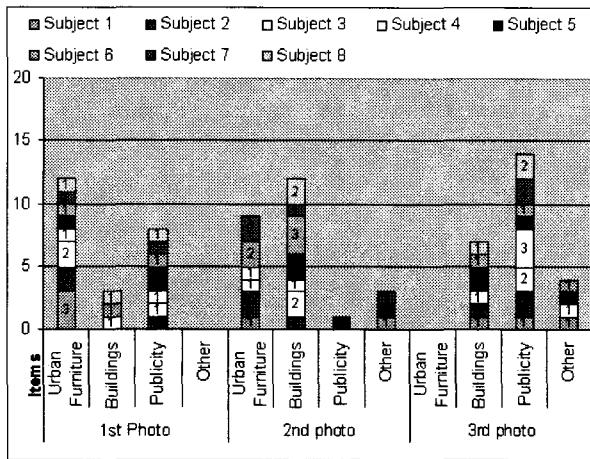


Figure 7: Items identified by the subjects in the first phase of the experiment.

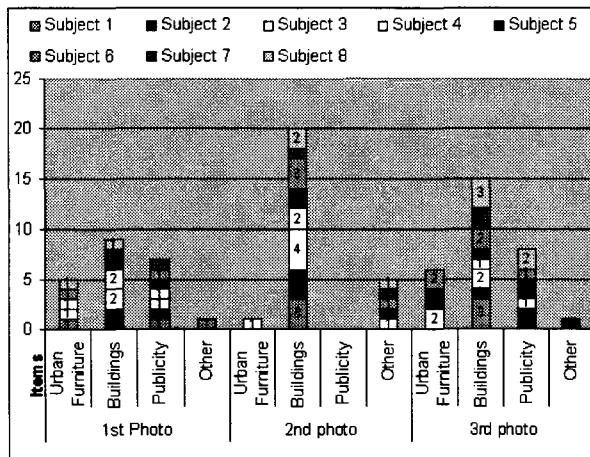


Figure 8: Items identified by the subjects in the second phase of the experiment

the right spot is the about the same (23 elements in original photograph and 22 in the changed photograph). This was similar for the other two photographs.

Publicity is an important key feature as subjects always noticed it when it was present. We also noticed that the geometry of the features, for example the size of the buildings, was always used by the subjects to gain an approximate orientation, and subsequently, when the subjects wanted to have a more precise idea of the spot where the photograph was taken, the key features, in particular urban furniture and publicity, were used. The minimum number of elements used to identify any spot was two and the maximum eight.

Another useful result from this study is that the subjects always chose the closest elements to identify the correct spot. Only if they didn't recognize any key feature near them did they look further a field to try to identify another key element.

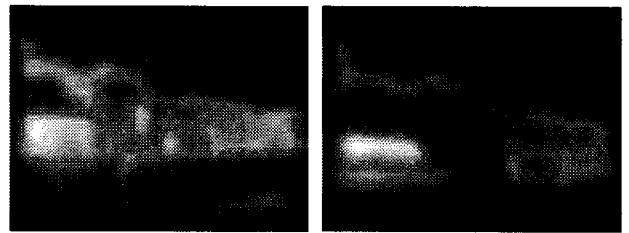


Figure 9: Saliency maps for photograph 1 (left) unmodified, (right) modified.

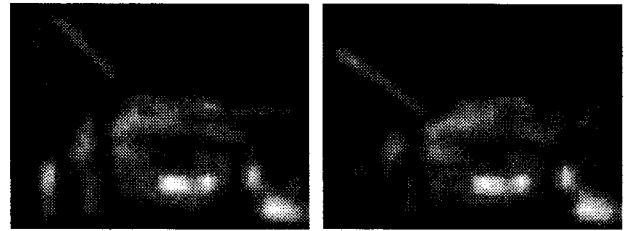


Figure 10: Saliency maps for photograph 2 (left) unmodified, (right) modified.

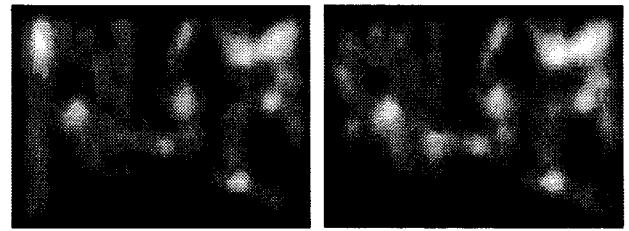


Figure 11: Saliency maps for photograph 3 (left) unmodified, (right) modified.

Figure 9 show the saliency maps for photograph 1 before and after it has been modified using the Itti and Koch's model[5]. Highly salient objects are in white and less salient

ones in increasing dark grey. In the unmodified photograph, the fountains were a key feature used in the orientation and yet they do not appear as a high saliency object in the saliency map, figure 9 (left). Similarly in figures 10 and 11, although some of the features identified by the users do appear as salient in the Itti and Koch model, a significant number don't.

5. CONCLUSIONS

The results of this pilot study have shown that a user does not need all the features of an environment to obtain a good orientation. In fact, only the overall geometry and a few key features are required, and these features are not necessarily those most salient to the human visual system. Other features in the image are all but ignored. Furthermore, the time taken to orientate oneself does not increase when key features are absent, but the accuracy of the orientation does.

This pilot study has provided empirical evidence for determining which objects in a scene should be selectively rendered to the highest quality as the user will be attending to them, and which objects are less perceptually important and can thus be rendered at a lower quality. The selection of features used in the orientation task is mostly driven by proximity and size. Certainly the nearest or biggest features are the most important. For example a big factory chimney or a nearby traffic light will probably be chosen as key features by the user. There is much more work that needs to be done before these results can be incorporated into a perceptually driven selective renderer for navigating on mobile devices. The creation of an ordered list of features for rendering is the key issue for this approach.

Future work will show selectively rendered images to users to confirm that they do not in fact notice the rendering quality difference. Other issues that will also be considered are: how different lighting and weather conditions may affect the perception of the scene and whether the method of orientation is different for men and women. The empirical data which will be gathered will then be used to modify the Itti and Koch model [5] to develop an urban navigation saliency map based on the GPS position of the specific user and the prevalent weather to ensure the user can orientate himself/herself rapidly to a high precision and from there navigate efficiently through the unfamiliar urban environment.

6. ACKNOWLEDGMENTS

This work is partially supported by the FCT (Portuguese science and technology national foundation), the POSI, the European Union and FEDER through the project POSI / CHS / 48220 / 2002 entitled "3D4LBMS - Three-dimensional Urban Virtual Environment Modelling for Location Based Mobile Services".

7. REFERENCES

- [1] K. Cater, A. Chalmers, and P. Ledda. Selective quality rendering by exploiting.
- [2] K. Cater, A. Chalmers, and G. WARD. Detail to attention: Exploiting visual tasks for selective rendering. *Eurographics Rendering Symposium, Leuven*, June 2003.
- [3] J. Ferwerda and P. et al. A model of visual adaptation for realistic image synthesis. In *Proceedings of SIGGRAPH 1996*, pages 249–258. ACM, 1996.
- [4] D. Greenberg, K. Torrance, P. Shirley, J. Arvo, J. Ferwerda, S. Pattanaik, A. Lafontaine, B. Walter, S. Foo, and B. Trumbore. A framework for realistic image synthesis. In *Proceedings of SIGGRAPH 1997 (Special Session)*, pages 477–494. ACM, 1997.
- [5] L. Itti and C. Koch. A saliency-based search mechanism for overt and covert shifts of visual attention. *Vision Research*, 40(10-12):1489–1506, 2000.
- [6] D. Lubeck and B. Hallen. Perceptually driven simplification for interactive rendering. In *Proceedings of 12th Eurographics Workshop on Rendering*, pages 221–223. Eurographics, 2001.
- [7] P. Maciel and P. Shirley. Visual navigation of large environments using textured clusters. In *Proceedings of Symposium on Interactive 3D Graphics*, pages 95–102, 1995.
- [8] A. Mack and I. Rock. *Inattentional Blindness*. Massachusetts Institute of Technology Press, 1998.
- [9] A. McNamara, A. Chalmers, T. Troscianko, and I. Gilchrist. Comparing real and synthetic scenes using human judgements of lightness. In *12th Eurographics Workshop on Rendering*, pages 207–219, 2000.
- [10] K. Myszkowski, T. Tawara, H. Akamine, and H. Seidel. Perception-guided global illumination solution for animation rendering. In *Proceedings of SIGGRAPH 2001*, pages 221–230. ACM, 2001.
- [11] M. Ramasubramanian, S. Pattanaik, and D. Greenberg. A perceptually based physical error metric for realistic image synthesis. In *Proceedings of SIGGRAPH 1999*, pages 73–82. ACM, 1999.
- [12] A. Yarbus. Eye movements during perception of complex objects. *Eye Movements and Vision*, Chapter VII:171–196, 1967.
- [13] H. Yee, S. Pattanaik, and D. Greenberg. Satiotemporal sensitivity and visual attention for efficient rendering of dynamic environments. *ACM Transactions on Computer Graphics*, 20(1):39–65, 2001.

Exploring the Potentials of Combining Photo Annotating Tasks with Instant Messaging Fun

Yuechen Qian

Eindhoven University of Technology
5600 MB, Eindhoven, The Netherlands
y.qian@tue.nl

Loe M.G. Feijs

Eindhoven University of Technology
5600 MB, Eindhoven, The Netherlands
l.m.g.feijs@tue.nl

ABSTRACT

The combination of photo annotation tasks and with instant messaging fun offers great potentials to both end-users and researchers. In this paper, we first describe our prototype system that allows users to share and annotate digital photos over the Internet while they are chatting online. In addition to manual annotation, our system can extract information from conversations to generate extra annotations. The advantage of using our system is that the boring and tedious task of annotating photos is turned into an essential part of an attractive fun activity, viz. online chatting. Extracting meaningful information from instant messages is challenged by abbreviations, pronouns, jargon, ellipsis, grammatical errors, ambiguities and asynchrony, all of which frequently appear in message conversations. In the paper we provide a roadmap, i.e. a systematic analysis of linguistic aspects of automated interpretation of message conversations.

Categories and Subject Descriptors

H.2.8 [Database Management]: Database Applications—*Image databases*; H.5.2 [Information Interfaces and Presentation]: User Interfaces—*natural language, prototyping*; H.5.3 [Information Interfaces and Presentation]: Group and Organization Interfaces—*collaborative computing*

General Terms

Design, management, human factors, languages

Keywords

Digital photos, metadata, annotation, instant messaging systems

1. INTRODUCTION

Many systems have been prototyped to facilitate the easy and attractive use of digital photos. Using a zoomable user interface, PhotoMesa Image Browser allows users to visually

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MUM 2004, October 27-29, 2004 College Park, Maryland, USA.
Copyright 2004 ACM 1-58113-981-0 /04/10 ...\$5.00.

search and browse a large number of photos [1]. Shoebox provides a speech interface enabling users to add voice annotation to photos [2]. To support storytelling activities, a photo browsing device was designed [3]. More advanced systems support similarity-based image browsing [4], automatic image clustering [5], and photo concept browsing [6]. Unfortunately, the lack of meaningful metadata which describe the place, time and event in which photos were taken, remains one of the problems that hinder the easy use of photos. Most recent photo organizing tools, such as ACDSee, iPhoto and Adobe Photoshop Album, allow manual annotation. In the PhotoFinder system [10], users can select text annotations from a predefined schema and associate them with photos by drag-n-drop operations [11]. Using Minolta's DiIMAGE Messenger [13], users can link textual, visual and audio annotations to any part of a photo. Yet it is still recognized that annotating photos is a boring and tedious job, not fun or rewarding at all. The amount of metadata associated to photos is so limited that the power of metadata is not unleashed yet.

Extensive research has been done on extracting content information from images [7, 8, 9]. Most approaches deliver objective visual characteristics of images, but yield hardly the types of information that people use in their memory recollection. With Global Positioning Systems (GPS), it becomes feasible to embed geographic position information into image files. However, the user-understandable location information like "on the beach", "in a restaurant" or "at home", can't be obtained from GPS devices. Truly meaningful metadata can only come from end users.

Instant messaging systems, such as MSN Messenger and Yahoo Messenger, provide a variety of technical solutions for social communication. Users of such systems can send messages, access a shared white board and play games! Picasa's Hello system [14] allows online users to browse photos simultaneously while chatting. Recent reports confirm a significant growth of both number of users as well as intensity of use [12]. Most recently developed cell phones with built-in cameras and touch screens even offer users the possibilities of taking pictures, annotating them, and sharing them via MSN Messenger on the fly! The information passing by instant messaging systems provides a rich source of data that can potentially be used in the quest for meaningful metadata of digital photos.

We developed a chatting system that not only allows users to share and annotate photos, but also can automatically extract meaningful data from messages. The extracted information is treated as extra annotations and stored in local

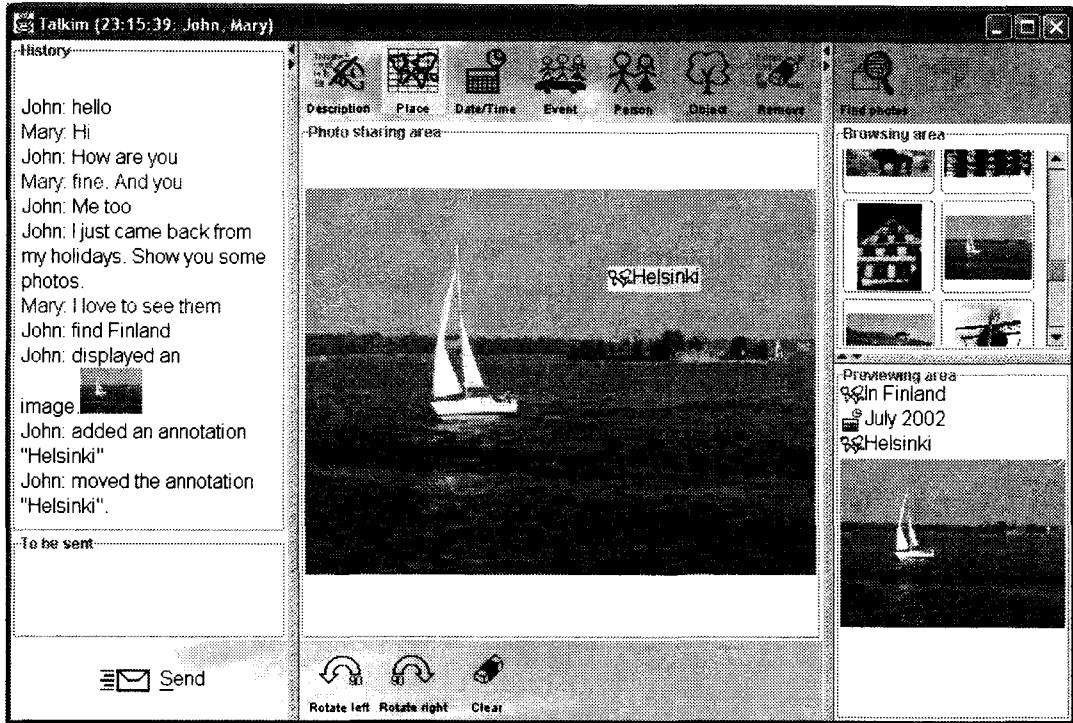


Figure 1: A snapshot of Talkim's user interface.

databases. Users can use such information to categorize and search their photos. Linguistic analysis is the key element to improve the analysis and understanding of instant-message-based conversations. We developed a roadmap towards automated interpretation of textual information in message conversations. The rest of this paper is organized as follows. We present our prototype system in Sect. 2 and explain its annotation extraction mechanisms in Sect. 3. In Sect. 4 we provide several important results of the first round of user trials. We present the roadmap in Sect. 5. Finally we draw a few conclusions in Sect. 6.

2. ANNOTATING WHILE CHATTING

We developed the Talkim system for people to share photos online. Fig. 1 shows Talkim's user interface. This user interface consists of the message area on the left, the photo sharing area in the middle and the photo search area on the right. Participants of a chatting session have the same view on the content of message and photo sharing areas.

To share a photo, a user simply drags an image file from a Windows directory and drops it onto the photo sharing area. The image is displayed locally, sent to other participants, and then displayed in the photo sharing area of their chatting windows, as in Picasa's Hello system. Different from the Hello system, Talkim's photo sharing area does only display photos, but also functions as a whiteboard: users can simultaneously put text annotations to any places in the photo sharing area and move annotations freely. To the best of our knowledge, it is for the first time that the photo annotating task is achieved by using the online whiteboard concept.

The idea of allowing users to put annotations on photos is quite similar to PhotoFinder's direct annotation function.

This functionality becomes rather essential in our system: It helps to raise attentions and create the focus of a discussion. For example, a user would otherwise have to type lengthy sentences in order to explain interesting elements in a photo. It is exactly the lengthy sentences that will harm the fluency of a message conversation. With the help of freely placable annotations, the user can easily pinpoint interesting elements such as a building, a person, or a pet in the photo. Freely placable annotations are complementary to messages during online photo sharing.

2.1 Types of annotation

In the Talkim system, annotations are categorized: the user first clicks any annotation type button and then types information on a selected place of the photo in view. Six annotation types are provided: *general description*, *location of taking*, *time/date of taking*, *event of taking*, *persons in photo* and *objects in photo*, as indicated by the buttons above the photo sharing area in Fig 1. Categorizing annotations not only help users to better organize their photos, but also help to improve linguistic analysis of message conversations.

We could have included a predefined schema from which users select specific annotations. For example, to add an annotation "Helsinki", a user can select a city name from a location hierarchy. Such a selection involves several interaction steps and might hamper the fluency of conversations. Furthermore, what the user wants to write might be "on sea" or "on a boat". Similar examples can be found of time-related annotations. People users may prefer to put more meaningful information like "last summer" or "in the evening". One consequence of not using the schema selection option is the loss of accuracy of annotations: Without

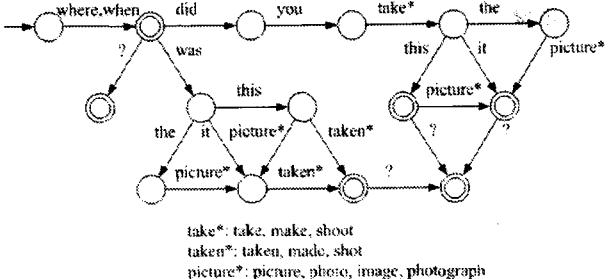


Figure 2: The finite-state-machine diagram is used to parse where/when questions.

validating mechanism, it is likely that users choose one annotation type while the entered information is of another type. Such errors can be corrected at later stages and even without correction, users can still find back photos by using keyword-based search.

2.2 Integrated interface

To help users to easily retrieve their photos, a photo search area is integrated into the chatting window, as illustrated in Fig. 1. Initially the search area is hidden in a splittable panel. The search area is shown when the user types in “find Helsinki” in his message box. Alternatively, the user can click the open button on the splitbar. Here “Helsinki” can be replaced by any sequence of keywords. Search results are displayed in the search area. Once a photo is selected, the user can preview associated annotations and browse the photos that are stored in the same directory as the selected one. To share a photo, the user drags the photo from the search panel and drops it on the sharing area. Using our system, users can quickly find and share photos, without using other applications to browse and select photos.

3. EXTRACTING ANNOTATIONS

Talkim can extract annotations from messages sent by users. Four annotation types were worked out in our prototype system. They are *location of taking*, *time/date of taking*, *event of taken* and *persons in photo*.

As using natural language processing techniques in any application, extracting metadata from messages is a sophisticated process. Lexically, it should categorize correctly various words. Syntactically, it should parse diverse sentence structures. Furthermore, it involves semantic and pragmatic analysis. Finally, proper words need to be extracted as annotations. Processing message conversations is even harder, due to abbreviations, pronouns, jargon, ellipsis, grammatical errors, ambiguities and asynchrony that frequently appear in message conversation.

We took a rather practical approach. Our objective was to derive the annotation type of a message, either from the sentence itself or from the context of the message; if the message falls into one of the annotation types, the message is stored as an annotation as a whole, instead of words of interest being extracted.

3.1 Parsing messages

In the Talkim system, the first type of parsable messages are complete fact sentences, for instance, “This photo

was taken in Finland.”. Once recognized, such messages are directly added to the photo sharing area as annotation. The second type of parsable sentences are complete questions, such as “where was it taken?”. Such message provide contexts for processing subsequent messages. The third type of parsable messages includes ellipsis sentences such as “where?”, “when?”, “in a restaurant”, “July”, “my friends”, or messages that are not parsable or understandable by our system at this stage. We defined several schemas to recognize and categorize different types of messages. Fig. 2 shows part of the finite-state-machine diagram that parses both complete questions and ellipsis questions on where/when a photo was taken.

3.2 Analyzing dialogues

Interpreting the third type of messages needs contextual information. The message “In a restaurant” is not processed when there is no photo in the photo sharing area. When this message is a reply to a question, say “Where was it taken?”, it is then treated as a location-related annotation. Moreover, the “current” topic in a message conversation might be constantly shifting. After he sends a photo to the others and discusses the photo for a while, the user may switch to another topic while the photo is still being viewed. Identifying these dialogues is essential to annotation extraction.

In order to identify dialogues, Talkim seeks for linguistic signals in messages. After a photo is shown in the photo sharing area, Talkim starts to look for the first message that contains a complete question sentence describing the photo by an image receiver or a complete fact sentence describing the photo by the image sender. For example, a triggering message can be “When was it taken?” from a photo receiver or “It was taken in July” from the photo sender.

A response to a question message asked by the photo receiver might contain a complete fact sentence. In this case, the response message can be handled by Talkim’s sentence parsing and categorization. If the response message from the image sender is a phrase, abbreviation, ellipsis or if it is not recognized at all, however, Talkim derives the annotation type of the message from the preceding question message asked by the photo receiver.

4. USER TRIALS

We conducted several user trials on the Talkim system. The intention of this evaluation was to collect field data, especially text messages, for further linguistic analysis and to obtain early feedback on the usability of our prototype.

Four groups of people were invited, each consisting of two persons who know each other well. The first two groups were students from the Department of Industrial Design at the Eindhoven University of Technology. The third group of participants were teenage girls from a secondary school. The last group of users were a couple. Each group used our system to chat and share photos for 30 minutes.

We recorded all messages that were sent by users during their online conversations, and identified several occurrences of question-and-answer dialogues. The left picture in Fig. 3 is part of the log file recording the conversation by a student group. It shows that one user sent a question “where did you take the picture?” and another user replied “at home”. As shown in this picture, Talkim identified this dialogue and treated “at home” as an annotation. The annotation type *location of taking* was assigned to the message “at home” in

10 Mar 2004 09:30:04 GMT	Youpca	displayed an image. 00fabb9e6818_036589_Car 1.jpg	
10 Mar 2004 09:30:10 GMT	Youpca	ShowMessage nice car	
10 Mar 2004 09:30:11 GMT	Youpca	ShowMessage hu	
10 Mar 2004 09:30:14 GMT	Youpca	ShowMessage h	
10 Mar 2004 09:30:21 GMT	wouter	ShowMessage where did you take the picture?	
10 Mar 2004 09:30:34 GMT	Youpca	ShowMessage at home	
10 Mar 2004 09:30:34 GMT	Youpca	abstracted an annotation "at home"	
10 Mar 2004 09:30:37 GMT	Youpca	moved the annotation "at home"	
10 Mar 2004 09:30:37 GMT	Youpca	added an annotation "mitsubishi"	
10 Mar 2004 09:30:43 GMT	Youpca	ShowMessage cool	
10 Mar 2004 09:31:01 GMT	wouter	ShowMessage yes	
10 Mar 2004 09:31:03 GMT	wouter	ShowMessage it is	
10 Mar 2004 09:31:25 GMT	Youpca	ShowMessage ask me more	
10 Mar 2004 09:31:41 GMT	wouter	displayed an image. 00B8bbcd8e8_0b246a_000_1087.JPG	
10 Mar 2004 09:31:42 GMT	Youpca	ShowMessage 6-9-2004	
10 Mar 2004 09:31:56 GMT	Youpca	ShowMessage when did you take it?	
10 Mar 2004 09:31:57 GMT	wouter	ShowMessage rotterdam	
10 Mar 2004 09:31:57 GMT	wouter	abstracted an annotation "rotterdam"	
10 Mar 2004 09:32:04 GMT	Youpca	ShowMessage ?	
10 Mar 2004 09:32:12 GMT	wouter	ShowMessage ok	
10 Mar 2004 09:32:15 GMT	wouter	ShowMessage sorry	
10 Mar 2004 09:32:20 GMT	wouter	ShowMessage dyslexia)	
10 Mar 2004 09:32:49 GMT	Youpca	ShowMessage lets puont some music	
10 Mar 2004 09:32:54 GMT	Youpca	ShowMessage put	
10 Mar 2004 09:33:01 GMT	wouter	ShowMessage i took this picture at one pm	
10 Mar 2004 09:33:09 GMT	Youpca	ShowMessage cool	

Figure 3: Part of the log of the conversation performed by one student group.

our databases.

The right picture in Fig. 3 shows another dialogue. In this example, one user asked “when did you take it?” and the other user replied “rotterdam”. Both users immediately noticed this error and the answer to the question “when did you take it?” comes after several messages. The asynchrony in this example confused our annotation extraction mechanism. We will address the asynchrony of message conversations in Sect. 5.

There are several interesting observations during our user trials. First, participants from all groups liked the integration of photo sharing and chatting. The group of a couple continued to use our system for a while after their session was finished. Secondly, all participants liked the functionality that they can put annotations onto the photos and move annotations around. Our analysis shows that large amount of metadata were added to photos. Thirdly, for the group of teenage girls, right after they found that they could move annotations, they started to play a “grabbing” game: one moves an annotation while the other tries to grab it and move it to another place. They thought it was fun and they played this game many times throughout the user trial. Finally, it was also found in the group of teenage girls that they often use the on-screen annotating functionality to chat, not only for annotating photos.

5. ROADMAP

Linguistic analysis is the key element to improve the automated analysis and understanding of message conversation. In this section we make an inventory of the scientific problems and possible solutions that have to be addressed in order to further develop and improve applications that are similar to Talkim. In order to break down this problem into sub-problems, we follow a compositional approach [18], assuming that meanings can be derived by following the syntactic structure of the text. Therefore we distinguish three levels: words, sentences and dialogues. In general, the analysis results of the lower level are used to achieve an analysis result at the higher level.

5.1 Words

At this level words have to be recognized, that is, they have to be assigned a grammatical category and if possible aspects of their meaning have to be retrieved.

Two issues deserve special attention: errors and abbreviations. Their role is much larger for chat texts than for traditional texts. Typical errors are “picuter” for “picture”, “anohter” for “another”, “eaysy” for “easy”, “girlfriendshouse” for “girl friend’s house” (these are all examples from our user trials). If the conversation language is known, a traditional minimization of the Levenshtein edit distance [15] to the closest dictionary word should suffice for the first three errors. There exist efficient dynamic programming algorithms to do this but they should be adapted to reflect the fact that certain typing errors are more likely than others (for example interchanged letter order, which is a typical result of two-finger typing, e.g. “picuter” and “anohter”). The weight of the edit actions such as adding a letter or flipping two letters can be taken as parameters of the Levenshtein edit distance. As “girlfriendshouse” demonstrates, gluing words together is another typical mistake. We propose a separate dictionary-based splitter for long un-recognizable words.

Next we discuss abbreviations, which are more common in message conversations than in traditional text (for example business letters). Abbreviations and chat jargon are derived in a number of ways: real abbreviations like “LOL” for “Laugh Out Loud”, pronunciation-based short-hands like “IC” for “I see”, old codes from ham-radio like “QSO” for “conversation” or “73” for “best regards”, and abbreviations based on the English pronunciation of certain numbers like “F2F” for “face to face”.¹. Again a dictionary-approach seems appropriate but we propose using on-line dictionaries with mechanisms or people in place to keep them up-to-date. There will be a need to deploy several dictionaries containing specific slang per language, per country, and even for a specific per user-group.

After errors and abbreviations have been eliminated by

¹These examples come from www.stevegrossman.com/jargpge.htm.

a pre-processor, the real word-level recognition can begin. A very important word category is pronouns ("I", "me", "you", "he", "him", ..., "this", "there", "who", "where", ...). The personal pronouns are easy to recognize. Their meaning is that they indicate persons and they will be a rich source of annotations, provided the text analysis is powerful enough to identify the person meant. A simple but effective source of identities are the chat-names of the sender and the addressee (although a chat box is not two-person, the dialogue level analysis may find that two persons are chatting with each other temporarily and in such a context, the pronoun "you" gets a meaning). Pronouns such as "he", "him", "she", "her", ..., "this", "there", assume that a person or a location has been introduced earlier so again a higher level can provide the context. In particular the topic-level analysis can accumulate a list of events, locations, people, times, and things which can be used for trying to dis-ambiguate a pronoun by identifying the name of the person or the place meant. The frequently occurring "it" can be handled in the same way. The "wh" pronouns like what, where, who, when, which etc. indicate questions and they can be used to determine the annotation type of the subsequent answer (event, location, people, time, object, respectively). In normal English the "wh" words are also used for sub-ordination ("I wonder which dish they picked") but our user trials suggest that chat text sentences are short and simple and that all the complications of sub-ordinate sentences are not needed.

Another important word category is prepositions, which, like some of the pronouns, determine the annotation type ("in" is either place or time, "at" is place, "during" is time or event). As the examples, show some further dis-ambiguation is necessary, which is doable if the following word is likely to be a place/time/event indication. Names are important too and it is not hard to imagine tables of places (Amsterdam, London, Kyoto, ...), times (January, February, ... spring, summer, yesterday, today, etc.). They also determine the annotation type. Also note that words like "today" and "yesterday" can be automatically identified with real dates by using the current date, that of course is known by the application itself. Finally we mention the issue of flection which can be handled at the word level (recognizing "drove" as the past tense of "drive" and "him" as the 4-th case of "he").

5.2 Sentences

At this level sentences have to be recognized, that is, they have to be assigned a sentence type, the phrase structure has to be parsed and if possible aspects of their meaning have to be retrieved.

Computerized natural language understanding and translation has been a research area for decades and we expect that many existing techniques can be re-used for annotation extraction. The main difficulties at the sentence level do not lie in the computer programming but in the complexities of human language itself. Even at a syntactic level there is no single stable and widely accepted theory. Specific theories have been developed for dealing with certain aspects, such as Montague grammars [16] facilitating the proper treatment of quantifiers, minimalist grammars [17] facilitating the parsing of complex sentences that contain subordinate clauses, M-grammars [18] facilitating compositional translation, Categorial grammars[19] which handles the polymorphy of words in a natural way , and type-logical gram-

mars [20] which can deal with logical aspects very well. All of these could be used, in principle, but strangely enough, instant messages do not seem to contain complex subordination or complex logic. Though messages are short and simple, the semantic analysis still is a big challenge, even a bigger challenge than for business letters because of the frequent ellipsis and linguistic pointers (it, he, that, then, etc.) in message conversations.

We propose to split the discussion into two parts: syntax and semantics. For syntax we propose to use a straightforward phrase structure grammar, similar to the Backus-Naur Form grammars used in computer science. Any bottom-up parsing technique or backtracking top-down technique will do the job since the sentences are short and no recursion is needed. Each sentence must be classified as either statement, imperative, exclamation, question or answer. All of these appear in our user trials. Word order is very helpful for this classification (the auxiliary verb do in English and inverse word order in German, French, Dutch, etc, next to the obvious question mark are typical for questions). Imperatives have no subject or begin with "lets", "please", etc.). In parsing the sentence itself, prepositional phrases are particularly important because they contain many of the clues that lead to new annotations ("in France", "on the beach").

For a semantic analysis it is necessary to develop underlying models. To a certain extend this will be feasible for personal preferences (certain person liking or disliking places, objects, etc.). It also seems feasible for relations between people (who is whose girl-fried?, etc.). What will be very hard is recognizing the meta-level discussions that are going on ("sorry", "dyslexia", ";"). At present, it also appears that emotional icons are an indication that some subtle joke is going on, which will be the hardest thing to fit into a semantic model.

5.3 Dialogues

At this level dialogues must be recognized, which will be used to determine whether recognized sentences should be treated as annotations or not. As indicated in Sect. 3.2, such dialogic analysis is necessary to handle ellipsis and ambiguous sentences and sentences that recognition systems may fail to parse. Asynchrony is identified in our study as one of the most significant distinctions from speech dialogues.

For example, during a dialogue one recipient of an image sends a message "Where was this photo taken?" to the sender of the image. He might send another message "when?" before he receives any reply from the image sender. On the image sender side, the sender now see two questions in queue. He response firstly "In Helsinki" and then reply "Last July". Alternatively, he can also reply "Last July" and then send a message "In Helsinki". While such communication does not often appear in speech dialogue, they appear frequently in message dialogues. In either way, the sender's responses are justified. In the first case, he follows the order in which the questions were asked. In the second case, he answers the most recent question first, which is most fresh and often has the role of "overriding" the previous question. In Fig. 4, the first two diagrams illustrate the above-mentioned two scenarios. The third diagram is similar to the second one, except that the first question was repeated. This repetition is a kind of enforcement or reminder which the sender of the image should react to. The annotation extraction should also handle such phenomena.

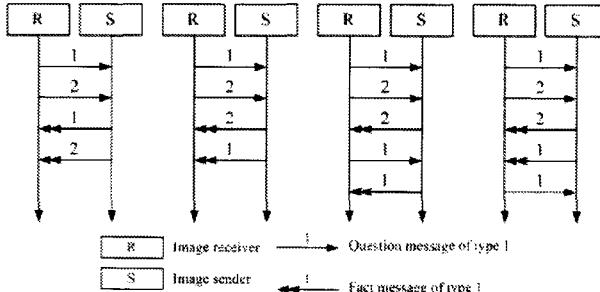


Figure 4: Several scenarios of asynchrony in message dialogues.

The fourth diagram shows a repetition of the first question comes after the answer. In this case, the question message can be dropped.

In message conversations, participants may talk about things not only in a sequential manner, but also in an interleaving manner. The second example depicted in Fig. 3 provides a real example of this sort. During a conversation, one user asked a question “when did you take it?”. The other user replied “rotterdam”. Both immediately noticed this strange answer. The user who gave the wrong answer, sent a few messages, one is “dyslexia:)”. As can be seen, the conversation at this moment already switched to a dialogue on languages, not on the photo being viewed. The dialogue on the photo resumed when the answer to the question appeared. Interleaving of several parallel threads of topics are quite common in message conversations, which challenges automated annotation extraction. This is one of our ongoing research topics. We also expected that formal dialogue theory will be of help in solving these issues [21, 22, 23].

6. CONCLUDING REMARKS

In this paper, we reported our work of integrating photo annotating tasks into instant messaging applications. Our system turns the boring and tedious task of annotating photos into an essential part of an attractive fun activity, viz. online chatting. Whereas taking photos, editing them and sharing them with others is considered a creative and highly rewarding process by many people, the difficulty of classifying and grouping them has always been difficult and tedious (as was the case in the days of non-digital photography). Most of the propositions that promise to automate these tasks rely on the assumption that photos are annotated, which in fact they are not, thus shifting the problem from one tedious task to another even more tedious task. Precisely the latter task is turned into fun by the type of application investigated here.

The result of the user trials of our system shows the enthusiasm that the participants had on our prototype system. The integration of photo browsing and chatting facilitates online photo sharing and discussing activities. Putting this annotating task in a social context makes the photo annotating task fun and appealing. Allowing users to freely annotate photos on the photo itself produce a large amount of annotations without many computational difficulties. Such useful information can be directly used in photo searching even without further processing. We conjecture that most of the fun and entertainment value of this type of chatting

comes from the co-presence: a reuse of being together in a mediated environment. We did not do a formal user experiments yet, but it seems that presence theory [24] provides a solid basis for further investigation.

Finally, the roadmap represented here shows the way ahead to improve automated annotation extraction by reusing existing knowledge from natural language processing and researching on new linguistic challenges.

7. ACKNOWLEDGMENTS

The authors would like to thank Peter Peters for his remarks and suggestions on this paper.

8. REFERENCES

- [1] Bederson, B.B.: PhotoMesa: a zoomable image browser using quantum treemaps and bubblemaps. In Proc. ACM UIST 2001. 71–80
- [2] Mills, T.J., Pye, D., Sinclair, D., Wood, K.R.: Shoebox: A digital photo management system. Technical Report 2000.10, AT&T Laboratories Cambridge, 2000.
- [3] Balabanovic, M., Chu, L., and Wolff, G.: Storytelling with Digital Photographs. In Proceedings of CHI 2000. 564–571
- [4] Rodden, K., Basalaj, W., Sinclair, D., Wood, K.: Does organisation by similarity assist image browsing? In Proc. SIGCHI CHI 2001. 190–197
- [5] Platt, J.C., Czerwinski, M., Field, B.A.: PhotoTOC: Automatic clustering for browsing personal photographs. MSR-TR-2002-17, Microsoft Research, 2002.
- [6] Qian, Y., Feij, L.M.G.: Stepwise Concept Navigation. In de Moor, A., Ganter, B. (Eds.): Using Conceptual Structures of ICCS 2003. Germany. 2003. 255–268
- [7] Pentland, A., Picard, R., Sclaroff, S.: Photobook: Content-based Manipulation of Image Databases. In SPIE Proceedings Vol. 2185 (1994) 34–47
- [8] Jain, A., Vailaya, A.: Image retrieval using color and shape. Journal of Pattern Recognition 29(8): 1233–1244, August 1996.
- [9] Smith, J.R., Chang, S.: VisualSEEk: A Fully Automated Content-Based Image Query System. ACM Multimedia, 1996, 87–98.
- [10] Kang, H., Shneiderman, B.: Visualization Methods for Personal Photo Collections: Browsing and Searching in the PhotoFinder. In Proc. IEEE ICME 2000. 1539–1542
- [11] Shneiderman, B., Kang, H.: Direct Annotation: A Drag-and-Drop Strategy for Labeling Photos. Proc. International Conf. on Information Visualisation, 2000.
- [12] Madden, M.: America’s Online Pursuits: The changing picture of who’s online and what they do. Pew Internet & American life project. www.pewinternet.org. 2003.
- [13] Minolta Ltd. DiMAGE Messenger. <http://www.dimagemessenger.com/>
- [14] Picasa Inc. Hello. <http://www.picasa.net/>
- [15] Levenshtein, V. I.: Binary codes capable of correcting deletions, insertions and reversals. Doklady Akademii Nauk SSSR 163(4), 1965. 845–848

- [16] Montague, R.: Formal Philosophy; Selected papers of Richard Montague. Yale University Press, New Haven. (1974).
- [17] Radford, A.: Syntactic theory and the structure of English. Cambridge University Press (1997).
- [18] Rosetta, M.T.: Compositional Translation. Kluwer International Series in Engineering and computer science: Volume 273. (1994).
- [19] Moortgat, M.: Categorial investigations. University of Amsterdam (1988).
- [20] Carpenter, B.: Type-logical Semantics. MIT Press (1997).
- [21] Bunt, H: Dynamic Interpretation and Dialogue Theory. In M.M. Taylor, D.G. Bouwhuis, F. Neel(eds.): The Structure of Multimodal Dialogue II. John Benjamins. 1998.
- [22] Smith, R.W., Hipp, D.R.: Spoken Natual Languages Dialog Systems: A Practical Approach. Oxford University Press. 1994.
- [23] Frederking, R.E.: Integrated Natual Language Dialogue: A Computational Model. Kluwer Academic Publishers. 1988.
- [24] Freeman, J., Lessiter, IJsselsteijn, W.A.: An Introduction to Presence: A Sense of Being There in a Mediated Environment. The Psychologist. April 2001.

Pocket PhotoMesa: A Zoomable Image Browser for PDAs

Amir Khella, Benjamin B. Bederson

Human-Computer Interaction Lab

Institute for Advanced Computer Studies, Computer Science Department

University of Maryland, College Park, MD 20742

+1 301 405-2764

{akhella, bederson}@cs.umd.edu

1. ABSTRACT¹

Small devices such as Palm and Pocket PC have gained wide popularity with the advance and affordability of mobile technologies. Image browsers are among popular software applications on small devices. The limitations introduced by these devices such as screen resolution, processing power and storage impose a challenge for multimedia applications designed for larger displays to adapt to small screens. For an image browser, layout of images and navigation between them are critical factors of the users' experience.

Motivated by these challenges, we developed Pocket PhotoMesa: an image browser for the pocket pc that employs quantum strip Treemaps for laying out images and Zoomable User Interfaces for navigation. In this paper, we discuss the development of Pocket PhotoMesa and we describe a usability study comparing the performance and users' experience using Pocket PhotoMesa and ACDSee image browser (a current commercial offering) for the Pocket PC.

1.1 Keywords

Image browsers, Information visualization, mobile devices, mobile multimedia, Treemaps, Zoomable User Interfaces (ZUIs), Animation, Graphics, Pocket PC.

2. INTRODUCTION

The past decade witnessed a major advance in the development of mobile technologies that provided ubiquity and affordability of small devices, fitting every day's needs and everyone's pocket. Starting in the mid nineties, several companies introduced monochrome portable displays for scheduling and address books. Few years later, Pocket PCs were introduced with color screens, more processing power, and larger storage. However, many limitations still exist for application development on mobile devices: screen resolution and size, limited processing power and stylus interaction are among the toughest challenges for mobile application developers.

Popular applications on PDAs include Personal Information Managers (PIMs), file explorers, board games and image browsers. Due to limited screen size and resolution, image browsing applications use scroll bars to cycle through image thumbnails and locate images of interest. Since scroll bars require finely tuned pointing skills on small devices, we designed and implemented an image browser that eliminates the need of scroll bars. We use Treemaps [3] to layout image thumbnails on a single screen and Zoomable User Interfaces (ZUI) [1] to navigate through images. In this paper, we show our design and the results of a usability study that investigates users' performance and satisfaction with our interface compared to a traditional image browsing interface.

We believe that laying out images on a single screen in groups corresponding to physical folders will help users to visually identify themes of each group of images and be able to locate images faster than using traditional interfaces that display only folder names and We also believe that animated zooming will enable users to maintain the navigation context and improve the users' experience.

3. RELATED WORK

3.1 Zoomable User Interfaces (ZUIs)

Zoomable User Interfaces present users with a single view of large information space populated with graphical objects (images and image groups in our case). The interface allows users to navigate the object hierarchy using smooth animated zooming through different levels of the details. Initially, a ZUI renders the information space in a single screen allowing users to get an overview of the information domain allowing them to identify themes and patterns in the big picture.

ZUIs were introduced more than thirty years ago in Sketchpad interface [4] which implemented an interactive object oriented 2D graphics system that enabled zooming and rotation of rendered objects. Almost a decade later, several systems started implementing interactive zooming: Spatial Data Management System (SDMS) [5] implemented two levels of semantic zooming, Pad and Pad++ [1] were developed as toolkits for building Zoomable User Interfaces. Zooming has also been a component of several other interfaces and toolkits developed later. Two of the major zooming toolkits available are Jazz [7], and its successor Piccolo [8], a toolkit for interactive structured graphics available in Java and C# on the desktop and on Pocket PC. These toolkits have been used in several domains such as slide show presentations [2], navigating ontology information [9], image browsing [10] as well as several other applications.

ZUIs have shown a statistically significant interaction improvement over several image browsing interfaces, but have not yet been shown to outperform traditional thumbnail grid

¹ Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

interfaces [11]. The same study concluded that the number of images displayed within the browser is an important factor for users' performance and error rate. However, we believe that the potential for ZUIs is more promising on small devices when screen space is at an even greater premium.

A basic characteristic of ZUIs is that objects can be rendered quite small when you zoom out. Since image thumbnails can be difficult to understand when they are small, it is crucial that the thumbnails focus on the relevant part of the image. One approach to this is via automatic image cropping to generate thumbnails focusing on the salient parts of the image [6]. We think this is an important approach that should be considered in photo browsers, but is not one that we have pursued in Pocket PhotoMesa.

3.2 Treemaps

Treemaps are space-filling visualizations for large hierarchical datasets where the display area is divided into several rectangles whose areas correspond to some attribute of the dataset. Among the algorithms used for the Treemap layout are slice and dice [3], clustered [12], squarified [13] and strip Treemaps [16]. Treemaps are used in several visualizations such as SmartMoney's market map [14], image browsing [15] and many other domains. In image browsing, Treemaps suffered from the problem of aspect ratio: thumbnail sizes varied from one group to another. Quantum treemaps and quantum strip treemaps [16] solved the aspect ratio problem by using fixed size elements (quantum) across different areas.

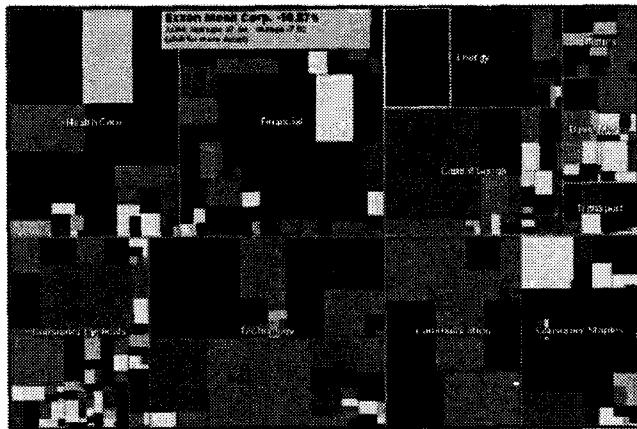


Figure 1: SmartMoney (www.smartmoney.com): An example of the use of treemaps in practice.

3.3 Desktop PhotoMesa

Desktop PhotoMesa is a zoomable image browser for the desktop PC(www.photomesa.com). The browser enables users to view hierarchies of directories in a Treemap layout and browse through them in an animated navigation mechanism. PhotoMesa uses quantum strip Treemaps to arrange images over a single screen. It also uses semantic zooming to display different thumbnail resolution at each level of detail. PhotoMesa is efficient for browsing large set of images, which would otherwise span several screen pages and requires the use of scrollbars in traditional image browsers.

Figure 2 shows 132 images loaded into PhotoMesa from 8 groups into a single screen. When users click inside a group, the

view is smoothly zoomed into this directory. When the user is zoomed in to a group, a viewfinder, which can grow and shrink in size using the mouse wheel, allows users to specify a subset of the group images to zoom into or double click on a single image to zoom into a full resolution version of the image. At any time, users can press the right mouse button to zoom out to the previous zoom level. Users can also use arrow keys to navigate between images and groups. When the cursor is left to dwell over a thumbnail for a short period, a 200 pixel wide preview of this image is displayed in a tool tip overlay fashion (shown in the figure). The preview is removed as soon as the mouse moves.

Desktop PhotoMesa is implemented in JAVA using the Jazz toolkit [7]. Jazz is a polyolithic structured graphics toolkit developed at the Human Computer Interaction Lab to support building interfaces in Java. The implementation of Jazz empowers programmers to create interfaces in a scene graph object oriented fashion. The scene graph is a hierarchy of nodes that represent relationships between objects. Functionality is built up by composing a number of simple objects. Complexity is therefore tackled by small reusable nodes that build up the scene graph.



Figure 2: Desktop PhotoMesa (www.photomesa.com)

4. POCKET PHOTOMESA

We faced several challenges during the design and the implementation phase to port PhotoMesa from the desktop to the Pocket PC. Limited interaction on the Pocket PC and small screen space imposed many restrictions on the interface design. In the following subsections, we present these challenges and the proposed solutions.

While the progress in mobile technologies has been promising, mobile processors are still not powerful enough to cope with the increasing demands of multimedia applications. Current PDAs are powered by 60-600 MHz processor with 16 to 64 Megabytes of memory. The processing power and memory limitations were critical factors in developing efficient algorithms and interaction techniques tailored to fit these particular devices.

Another limitation is the screen size and resolution. Pocket PCs have a screen resolution of 240x320 with 8 to 16 bit color depth. It is obvious that with this limited screen size, there is a maximum number of images that can be displayed on one screen. Moreover, with the current color depth, it is even harder

to identify smaller thumbnails. A final limitation was imposed by the stylus input mechanism. A stylus has three modes of operation: up, down, and tap-and-hold, as opposed to the mouse having six modes: mouse over, left button up, left button down, right button up, right button down, and wheel scroll. It is obvious that the lack of these extra modes using stylus interaction introduce major design limitations, especially that all these interactions modes are used in the desktop version of PhotoMesa.

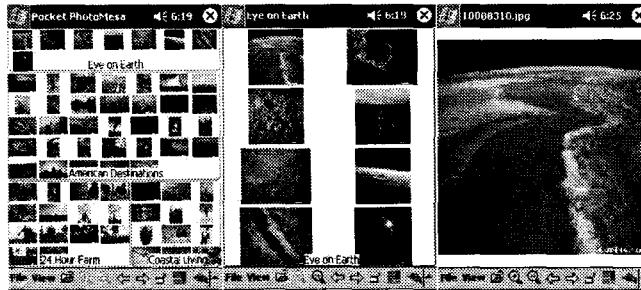


Figure 3: Three views of Pocket PhotoMesa at different zoom levels (www.photomesa.com).

4.1 Pocket PhotoMesa Interface Overview

Pocket PhotoMesa tackles the problem of limited screen space by using quantum strip Treemaps to efficiently layout images on the screen, minimize the amount of white space and render fixed size thumbnails for all the images.

Using the stylus, users can perform the following functionalities:

- Zoom into a specific group by tapping into a white space inside the group or the group's name.
- Zoom into an image by tapping on the image's thumbnail.
- When zoomed into an image, tapping on the image renders the full resolution version and enables users to pan around by dragging the stylus or go back to the fit-screen image mode by tapping on the image again.
- Users can also select an area of the image to zoom into by tapping and dragging to draw the zoom viewport.
- Tapping on a white space within a current level brings the user up one level.
- Hardware keys and toolbar icons can be configured to provide more functionality, such as reordering the group to fill the screen and predefined zooming levels.

4.2 Pocket PhotoMesa Implementation

Overview

While the original PhotoMesa was developed using Jazz toolkit, Pocket PhotoMesa was implemented from scratch without the use of any existing API. Several factors influenced this decision: First, there is no structured graphics toolkit available for building applications on the Pocket PC. In addition, Jazz was not suitable since the Java runtime environment is neither stable nor fast enough on the Pocket PC. Now, Piccolo.NET is available for Pocket PC (www.cs.umd.edu/hcil/piccolo), and so

at this point, we probably would have chosen to develop Pocket PhotoMesa using Piccolo.NET.

In our case, it was essential to develop a tailored implementation to optimize the application for maximum performance. The bottleneck was rendering smoothly animated zooming transitions. At each step of the zooming, many arithmetic operations were involved to interpolate the position of visible images from the initial to the final position. These calculations were taking more time than rendering the visible portion of the canvas, which caused jagged animations. A successful solution for this problem was to pre-compute all the intermediate positions of each image and store them in a temporary structure. This way, at each zooming step, only the rendering overhead is considered which was fast enough to give a smooth transition. Other optimization techniques were used at several parts of the implementation to improve the performance of the interface.

The application was implemented in Microsoft Embedded Visual Studio 3.0 using MFC and Pocket PC SDK and consists of approximately 10,000 lines of code.

5. USABILITY STUDY

An experiment was conducted to compare Pocket PhotoMesa to Pocket ACDSee (www.acdsee.com). The split interface of ACDSee shown below is a traditional image browsing interface based on choosing a group of images (folder) to browse, and displays the images in fixed thumbnail size, scrollable interface. It does not use any animated transitions.

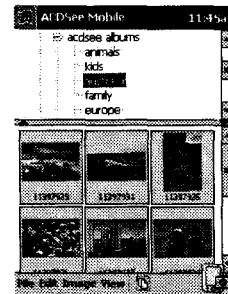


Figure 4: Pocket ACDSee (www.acdsee.com)

5.1 The Experiment

We designed an experiment to answer the following questions:

- Is it better to fit all images in one screen or in multiple screens with scrollbars?
- What is the best number of images displayed on a single screen so that thumbnails are still identifiable?
- Do Treemaps provide a better layout than a traditional layout providing equal size rectangles to every group?

Our hypothesis was that the use of an image browser that lays out all the images efficiently in one screen would enable users to quickly locate images of interest by visually identifying the themes in each image group and remembering the location of a previously visited image. Moreover, we thought that the use of animation in zooming will improve users' satisfaction while not having a significant effect on the time to locate images.

Our independent variable is the application used. For this variable, we have three treatments:

- Pocket PhotoMesa with animated zooming

- Pocket PhotoMesa with single step zooming
- Pocket ACDSee

Our dependent variables are:

- Objective: The time required to locate a specific image.
- Subjective: User satisfaction.

The experiment was run within subjects, where each used both Pocket PhotoMesa and Pocket ACDSee for the tasks (order of use was randomized between participants to insure balance). Each interface was used with a different set of images to cancel learning effects. To insure that users use visual identification of images in Pocket PhotoMesa, group labels were disabled. The entire experiment took approximately 30 minutes per participant. The experiment had 15 participants who are all computer science students, of which two are females. All participants were already familiar with pen based interaction and most of them own a Palm or Pocket PC.

After a training period with each interface, users were given a couple of minutes to navigate the interface and remember the location of the images. Users were then asked to locate 5 different images. These five images were given to the users via a written textual description, printed color version, and shown on the Pocket PC screen for 2 seconds. Target images were carefully chosen for each task to ensure a balance between tasks. Tasks ranged in difficulty from locating one of many existing images from description to locating a visually ambiguous image displayed for 2 seconds. By visually ambiguous we mean that the thumbnail of the image is visually similar to some other thumbnails.

We chose sets of 75 images categorized into 6 groups from the Corbis image library (www.corbis.com). The number of images was carefully chosen after running two pilot experiments with

100 images and the users' feedback was clear that identifying thumbnails at this small size was too difficult. A third pilot experiment was run with 75 images and users were able to visually identify most thumbnails on the screen.

5.2 Results

5.2.1 Quantitative results

The compilation of background surveys showed that the participants had mixed backgrounds. All the users had some background using pen based devices (most of them use Palms), mostly for appointments and contacts. None of the participants used it for image browsing.

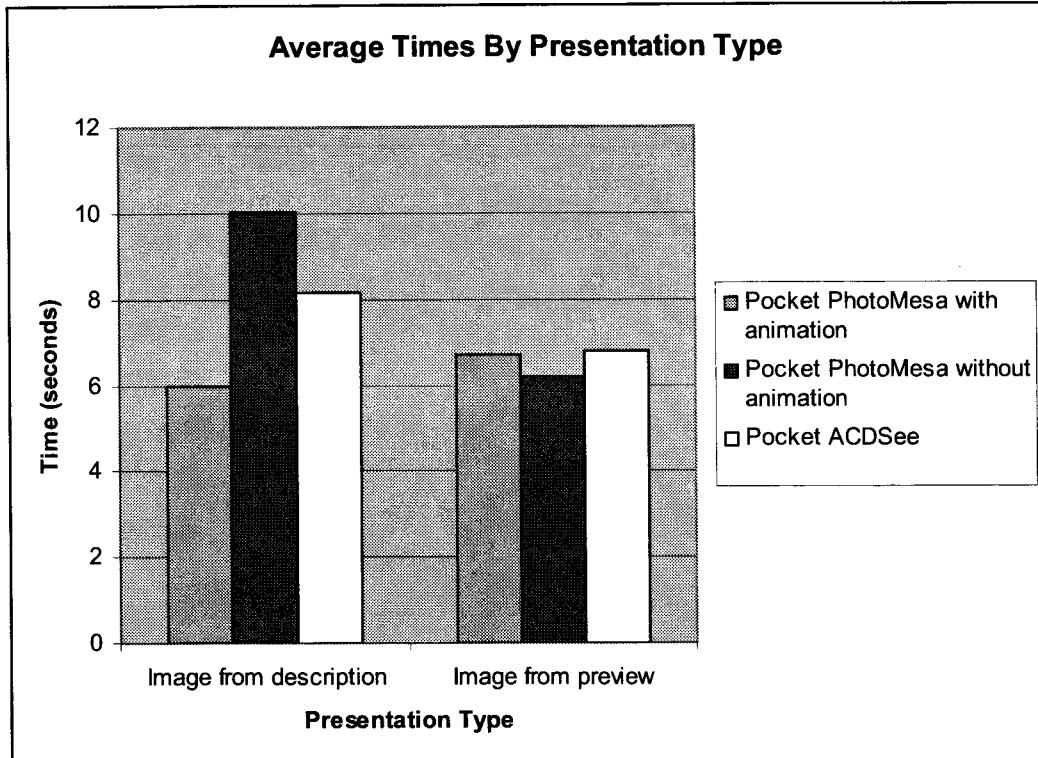
An analysis of the results shows that we got the best results using Pocket PhotoMesa without animated zooming when the image has been previously seen. Pocket PhotoMesa with animated zooming gives the best average time for locating images from a written textual description.

The total average time for locating an image on Pocket PhotoMesa with animation was 6.2 seconds, the time for the same interface without animation was 7.4 seconds and finally the average time using Pocket ACDSee was 6.4 seconds.

The previous analysis shows that there was no significant difference in time between Pocket PhotoMesa with animation and Pocket ACDSee, but interestingly, users' performance slowed down by over 1 second when animation was removed from Pocket PhotoMesa.

5.2.2 Subjective satisfaction results

On a scale of 1-9, Pocket PhotoMesa scored an average ease of use of 7.5 and interface enjoyment of 6.6 while ACDSee scored 6.2 and 4.8 respectively.



All users found that non-animated zooming was helpful for them to perform the given task (non-animated zooming satisfaction of 4.5/5) and screen layout was useful in arranging the images into equal sized thumbs wasting the least amount of screen real estate possible (layout satisfaction of 4.25/5). When it came to the use of animated zooming, users had mixed opinions: While most of them agreed that the animation increased slightly the time it takes to find the image, they mentioned that it helped them to maintain the context of navigation.

Participants also agreed that the biggest advantage of having all the image thumbnails laid out in one screen is to visually identify themes from colors and contrasts of thumbnails. During the experiment, we hid the group labels in Pocket PhotoMesa to ensure that the navigation would be based on visual grouping of themes and fast identification of thumbnails by targeting a candidate group theme that contains the image of interest. Users easily remembered the constant location of thumbnails provided by the Treemap layout and tend to zoom out to the original view and navigate to another image.

Users also agreed that when the image of interest does not have distinct color theme or contrast features, the task of identifying the image thumbnails visually became difficult at the level of detail showing all images. They also stated that the folder names in ACDSee helped them identify the themes but it was tricky during one of the tasks (e.g. to identify the image of a kangaroo, users using ACDSee went to inspect the folder named "animals" while the original image was in folder named "Australia"). To perform the same task on Pocket PhotoMesa, users were relying on their mental model of a kangaroo (light brown vertical shape, long tail and small head, usually exists in the desert) to visually identify thumbnails having mostly brown colors.

Three users said that the thumbnail sizes were large enough to identify average colors and patterns but were too small to

identify shapes if the objects don't occupy a good percentage of the image.

5.2.3 Observations

Our major observation for the Pocket PhotoMesa interface is that almost all the users tried to take advantage of the screen layout to reduce interaction steps whenever possible: When they were given a task, they usually start by scanning the interface looking for a specific color or intensity. They were also limiting their visual search to one or more groups that have promising color themes for the target image. If users did not find the target image in the first 5-10 seconds, they start navigating the groups using the stylus. Most users were able to identify most of the images without the need to navigate the interface.

We also observed that in Pocket PhotoMesa, it was easier to locate images from memory than to locate it by description. For example, browsing for a picture of a zebra, users were not thinking about locating an image of an animal, instead they were looking for an image having a pattern of white and black vertical stripes. When they were given a task to locate a chessboard, it took them more time since the board was slightly oriented and had an unusual color theme.

While the interaction (zooming in and out using stylus on different areas of the screen) was not intuitive, users learned to use it quickly and had no problems performing the given tasks. The rate of errors due to wrong interaction was negligible.

6. CONCLUSION

While the quantitative results did not show a significant time improvement in locating images using a zoomable user interface with Treemap layout, user satisfaction showed that the interface was easy and fun to use. We believe that the main use of the interface is exploring and navigating rather than locating images. A search option that finds images by name or by average dominant color will improve the time locating specific images. Since all the images fit in a single screen, thumbnail

size depends on the number of images. We believe that having more than 75 images on the pocket pc screen will increase the time to locate a specific image since some users already had difficulties working with the current thumbnail size. We have found that Pocket PCs are not usually used to hold a large number of images because of their storage limitation. Typical Pocket PC users hold pictures of their families, some memorable moments, or some portfolio work, hence 75 images is a good upper bound for such typical usage.

7. FUTURE WORK

Integration of more navigation controls and search features are on the top of our list for the next version of Pocket PhotoMesa.

8. REFERENCES

- [1] Bederson, B., Meyer, J., (1998). Implementing a Zooming User Interface: Experience Building Pad++, Software: Practice and Experience, 28 (10), pp. 1101-1135.
- [2] Good, L., and Bederson, B.B. (2002) Zoomable User Interfaces as a Medium for Slide Show Presentations, Information Visualization, pp. 35-49.
- [3] Johnson, B. and Shneiderman, B., Treemaps: A Space Filling approach to the visualization of hierarchical information structures, in Proc. IEEE Visualization '91, pp. 284-291, San Diego, IEEE Computer Society Press.
- [4] Bederson, B.B. et al.; Pad++: A zoomable graphical sketchpad for exploring alternate Interface physics. J. Vis. Lang. Comput.,7:3, Mar. 1996.
- [5] Herot, C.F., R. Carling, M. Friedell, and D. Kramlich. (1980). A Prototype Spatial Data Management System. Computer Graphics, 14(3), 63-70.
- [6] Suh, B., Ling, H., Bederson, B. B., & Jacobs, D. W. (2003). Automatic Thumbnail Cropping and Its Effectiveness. UIST 2003, ACM Symposium on User Interface Software and Technology, CHI Letters, 5(2), pp. 95-104.
- [7] Bederson, B. B., Meyer, J., and Good, L. (2000). Jazz: An Extensible Zoomable User Interface Graphics Toolkit in Java. UIST 2000, ACM Symposium on User Interface Software and Technology, CHI Letters, 2(2), pp. 171-180.
- [8] Bederson, B. B., Grosjean, J., & Meyer, J. (2004). Toolkit Design for Interactive Structured Graphics. IEEE Transactions on Software Engineering, 30(8).
- [9] Suh, B., & Bederson, B. B. (2002). OZONE: A Zoomable Interface for Navigating Ontology Information. In Proceedings of Advanced Visual Interfaces (AVI 2002) ACM Press, pp. 139-143.
- [10] Bederson, B. B. (2001). PhotoMesa: A Zoomable Image Browser Using Quantum Treemaps and Bubblemaps. UIST 2001, ACM Symposium on User Interface Software and Technology, CHI Letters, 3(2), pp. 71-80.
- [11] Combs, T. T. A., and Bederson, B. B. (1999). Does Zooming Improve Image Browsing? In Proceedings of Digital Library (DL 99) New York: ACM, pp. 130-137.
- [12] Wattenberg, M. (1999). Visualizing the Stock Market. In Proceedings of Extended Abstracts of Human Factors in Computing Systems (CHI 99) ACM Press, pp. 188-189.
- [13] Bruls, M., Huizing, K., and van Wijk, J. J. (2000). Squarified Treemaps. In Proceedings of Joint Eurographics and IEEE TCVG Symposium on Visualization (TCVG 2000) IEEE Press, pp. 33-42.
- [14] Wattenberg, M. Map of the Market (1998). <http://www.smartmoney.com/marketmap>.
- [15] Bederson, B. B. (2001). Quantum Treemaps and Bubblemaps for a Zoomable Image Browser. In Proceedings of User Interface and Software Technology (UIST 2001) ACM Press, (in press).
- [16] Bederson, B. B., Shneiderman, B., & Wattenberg, M. (2002). Ordered and Quantum Treemaps: Making Effective Use of 2D Space to Display Hierarchies. ACM Transactions on Graphics, 21(4), pp. 833-854.

Enabling fast and effortless customisation in accelerometer based gesture interaction

Jani Mäntyjärvi, Juha Kela, Panu Korpijää, and Sanna Kallio

VTT Electronics
P.O.Box 1100
FIN 90571 Oulu, Finland
Email: firstname.lastname@vtt.fi

ABSTRACT

Accelerometer based gesture control is proposed as a complementary interaction modality for handheld devices. Predetermined gesture commands or freely trainable by the user can be used for controlling functions also in other devices. To support versatility of gesture commands in various types of personal device applications gestures should be customisable, easy and quick to train. In this paper we experiment with a procedure for training/recognizing customised accelerometer based gestures with minimum amount of user effort in training. Discrete Hidden Markov Models (HMM) are applied. Recognition results are presented for an external device, a DVD player gesture commands. A procedure based on adding noise-distorted signal duplicates to training set is applied and it is shown to increase the recognition accuracy while decreasing user effort in training. For a set of eight gestures, each trained with two original gestures and with two Gaussian noise-distorted duplicates, the average recognition accuracy was 97%, and with two original gestures and with four noise-distorted duplicates, the average recognition accuracy was 98%, cross-validated from a total data set of 240 gestures. Use of procedure facilitates quick and effortless customisation in accelerometer based interaction.

General terms

Human computer interaction, input technology, mobile devices

Keywords

Gesture recognition, gesture control

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MUM 2004, October 27-29, 2004 College Park, Maryland, USA.
Copyright 2004 ACM 1-58113- 981-0 /04/10... \$5.00"

1. INTRODUCTION

Gestures are an intuitive communication channel, which has not yet been fully utilised in human-computer interaction. Mobile devices, such as, PDA's or mobile phones, and wearable computers provide new possibilities for interacting with various applications, but also introduce new challenges with I/O devices, e.g., small displays and miniature input devices. Gesture input devices (containing accelerometers for detecting movements) could be integrated to clothing, wristwatches, jewellery or mobile terminals, for providing a means for interacting with various of devices and applications located in surroundings. These input devices could be used to control, e.g. home appliances, with simple user definable hand movements.

In related work we focus on movement sensor based approaches, which utilise different kinds of sensors, e.g. tilt, acceleration, pressure, conductivity, capacitance, etc. to measure movement. An example of an implementation is GestureWrist, a wristwatch-type gesture recognition device using both capacitance and acceleration sensors to detect simple hand and finger gestures [11]. Accelerometer based gesture recognition is used for example in a musical performance control and conducting system [12], and a glove-based system for recognition of a subset of German sign language [3]. In wearable interface called Ubi-finger acceleration, touch and bend sensors are used to detect fixed set of hand gestures, and infrared LED for pointing a device to be controlled [14]. Yet another gesture based interaction device is XWand [15]. It utilises both sensor based and camera based technologies capable of detecting the orientation of the device using a 2-axis accelerometer, a 3-axis magnetometer and a 1-axis gyroscope, and position and pointing direction using two cameras. The user can select a known target device from the environment by pointing, and control with speech and a fixed set of simple gestures.

Gesture recognition has been also studied for implicit control of functions in cell phones, e.g. for answering and terminating a call, without user having to explicitly perform the control function [1], [7]. In implicit gesture control, the main problem is that users usually perform the gesture very differently in different cases, e.g. picking up a phone can be done in very many different ways depending on the situation. Explicit control directly presupposes that gestures trained for performing certain functions are always repeated as they were trained.

As a complementary interaction modality, acceleration based gesture command interface is quite recent, and many research problems still require solving. The topic is very wide in scope, since there are a very large number of possible suitable gestures

for certain tasks, as well as many tasks that could potentially be performed using gestures. Obviously, the recognition accuracy for detecting the gesture commands should be high. Nearly 100% accuracy is required for user satisfaction, since too many mistakes cause the users to abandon the method. Moreover, in some applications the control commands need to be trained by the user. If training is too laborious, it may again cause users to abandon the interaction method. Therefore, the training process should be as effortless and quick as possible.

As a sensing device, SoapBox (*Sensing, Operating and Activating Peripheral Box*) is utilised in this work. It is a sensor device developed for research activities in ubiquitous computing, context awareness, multi-modal and remote user interfaces, and low power radio protocols [13]. It is a light, matchbox-sized device with a processor, a versatile set of sensors, and wireless and wired data communications. Because of its small size, wireless communication and battery-powered operation, SoapBox is easy to install into different places, including moving objects. The basic sensor board of SoapBox includes a three-axis acceleration sensor and other sensors for monitoring environment [13].

Various statistical and machine learning methods can potentially be utilised for training and recognising gestures [1], [7]. This study applies discrete Hidden Markov Models (HMM), a well-known method, e.g., from speech recognition [10]. HMM is widely used in speech and hand-written character recognition as well as in gesture recognition in video-based and glove-based systems.

In our previous work we have found that people usually prefer defining their own gestures, concluding that gesture control should be customisable. Moreover, it should be customisable which functions are performed with gestures [6]. This result leads to the important problem, which is the ability to train and recognise free form gesture commands. The users should be able to carry out training of personal gestures with as few repetitions as possible, since making many repetitions can be a nuisance. The novelty and main contribution of this paper is that the amount of required repetitions performed by a user, thus user's effort in training can be decreased, when discrete HMMs are applied. According to the results, this is feasible based on a procedure where noise-distorted signal duplicates are used in training. It is shown that with the procedure applied the amount of required repetitions is decreased while good recognition accuracy is maintained.

Organisation of the paper is the following. Gesture interface basic concepts are first defined and categorised. Methods for gesture training and recognition and for decreasing amount of training repetitions are presented. The experiments and results are provided. The experiment is based on a DVD gesture control prototype, which demonstrates the practical feasibility of gesture recognition. Finally, discussion and suggestions for future work are given together with conclusions.

2. GESTURE CONTROL

The main purpose of this paper is to present a procedure for decreasing user effort in customising accelerometer based gesture control when discrete HMMs are applied. As discussed in previous section, accelerometers are utilised in implementing various types of interfaces. To clarify the differences between various approaches we clarify the types of

movement sensor based user interfaces by the categorisation, Table 1.

Table 1: Categorisation and properties of movement sensor based user interfaces

Interface type	Operating principle	Customisation	Complexity
1. Measure & control	Direct measurement of tilting, rotation, or amplitude	-	Very low
2. Discrete gesture command	Gesture recognition	Machine learning, freely customisable	High
3. Continuous gesture command	Continuous gesture recognition	Machine learning, freely customisable	Very high

Direct measurement and control systems are not considered as gesture recognition systems since in their operating principle, measurement of tilt, rotation or amplitude is mapped directly to control. In this paper, we refer gestures to as user hand movements collected with a set of sensors in a handheld device. Hand movements are modelled by machine learning methods in such a way that any movement performed can be trained for later online recognition. Furthermore, a gesture based device control command is executed based on hand movement recognised. In discrete gesture command start and stop of a gesture is defined, e.g., with a button while in continuous gesture command recognition of gestures is carried out online from a flow of hand movements. Handheld devices with gesture recognition enable the control of applications located in a handheld device or in external devices in vicinity.

Concerning sensor based hand movement interfaces the paper hence focuses on discrete gesture command interfaces (second category in table 1). There exist multitudes of simple measure & control applications e.g. tilt and rotation based actions. We consider them to belong to the category one in table 1.

3 GESTURE TRAINING AND RECOGNITION

According to our previous study, users prefer intuitive user definable gestures for gesture-based interaction. This is a challenge for gesture recognition and training system, since both online training and recognition are required. To make the usage of the system comfortable, low number of repetitions of a gesture is required during the training. On the other hand, a good generalisation performance in recognition must be achieved while maintaining good recognition accuracy. Other requirements include; a recogniser must maintain models of several gestures, and when a gesture is performed, training or recognition operations must not take a long time by a system. This section presents methods used in online gesture recognition in our prototypes.

In accelerometer based gesture interaction, sensors produce signal patterns typical for gestures. These signal patterns are used in generating models that allow the recognition of distinct gestures. We have used discrete Hidden Markov Models (HMM) in recognizing gestures. Main motivation for choosing HMM for our purposes is that the method a modelling tool that

can be applied for modelling time-series with spatial and temporal variability. The HMM has also been utilised in other experiments for gesture and speech recognition [7], [11]. Acceleration sensor based gesture recognition using HMM has been studied for example in [3], [4], [7].

The recognition system works in two phases: training and recognition. A schematic block diagram of the system is presented in figure 1. Common steps for these phases are signal sampling from three accelerometers to 3D sample signals, preprocessing, and vector quantisation of signals. Repeating the same gesture produces variation of measured signals, because the tempo and the scale of the gesture can change. In preprocessing, data from gestures is first normalised to equal length and amplitude. The data is then submitted to a vector quantiser. The purpose of the vector quantiser is to reduce the dimensionality of the preprocessed data to one-dimensional sequences of discrete symbols that are used as inputs for the HMMs in training and in recognition. One vector quantiser is designed using an extensive set of gesture data. The vector quantiser is used to quantise all data in our experiments. Our procedure for gesture training/recognition includes adding various types of noise to data. The procedure is explained in more detail in next section.

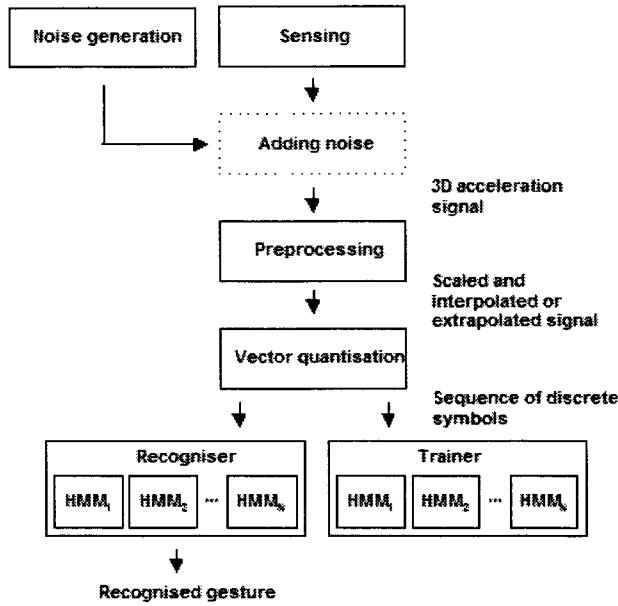


Figure 1: Block diagram of a gesture recognition/ training system

3.1 Preprocessing

The preprocessing stage consists of interpolation or extrapolation and scaling. Gesture data is first linearly interpolated or extrapolated if data sequence is too short or too long, respectively. Then amplitude of data is scaled using linear min-max scaling. The same parameters are used both in the training and in the recognition phase.

3.2 Vector quantisation

The vector quantisation is used to convert preprocessed three dimensional data into one dimensional prototype vectors. The

collection of the prototype vectors is called a codebook. In our experiments the size of the codebook is selected empirically to be 8. Vector quantisation is done using k-means algorithm [2].

3.3 HMM

Hidden Markov Model is a stochastic signal modeling method. HMM is an extension of Markov process. The output of Markov model is the set of states at each instant of time, where each state corresponds to a physical, observable event. The HMM includes the case where the observation is a probabilistic function of the state. Hence, the HMM is a double stochastic process with an underlying stochastic processes that is not observable, but can be observed through another set of stochastic processes that produce the sequence of observation. Formally a HMM can be expressed as

$$\lambda = (\mathbf{A}, \mathbf{B}, \pi) \quad (1),$$

where \mathbf{A} denotes the state transition probability matrix, \mathbf{B} is the observation symbol probability matrix and π is the initial state probability vector. The specification of the discrete HMM involves a choosing number of states, a number of discrete symbols, and definition of the three probability densities with matrix \mathbf{A} , \mathbf{B} , and π .

Usually, three basic problems must be solved for the real applications: the classification, the decoding and the training. In this study only classification and training are relevant and are solved using Viterbi and Baum-Welch algorithms, respectively. The global structure of the HMM recognition system is composed of parallel connection of each trained HMM ($\lambda_1, \lambda_2, \dots, \lambda_M$), where λ_i indicates a trained HMM model for each gesture and M is a number of gestures [16]. Hence adding a new HMM or deleting the existing one is feasible. The recognition of the given unknown gesture is performed by finding an index of the discrete HMM which produces the maximum probability of the observation symbol sequences.

In this paper an ergodic topology, was utilised. Left-to-right model is a special case of ergodic model and often preferred when modelling time-series whose properties sequentially change over time. However, in the case of gesture recognition from acceleration signals, both ergodic and left-to-right models have been reported to give similar results [3]. The alphabet size, which is the codebook size, used here is 8 and number of states in each model is 5. In [3] it has been concluded that an ergodic model with five state works best for gesture set they use. It has been reported that the number of states does not have significant effect on gesture recognition results [8].

3.4 Decreasing user effort in training

The user experience in accelerometer based gesture interaction should be as positive as possible and making several training repetitions can be a nuisance. Thus, approach for trying to decrease the amount of training repetitions is well justified. It has been shown that adding noise increases detectability in decision making in certain conditions [5]. In this paper, we apply the idea of adding noise to examine whether the idea can be used in decreasing the amount of training repetitions done by the user when discrete HMM:s are applied for the training and recognition task.

The approach is to generate new training data, the three dimensional noise-distorted gesture signal duplicates $\mathbf{x}_i + \mathbf{n}_i$, by copying the original gesture data vector \mathbf{x}_i and adding random noise vector \mathbf{n}_i into the copy. We consider two random noise distributions in our experiments:

- Uniform distribution. Random samples are generated between $[-a, +a]$, mean $\mu = 0$, variance $\sigma^2 = a^2/3$.
- Gaussian distribution. Random samples are generated from normal distribution, mean $\mu = 0$, variance σ^2 .

Various signal to noise ratios (SNR) are experimented with. SNR is determined as ratio of signal variance to noise variance.

4 PROTOTYPE

We have implemented a prototype based on wireless handheld sensor box, SoapBox, and PC software for practical examination of gesture based interaction, figure 2.

SoapBox acceleration sensors (ADXL202) measure both dynamic acceleration (e.g. motion of the box) and static acceleration (e.g. tilt of the box). The acceleration is measured in three dimensions and sampled at a rate of 46 Hz. The measured signal values are wirelessly transmitted from the remote SoapBox to a central SoapBox that is connected to a Windows PC with a serial connection. The gesture start and end are marked by pushing the button on the SoapBox at the start of the gesture and releasing it at the end, which then activates either training or recognition algorithm. All signal processing and pattern recognition software runs in a PC. Recognition results can be mapped to different control commands and transmitted to the control target using e.g. infrared control signals or TCP/IP socket communication. The mapping between gestures and output functions is done in the training phase by naming the gestures using specific command names, e.g. *DVD Play*, for each gesture.

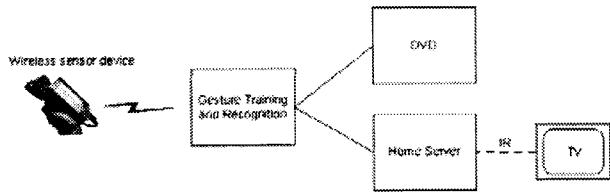


Figure 2: Conceptual overview of DVD player prototype

4.1 Gesture control of DVD player

Gesture recogniser is trained with eight popular gestures and used to control basic functions of DVD player. The gesture control is provided as an alternative control method to the existing remote controller. Table 2 presents the control functions and corresponding gestures used in DVD control prototype.

Table 2: Gestures used in DVD prototype. Origin of the gesture is presented as a dot and arrow indicates the trajectory of the gesture. X,y-co-ordinates indicate that gesture is drawn in the air in x,y-plane in front of the user.

Play	Stop	Next	Previous
Increase	Decrease	Fast forward	Fast rewind

The recognition results are mapped to infrared remote control signals and sent to DVD player by means of IR transmitter. This prototype utilizes only discrete gesture commands.

5 EXPERIMENTS AND RESULTS

5.1 Experiments

Eight gestures were selected to control DVD player. Recognition capability of HMM based gesture recognition system was tested using gestures in table 2. For each gesture, 30 distinct three-dimensional acceleration vectors were collected from one person, and thus the total test data set consisted of 240 repetitions. Length of the three-dimensional acceleration vectors varied depending on the duration of the gesture, mean length l of gestures was 25 samples $l \in [13,45]$, with $\sigma = 9.85$. Figure 3 illustrates three-dimensional acceleration vector for gesture *Play*.

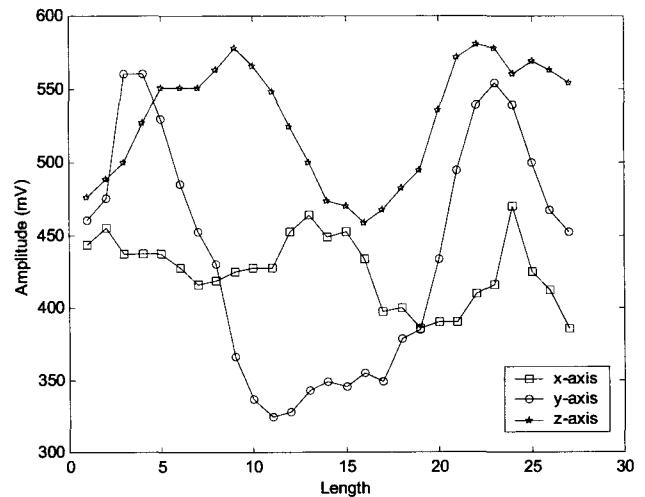


Figure 3: Three-dimensional acceleration vector for gesture *Play*

Each three-dimensional vector was either interpolated or extrapolated to a length of 40 samples. Thereafter, vector quantizer was used to map three-dimensional vectors into one-dimensional sequence of codebook indices. The codebook was

generated from collected gesture vectors using k-means algorithm. After the vector quantization, the gesture was used either to train HMM or to evaluate recognition capability of HMM.

The experiments include the following tests (printed with Italics):

Examination of the optimal threshold value to determine the convergence of the HMM. Training of the HMM is an iterative process, which will continue until convergence. Here, the convergence of the HMM was determined using threshold value and following form

$$\frac{|f(t) - f(t-1)|}{avg} < \text{threshold} \quad (2)$$

where

$$avg = \frac{(|f(t)| + |f(t-1)|)}{2} \quad (3)$$

and $f(t)$ is log-likelihood of the HMM at iteration t . When testing threshold value, the number of training vectors was kept in 5.

Examination of the recognition accuracy using different amount of training repetitions. In this test, the recognition rate for each gesture is first calculated by using 2 data vectors for training and the remaining 28 for recognition. The recognition accuracy for each gesture is the result of cross validation, so that in the case of 2 data vectors, there are 15 training sets, and the rest of the data (28) is used as the test set 15 times. The procedure is repeated for each gesture, and the result is averaged over all the 8 gestures. This procedure is then repeated with 4, 6, 8, 10 and 12 data vectors for training to find out the recognition accuracy when only original data is used and how many training vectors is required for reaching a proper accuracy.

Examination of noise SNR levels to find a noise level leading to best results. In this test, two actual gestures are used for training, plus two copy gestures, in total four. Tests are carried out with Gaussian and uniformly distributed noise. Also in this test the accuracies are the result of cross-validation.

Examination of the effect of using noise distorted signal duplicates in training. In this test, we study the procedure for decreasing the required user effort in training by copying the original gesture data and adding noise into the copy. The noise distorted copy is then used as training data. Experiments are carried out with varying number of original + noise distorted signal duplicates. Also in this test the accuracies are the result of cross-validation.

5.2 Recognition results and discussion

With user definable gestures, the training has to be done by the user. This means that training situation should be as easy and as quick as possible. Thus, it is important to keep the number of repetitions required from user.

Examination of the optimal threshold value to determine the convergence of the HMM. When testing threshold value, the number of training vectors was kept in 5. Figure 4 shows the recognition rate for different threshold values. There is some variance between gestures, but on average the best result was achieved with threshold value $1,0e^{-3}$. This value keeps the

number of training iterations between 10-30, depending on gesture. With additional training iterations models do not seem to learn more, but instead overfit and recognition capability suffers.

Examination of the recognition accuracy using different amount of training repetitions. Effect of the number of the training vectors for recognition rate is shown in figure 5. Recognition accuracy over 90% is achieved already with four training vector while one and two training vectors reach to accuracies below 90%. However, it can be seen that recognition results get better as the number of training vectors increases. With six original training vectors the accuracy is over 95%. But training eight gestures, for example, with six training repetitions requires already the user to make 48 repetitions which is obviously a nuisance.

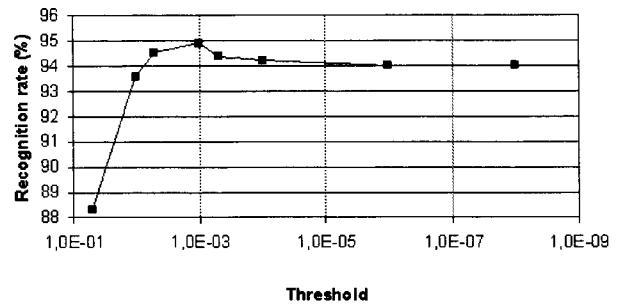


Figure 4: Recognition accuracies for various threshold values

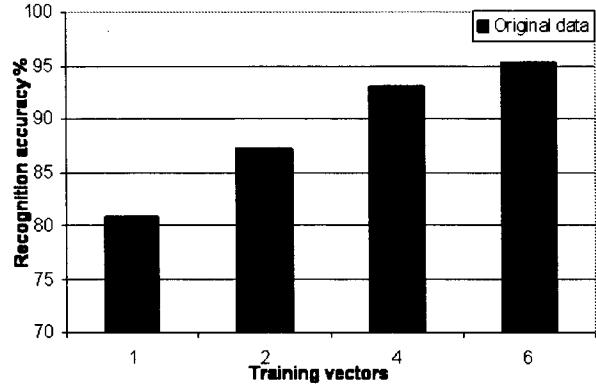


Figure 5: Recognition accuracies for various number of training vectors (original data)

Examination of noise SNR levels to find a noise level leading to best results. Recognition accuracies with various SNRs for Gaussian and uniformly distributed noise are calculated using two original and two noise distorted signal duplicates (2+2) as training vectors. Recognition accuracies with various SNRs are shown in figure 6. It shows that best accuracy 97,2% is obtained with Gaussian distributed noise with SNR = 3 while best accuracy with uniformly distributed noise 96,3% is obtained with SNR = 5. This suggests that it is reasonable to use those relative noise levels in further experiments.

Examination of the effect of using noise distorted signal duplicates in training. The recognition accuracies for various number of noise distorted training vectors for uniformly and Gaussian distributed noise are shown in figure 7. It shows that with both types of noise recognition accuracy over 96% can be

achieved with one original and three noisy copies (1+3) with both types of noise. However, performance with Gaussian noise is slightly better; almost 98% accuracies with two original and two/four noisy copies (2+2, 2+4).

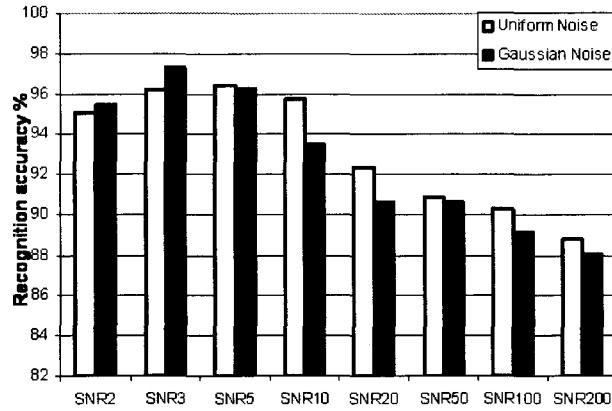


Figure 6: Recognition accuracies for various relative noise levels for Uniform and Gaussian distributed noise.

Compared to the situation where HMMs are trained using only one original training vector (Fig. 5), the gain achieved by adding one noisy copy (1+1) to training set is over 11% with both noise types, and the gain achieved by adding three noisy copies (1+3) to training set is over 15% with both noise types. Furthermore, compared to the situation where two original training vectors are used in training (Fig. 5), the gain achieved by using two originals and noisy signal duplicates is at least ~11% percent when Gaussian noise is used. With uniformly distributed noise results are also good. It must be noted that accuracies obtained by using noisy copies with 1 to three original data vectors (1+3, 2+2, 2+4, etc.) are all better than accuracy obtained using six original training vectors (Fig. 5). Adding the noisy vectors to the original training set improves natural variation of the gesture and it becomes better captured, and thus the new training set is more representative sample of the vectors describing the gesture.

These results show that good accelerometer based gesture recognition accuracies can be achieved using noise distorted signal duplicates and low amount of original data vectors in training. This decreases user effort considerably because one or two training repetitions of a gesture instead of six is adequate for reaching proper functionality of gesture interaction with this dataset and discrete HMMs.

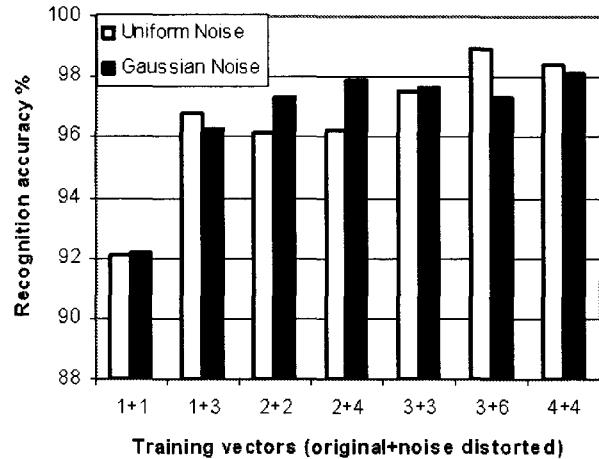


Figure 7: Recognition accuracy versus various number of noise distorted training vectors for Uniform and Gaussian distributed noise.

6 CONCLUSIONS AND FUTURE WORK

An approach for enhancing customisation in accelerometer based gesture interaction was presented. Experiments were conducted to evaluate a procedure for training/recognizing accelerometer based gestures with minimum amount of user effort in training. Discrete Hidden Markov Models were applied. Recognition results were presented for a DVD player gesture commands. A procedure based on adding noise-distorted gesture signal duplicates to training set was applied and it was shown to increase the recognition accuracy while decreasing the user effort in training. For a set of eight gestures, each trained with two original gestures and with two Gaussian noise-distorted duplicates, the average recognition accuracy was 97%, and with two original gestures and with four noise-distorted duplicates, the average recognition accuracy was 98%, cross-validated from a total data set of 240 gestures. Use of the procedure facilitates quick and effortless customisation in accelerometer based interaction. For some tasks, gesture control can be natural and quick. However, many targets of future work remain. The results of user dependent recognition should be extended to user independent recognition, as is the goal in speech recognition. Other sensors, e.g., gyroscopes, should be studied. Furthermore, the feedback of the gesture interface, e.g., by means such as vibration or audio should be studied.

ACKNOWLEDGEMENTS

We gratefully acknowledge the support of our partners in the Ambience project and Tekes for the funding.

REFERENCES

- [1] Flanagan J, Mäntylä J, Korpiaho K, Tikanmäki J (2002). Recognizing movements of a handheld device using symbolic representation and coding of sensor signals. Proceedings of the First Intl. Conference on Mobile and Ubiquitous Multimedia, pp. 104-112.

- [2] Gersho A, Gray R.M (1991). Vector Quantization and Signal Compression. Kluwer.
- [3] Hoffman F, Heyer P, Hommel G (1997). Velocity Profile Based Recognition of Dynamic Gestures with Discrete Hidden Markov Models. Proceedings of Gesture Workshop '97, Springer Verlag.
- [4] Kallio S, Kela J, Mäntyjärvi J (2003). Online Gesture Recognition System for Mobile Interaction. IEEE International Conference on Systems, Man & Cybernetics, Volume 3, Washington D.C. USA pp 2070-2076.
- [5] Kay S. (2000) Can detectability be improved by adding noise? IEEE Signal Processing letters, Vol 7 No 1. pp. 8-10.
- [6] Korppiä, P., Häkkilä, J., Kela, J., Ronkainen, S., Känsälä, I. Utilising Context Ontology in Mobile Device Application Personalisation. To appear in proc. International Conference on Mobile and Ubiquitous Multimedia. 2004.
- [7] Mäntylä V-M, Mäntyjärvi J, Seppänen T, Tuulari E (2000). Hand Gesture Recognition of a Mobile Device User. Proceedings of the International IEEE Conference on Multimedia and Expo, pp. 281-284.
- [8] Mäntylä V-M (2001). Discrete Hidden Markov Models with Application to Isolated User-Dependent Hand Gesture Recognition. VTT Publications.
- [9] Peltola J, Plomp J, Seppänen T (1999). A Dictionary-adaptive Speech Driven User Interface for Distributed Multimedia Platform. Euromicro Workshop on Multimedia and Telecommunications, Milan, Italy.
- [10] Rabiner L (1998). Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. Proceedings of the IEEE, Vol. 77, No. 2.
- [11] Rekimoto J (2001). GestureWrist and GesturePad: Unobtrusive Wearable Interaction Devices. Proceedings of the Fifth International Symposium on Wearable Computers, ISWC 2001.
- [12] Sawada H, Hashimoto S. (2000). Gesture Recognition Using an Accelerometer Sensor and Its Application to Musical Performance Control. Electronics and Communications in Japan Part 3, pp 9-17.
- [13] Tuulari E, Ylisaukko-oja A (2002). SoapBox: A Platform for Ubiquitous Computing Research and Applications. First International Conference, Pervasive 2002, pp. 26-28.
- [14] Tsukada K, Yasumura M (2002). Ubi-Finger: Gesture Input Device for Mobile Use. Proceedings of APCHI 2002, Vol. 1, pp 388-400.
- [15] Wilson A, Shafer S (2003). Between u and i: XWand: UI for intelligent spaces. Proceedings of the conference on Human factors in computing systems, CHI 2003, April 2003. pp 545-552.
- [16] Yoon H.S (2001). Hand Gesture Recognition Using Combined Features of Location, Angle and Velocity. Pattern Recognition 34, pp 1491-1501.

The Road Rager - Making Use of Traffic Encounters in a Mobile Multiplayer Game

Liselott Brunnberg

Mobility Studio, The Interactive Institute

Box 24081

SE-104 50 Stockholm

liselott.brunnberg@tii.se

ABSTRACT

We present Road Rager, a prototype built in order to explore our hypothesis that proximity and a possibility to identify other players during temporary encounters could spur social interaction and enhance a mobile gaming experience. In this case, it is a multiplayer game designed to enable passengers in different cars to play against each other during a meeting in traffic. Using such meetings as resource opens new interesting possibilities for novel and engaging mobile experiences. In this paper we present the game concept, the implementation and the possibilities to interact - designed to successfully benefit from the dynamic and vivid mobile context created during a traffic encounter. We also present a technical test and some initial user feedback on the gaming experience.

Categories and Subject Descriptors

H.5.1 [Information Interfaces and Presentation]: Multimedia Information Systems - Artificial, augmented, and virtual realities; H.5.2 [Information Interfaces and Presentation]: User interfaces - Prototyping; K.8 [Personal Computing]: Games.

General Terms

Design, Experimentation, Human Factors

Keywords

Proximity-based mobile games, interaction design, traffic encounters, social interaction

1. INTRODUCTION

Future mobile technology will provide more services that exploit the benefits of mobile life [5]. Current mobile games are often portable versions of classic computer games [12]. There is also the possibility of incorporating different aspects of mobility to create immersive experiences. We suggest that a mobile game could become compelling in a new way, if it is aware of the vivid and dynamic mobile context. Travelling along a road means a continuous flow of impressions and new situations where changing scenes, sense of motion and contingent encounters provide for a very special experience. It can be seen as a sequential experience, resembling a dramatic play of space and

"Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee."

MUM 2004, October 27-29, 2004 College Park, Maryland, USA.
Copyright 2004 ACM 1-58113- 981-0 /04/10... \$5.00"

motion, also called the highway experience. Contingent traffic encounters such as rapid meetings, protracted overtaking or gatherings i.e. traffic jams or red light accumulations constitute an essential part of the experience of travelling along a road [1]. We explore how these meetings, the motion of the accompanying traffic, can be used to create a fun and compelling mobile game and how it can add to the gaming experience. Our hypothesis is that proximity and a possibility to identify other players could spur social interaction and enhance the experience. The target group is children who travel in the back seats of cars.

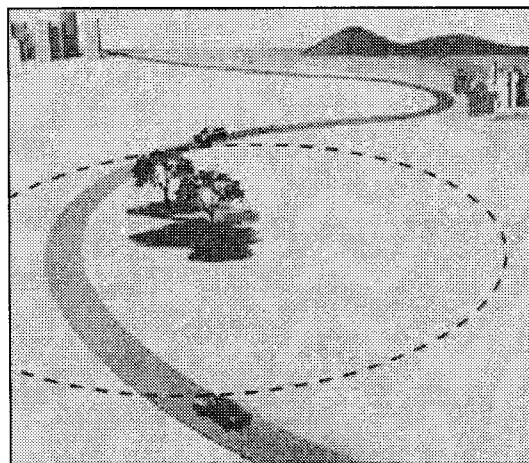


Figure 1. We explore how contingent traffic encounters can add to the gaming experience.

A game prototype, i.e. *Road Rager*, was created. *Road Rager* uses wireless ad hoc networking technology to enable game-play between car passengers as they convene within a limited range. Due to high relative speed an encounter can be extremely momentary, sometimes not longer than a couple of seconds. Consequently, a central design challenge concerns the possibility to enable and balance the player's engagement between virtual and real when the time for identification and interaction with the opponent player is very brief. However, drawing on a screen based interface risks having the player focusing on the screen rather than looking out through the window. This inspired us to explore the interaction in terms of a tangible interface. The fictitious connection between the game world and occurring encounters was achieved by means of direction and distance to the

opponent player. Additionally, it was important to recognize that traffic encounters occur in a variety of ways, this imply that different kinds of encounters call for different possibilities to interact. When designing the game we chose to focus on three different encounters, i.e. meeting in opposite lanes, overtaking and traffic-light accumulations. Furthermore, the game is designed in such way that it often is rewarding for the player to identify the kind of encounter taking place, in this way we further stimulate the player to engage with the surrounding physical world.

The paper is outlined as follows; we start by presenting a brief overview of traffic encounters and the idea of using them as resource in a game. We then move on to present the concept and the possibilities to interact within the game. Section four gives a discussion on how the game is designed in order to map to different traffic encounters. Further, we present the implementation and a small technical test in order to gain insight into its feasibility. Section seven gives a summary of initial user feedback from a field trial. Finally, we give a brief account for related research.

2. COMBINING MOBILE GAMING WITH TRAFFIC ENCOUNTERS

Any road user's journey often coincides with several other journeys. Traffic encounters arise when two or more people on the roads are co-located and are within visible sight of each other e.g. in intersections, passing in opposite lanes or when overtaking [11]. Encounters with other road-user can occur in many different ways. Due to high relative speed an encounter can be extremely momentary, others more persistent.

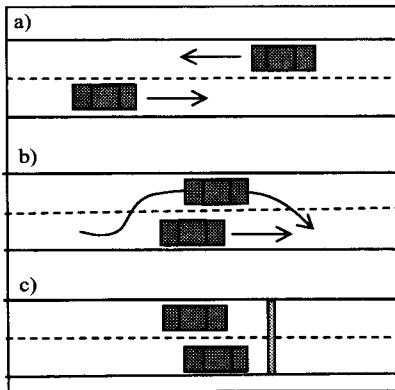


Figure 2. a) Meeting in opposite lanes, b) overtaking, c) traffic-light accumulations

When designing the game we have focused on three different types of encounters, i.e. meeting of two vehicles travelling in opposite lanes, overtaking and traffic-light accumulations (fig. 2). These encounters were chosen because we believe that they constitute short gaming events but bring about different challenges regarding interpretation, exploration and manipulation for the game-play [8]. Encounters where two vehicles travel in opposite direction generally last for a very short period of time, often not longer than a couple of seconds. Overtaking often mean a more protracted co-location than a meeting but contribute to the

disadvantage of having another player behind the back during parts of the encounter. Traffic-light accumulation characterise a situation where the players are standing still for a short period of time in close proximity of each other.

3. ROAD RAGER

Using temporal and unpredictable encounters as resource requires a game-design that takes into account sudden and unpredictable appearance and interruptions between players. A hypothesis is that the possibility to identify other players can enhance the gaming experience and spur social interaction. This motivated several design criteria:

- The game should be designed to support the fictitious connection between the game world and the physical world.
- It should support identification, awareness and social interaction between players.
- It should take different situations into account, i.e. it should recognize that different kinds of encounters call for different possibilities to interact.
- It should cultivate the player's fantasy and imagination.

With these design criteria in mind we will in this section present the *game* concept and the ability to interact within the game.

3.1. Game Concept

The game *Road Rager* consists of a framing story, a set of game level stories and of manipulative events automatically taking place when players are in the proximity of each other. The framing story is told when the game starts to provide the player with the story as well as an understanding of the rules and goals of the game. Game level stories are told in between manipulative events with the purpose of cultivating the fantasy of the game-play. When the game begins the player takes on the role as a character with magic powers. The player's goal is to gain as high power as possible before getting to the big yearly meeting for witches and warlocks. High power can be gained both by achieving knowledge, such as new spells or by gathering powerful objects by being the most powerful in battles. The implementation of the game is currently restricted to game-play between only two persons during a manipulative event. When two players are within wireless reach the game initiates a duel with the purpose of enchanting the opponent. The manipulative event ends if one player gets enchanted or if they get out of each other's wireless reach. If the opponent gets enchanted the player can trade objects and knowledge in possession for more powerful ones. If the connection is broken before any of the players gets enchanted they will receive objects and knowledge dependent on the result of the game-play.

3.2. The Interaction within the Game

In order to preserve the connection with the physical world during brief meetings it is essential that the player during these events can focus outside the window of the car rather than on a screen. We have partly used a tangible interface to directly link the digital and the physical world and provide a seamless method of allowing natural physical and social interaction between people [10]. In swift meetings, when the period of time for interaction with other players is limited, the player can concentrate on spotting the other player and act instantly without looking at a screen.



Figure 3. The Clutcher, a PDA and a Bluetooth GPS

The tangible interface is realized as a magic gadget, i.e. the Clutcher, equipped with fourteen LED's and a button. The LED's communicate certain information relevant for the game-play. Four of these, so-called locator LED's, inform the player about the direction to the opponent player (fig. 4). Ten smaller LED's, so-called power bars, are placed in two rows and are sequentially turned on and off to indicate the amount of magic power the players possess. One of the rows indicates the player's own power and the other the opponent's. The button is for changing tool (see section 3.3).

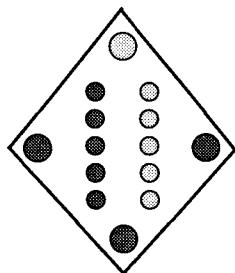


Figure 4. Feedback LEDs located on top of the Clutcher

To further encourage the player to interact directly with the physical world we use sounds as feedback on the interaction. We also use it as a two-sided feedback, meaning that both players will hear audio feedback as a result of an action. The purpose is to increase the awareness and feeling of presence of the other player and to encourage social interaction.

At the same time as the real world can provide for a rich space where the game can take place it is also important to cultivate the fantasy and imagination of the game and to provide the player with proper feedback and interpretation of the game-play. Therefore we have chosen to use the screen of a PDA as interface in between different manipulative events. The player can then view animated stories related to the game play, the identity of an encountered character, as well as results and status.

3.3. Virtual Tools

The interaction during manipulative events relates to the traffic encounters in terms of direction and distance to opponent player. These design parameters are varied to enable the Clutcher to be turned into any of three different virtual tools, i.e. an *Electro squeezer*, a *Sludge thrower* and a *Magic wand*, and are designed to be more or less suitable for the traffic encounters previously discussed (fig. 2).

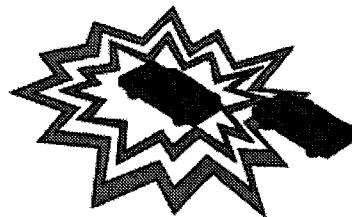


Figure 5. Sending out electronic pulses with the Electro squeezer

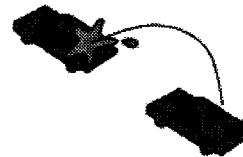


Figure 6. Throwing sludge with the Sludge thrower



Figure 7. Cast a spell with the Magic wand

The tool that demands least understanding of the opponent's physical location is the Electro squeezer, based on neither aiming direction nor distance. This tool can be used in a battle without knowing anything about the location or direction to the opponent player, as long as being within wireless reach. It sends out electric pulses and is fired by squeezing the Clutcher (fig. 5). The Sludge thrower is based on aiming direction which makes it more dependent on an understanding of the opponent's physical location than the Electro squeezer. When using the Sludge thrower the locator LED's are active and indicates, if lightened, the direction to the opponent. With this tool the player can throw magic sludge and is used in the same fashion as if throwing something, i.e. the player has to move the Clutcher forward/downwards at the same time as aiming it towards the opponent (fig. 6). The player will hear a sound indicating that something is flying through the air for two seconds and then a sound indicating

hit or miss. The Magic wand is the tool that demands most understanding of the opponent's physical location, being based on both aiming direction and distance. The Magic wand can be used to cast spells (fig. 7). To do this the wand should be swung to follow a circular pattern, but it can only be used once during an encounter. Similar to the Sludge thrower it shows the direction to the opponent player with the help of the locator LED's. It makes use of distance in the way that the closer the player is to the opponent player the more powerful is the tool.

- *The Electro squeezer:* No demand of aiming or identification
- *The Sludge thrower:* Aiming but not identification needed
- *The Magic wand:* Aiming and identification needed

4. MAPPING GAME MANIPULATION TO TRAFFIC ENCOUNTERS

The tools and the scoring are mapped to the type of traffic encounter accordingly. The Electro squeezer is quicker and easier to use than the other two tools and require no understanding of direction or identification of opponent. Consequently, the Electro squeezer is suitable for encounters that last for a very short period of time when the interaction time is very limited, such as in sudden meetings in opposite lanes. Additionally, it can be handy to use when it is hard to aim, such as during parts of an overtaking when the opponent is located behind the back. The Sludge thrower is a tool suitable to use at encounters that persist for a while longer such as during an overtaking or at traffic lights. This is due to the procedure of using the tool, which is a bit more time consuming than the Electro squeezer. Similar to the Electro squeezer the Magic wand can be favourable to use in a swift meeting. At a good hit in close proximity of the opponent player it is very powerful. Still, using the Magic wand is also related to a bigger risk of failing. It can for example be difficult to identify the location of the opponent player in time because of intense traffic or dense road networks, such as in a city-centre.

Table 1. Suitability of tools during different traffic encounters

	Meeting	Overtaking	Traffic light
Electro squeezer	Quick and easy to use	Quick and easy to use	Quick and easy to use
Sludge thrower	To slow/bad to use	Easy to use if opponent is in front. Difficult to use if opponent is behind	Easy to use if opponent is in front. Difficult to use if opponent is behind
Magic wand	Difficult to use	Difficult to use, especially if opponent approach from behind	Difficult to use

The reward system is designed so that the player need to choose tool depending on the encounter in order to be successful in the game. The more connection to the opponent player the tool

conveys the more powerful it is. But choosing the most powerful tool is not always the best solution as it also can be difficult to master during certain encounters. Firing the Electro squeezer is very quick and easy but has a low effect on the opponent character. The Sludge thrower is trickier and more time consuming to use than the Electro squeezer but is more powerful. The effect of the Magic wand is dependent on the distance to the opponent player, the closer the more powerful, and is much more powerful than any of the other tools if fired close enough.

5. IMPLEMENTATION

The game is developed on a PDA equipped with WLAN capability to enable network connection between the players. It is aware of the player's aiming direction by means of a digital compass and its geographical position by means of a GPS-receiver. A Basic stamp II microcontroller controls the LED's and the external button. An additional button is also mounted inside the Clutcher in order to accomplish the squeezable feature. A serial cable connects the Clutcher with the PDA (fig. 3).

5.1. Software Architecture

Gaming activity between players during multiplayer events is accomplished through peer-to-peer wireless ad hoc networking, allowing connection between the players without any further infrastructure. *Road Rager* uses the MongerLib library in order to handle this connection [14]. Mongerlib is based on a rapid mutual peer discovery protocol to quickly detect and connect the players when they meet. It takes care of transmitting and receiving information between the connected devices as well as makes sure the devices disconnect properly when coming out of reach from each other. Furthermore MongerLib also obtain the player's latitude and longitude coordinates from the GPS receiver and handles positioning arithmetics such as calculating distance and bearing to the other player.

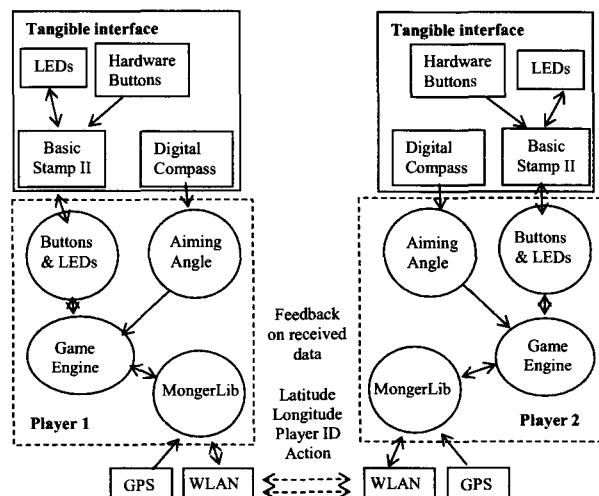


Figure 8. System architecture (during a manipulative event)

A multiplayer event typically proceeds in the following manner:

1. MongerLib detect when two devices are within each other's wireless reach. When MongerLib have established a connection a message is sent to the game engine that will set the game in connected mode. Data can now be sent between the devices.
2. As soon as the game is set in connected mode the game engine starts to continuously trigger its present longitude and latitude to be sent to the other device. When the location data is received from the other device the distance and desired aiming direction to it can be obtained. The desired aiming direction is achieved by calculating the bearing between the co-ordinates of the players. The bearing is defined as the angle measured horizontally from north to the direction of the other player's co-ordinate. By comparing this information with the current angle of the compass the game engine sends messages to the Basic Stamp microcontroller to switch on that locator LED which position corresponds to the desired aiming direction, i.e. towards the direction where the opponent player is physically located. The locator LED's are set to turn on within a range of 45° from the intended aiming direction. Except from the positioning data also information about character identity and of the player performed actions are sent between the connected devices. With performed action we mean if the player fire and with what tool. Rewards for the performed action are not achieved until a feedback on the sent data is received back from the other device. Upon reception of the feedback the magic power is counted up/down and a message is sent to the microcontroller to update the power bar LED's according to the new result.
3. MongerLib detects when two devices come out of each other's wireless reach, it then closes down the connection and sends a message to the game engine, which in its turn sets the game to disconnected mode.
4. The player is then provided with feedback on gaming achievements on the screen of the PDA.

6. TECHNICAL TEST

A technical test was conducted in order to investigate if the prototype would perform as expected. The networking capability had already been tested in prototypes such as Soundpryer [14] and Hocman [6, 7] and proven to work within this setting. A performance criterion critical for the game and important to investigate was rather the accuracy of the aiming direction during a critical situation, such as when the players are standing still in close proximity of each other or during the passing moment of a meeting. The test was carried out in its intended setting and involved a situation where one car passed by a stationary car (fig. 9). The test was monitored from within the moving car. During the test the Clutcher was continuously aimed toward that side of the car where the meeting with the other car eventually would take place, i.e. 90 degrees from driving direction. A camera was mounted to film both the Clutcher and the outside of the side-window at the same time. Afterwards, when looking at the recorded video, a measure of the aiming precision (α) during the moment of meeting could be made. This was accomplished by calculating the distance (x) in meters between the exact moment of the meeting (z) and the turning on/off of the frontal locator

LED. The distance was calculated with the help of the speed of the car and the time-encoding of the video. The test was carried out in 50 km/h as well as 70 km/h and the distance (y) between the cars in the moment of the meeting was 10 meters. The LED was set to turn on/off within an aiming range of 22° from the intended aiming direction (fig. 10).

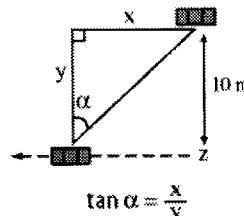


Figure 9. Test situation

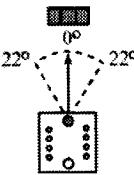


Figure 10. Aiming rage

6.1. The Result

The table below show the results from the test (table 2). A positive number indicates that the locator LED turned on before the point of the meeting and a negative indicates after. The test showed a satisfying result regarding the aiming precision in five out of seven test cases when the car drove 50 km/h. With a satisfying result we mean that the locator LED turned on before or right at the point of the meeting, i.e. it had at most 22 degrees inaccuracy. When the speed was changed to 70 km/h the aiming precision deteriorated considerably. At this speed the locator LED was in all test cases turned on after the exact point of the meeting with an average inaccuracy of 62 degrees. However, as a car at this speed travel 19.4 m/s, this means a lag of 0.43 seconds during the exact moment of the encounter. The aiming range in the game was then change to a value of 45 degrees. A smaller lag was also presumed when the game was designed. With these results we concluded that it would be feasible to carry out a field trial off the prototype, following section will present a summary of user feedbacks from this trial.

Table 2. Test results

	Driving 50 km/h			Driving 70 km/h		
	x (m)	α	inaccuracy	x (m)	α	inaccuracy
1	1,11	6,3°	15,7°	-10,11	-45,3°	67,3°
2	0,00	0,0°	22,0°	-8,56	-40,6°	62,6°
3	3,89	21,3°	00,8°	-9,33	-43,0°	65,0°
4	5,56	29,1°	7,1°	-5,44	-28,6°	50,6°
5	-17,22	-59,9°	81,9°	-7,78	-37,9°	59,9°
6	-1,11	-6,3°	28,3°	-9,33	-43,0°	65,0°
7	-10,56	-46,6°	68,6°			

7. INITIAL USER FEEDBACK

A field trial was conducted in order to discover design flaws and to observe the feasibility of using encounters as resource for the game-play. Furthermore, to get an indication if physical presence and a possibility to identify other players during temporary encounters would spur social interaction and enhance the gaming experience. The test was set up to involve a total of fourteen children, seven children in the age of eight and seven children in the age of ten. The two age groups played the game separately for approximately thirty minutes. Three cars drove simultaneously along a preset route with two to three children in each car. This ensured encounters with other players as well as made it possible to observe the game-play. Initially all participants got an explanation of the game. The activities were video recorded and an interview was carried out after the game-play. Unfortunately, because of certain technical problems, the test cases turn out to be fewer and the game-play sometimes uneven, but they are nevertheless valuable results that indicate possibilities and flaws for the coming evaluation.

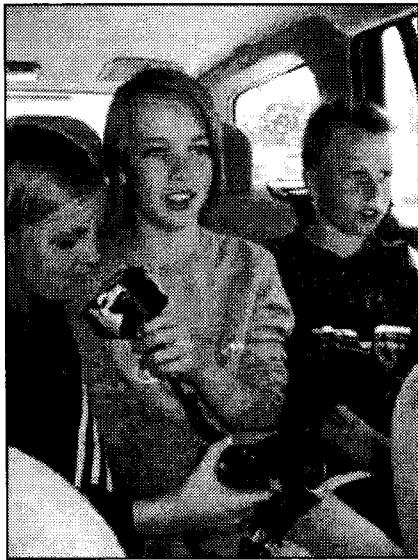


Figure 11. Kids playing Road Rager

It was clear both from the interviews and from observations of the players' behaviors and expressions during the game-play that these temporary encounters created a very thrilling gaming situation. This was not just the case for the player in charge of the Clutcher, but also for the rest of the children in the car. As these gaming events occurred suddenly and often during short periods of time it was usual that all children in the car were involved trying to spot the opponent and to suggest what tool to use. It was also usual that the children divided tasks in between each other so that one was in charge of the PDA and one of the Clutcher or that one was in charge of the game manipulation and one of the searching for the opponent. Situation also occurred when several children held the Clutcher at the same time trying to help each other. Many children mentioned in the interviews that it was the feeling when someone was in the proximity and the searching for the opponent that was the most fun and thrilling part of the game.

Equally, they also mentioned that one of the worst things was if they didn't manage to visually spot the opponent. Another thing that they mentioned as fun was the way they could move and manipulate the Clutcher in order to play the game.

After some experimenting, the majority of the children quickly got the idea of how to manipulate the Clutcher during the encounters and how to interpret the feedback from the LEDs. The tools that were most used during the game-play was the Electro squeezer and the Sludge thrower. Even though several children from the beginning had decided that the Magic wand was the most useful one they soon changed there minds. None of them got the concept of waiting until they where close up before using it, which resulted in disapproval. The tool that was generally considered as the most fun to use was the Sludge thrower, but it was often exchanged by the Electro squeezer because of the difficulty to aim during certain meetings.

8. RELATED WORK

Exploring the possibilities of using traffic encounters as resource in a game is one aspect of a bigger adventure game intended to combine game-play with the highway experience. One prototype has already been developed, called Backseat Gaming, which investigate how to integrate roadside objects as part of the gaming experience [3].

The possibility of using the physical world as game-board has been explored for several year by the industry. Commercially available Botfighters from It's Alive [itsalive.com – verified 1st July 2004] use mobility, location and proximity of players as a resource in the game. Road Rager is related to the ideas of Botfighters but explores the impact of proximity during temporary moments for the gaming experience. In Botfighters location is determined with GSM mobile phone positioning, which gives relatively high positioning inaccuracy making it highly unlikely that the players would ever meet while playing the game.

A number of research projects explore the idea of integrating tangible, social and human to physical world interaction into digital and ubiquitous games [2, 4]. These projects are designed for use in a pre-set room and exploring the possibilities of using true mobility as a resource in a gaming constitutes a different design challenge. Examples of games that draw on the other players' physical proximity without any preset infrastructure include PacMan Must Die and Earth Defenders [13] but these games are designed for use during longer periods of co-location of the players and not for short occasional encounters. An example of a game exploiting issues of incorporating different aspects of mobility and the physicality within the experience in an outdoor setting is Can you see me now? [9] This game explores collaboration between online participants and mobile participants on the street.

9. CONCLUSION

We have presented a game prototype, designed to make use of contingent traffic encounters as a resource, in order to explore our hypothesis that proximity and a possibility to identify other players during temporary encounters could spur social interaction and enhance a mobile gaming experience. We have also presented a technical test and some initial user feedback on the gaming

experience. Important design criteria include how to support the fictitious connection between the game and the real world and simultaneously cultivate the player's fantasy and imagination, how different kinds of encounters call for different gaming situations and how identification, social interaction and awareness could be supported between players. The initial user feedback gives a strong indication that encounters and the motion of the accompanying traffic, occurring during car traveling, can be used to create a compelling and fun game. The user feedback also indicates that the possibility to identify other players can spur social interaction and enhance the gaming experience. This result motivates us to proceed with our research and future work includes an extensive user evaluation of the prototype.

10. ACKNOWLEDGMENTS

We would like to thank Alberto Frigo, Kristina Hulterström, Oskar Juhlin, Mattias Östergren, Mattias Esbjörnsson and other members of the Mobility Studio. The Swedish Foundation for Strategic Research funded this research.

11. REFERENCES

- [1] Appleyard, D., Lynch, K., and Myer, J. *The View from the Road*. M.I.T. Press, Cambridge, Massachusetts, USA, 1964.
- [2] Björk, S., Falk, J., Hansson, R., and Ljungstrand, P. Pirates! Using the Physical World as a Game Board. *In Proceedings of Interact'2001*, Tokyo Japan, 2001.
- [3] Brunnberg, L., and Juhlin, O. Movement and Spatiality in a Gaming Situation - Boosting Mobile Computer Games with the Highway Experience. *In Proceedings of Interact'2003*, Zürich Switzerland, 2003.
- [4] Cheok, A.D., Yang, X., Ying, Z.Z., Billinghurst, M., and KATO, H. Touch-Space: Mixed Reality Game Space Based on Ubiquitous, Tangible, and Social Computing. *Journal of Personal and Ubiquitous Computing*, 2002.
- [5] Chincholle, D., Goldstein, M., Nyberg, M., and Eriksson, M.. Lost or Found? A Usability Evaluation of a Mobile Navigation and Location-Based Service. *In proceedings of the 4th International Symposium, Mobile HCI 2002, Pisa Italy*. Pages 211-224, 2002.
- [6] Esbjörnsson, M., Juhlin, O., and Östergren, M. Traffic Encounters and Hocman - Associating Motorcycle Ethnography with Design. *Journal of Personal and Ubiquitous Computing*. Pages 92-99, 2002.
- [7] Esbjörnsson, M., Juhlin, O., and Östergren, M. The Hocman Prototype - Fast Motor Bikers and Ad Hoc Networking. *In Proceedings of MUM 2002, Oulu Finland*. 2002.
- [8] Eskelinen, M. The Gaming Situation, in *Game Studies – The International Journal of Computer Game Research Issue 1*, 2001.
- [9] Flintham, M., ET AL. Where On-line meets on-the-streets: experiences with mobile mixed reality games. *In proceedings of CHI'03*, Pages 569-576, 2003.
- [10] Ishii, H., and Ullmer, B. Tangible bits: towards seamless interfaces between people, bits and atoms. *In proceedings of CHI'97*, 1997.
- [11] Juhlin, O. Traffic behaviour as social interaction – Implications for the design of artificial drivers. in Glimell and Juhlin (eds.), *Social Production of Technology: On everyday life with things*, BAS Publisher, Göteborg. Sweden, 2001.
- [12] Kuivakari, S. Mobile Gaming: A journey back in time. *Computer Games & Digital Textualities*, 2001.
- [13] Sanneblad, J., and Holmquist, L.E. Designing Collaborative Games on Handheld Computers. *In proceedings of the SIGGRAPH 2003 conference on Sketches & applications*, San Diego USA, 2003.
- [14] Östergren, M. Sound Pryer: Adding Value to Traffic Encounters with Streaming Audio. In proceedings of ICEC 2004, 3rd International Conference on Entertainment Computing, Eindhoven The Netherlands, 2004.

UMAR - Ubiquitous Mobile Augmented Reality

Anders Henrysson

Norrköping Visualization and Interaction Studio
Linköping University, Norrköping, Sweden

andhe@itn.liu.se

Mark Ollila

Norrköping Visualization and Interaction Studio
Linköping University, Norrköping, Sweden

marol@itn.liu.se

ABSTRACT

In this paper we discuss the prospects of using marker based Augmented Reality for context aware applications on mobile phones. We also present the UMAR, a conceptual framework for developing Ubiquitous Mobile Augmented Reality applications which consists of research areas identified as relevant for successfully bridging the physical world and the digital domain using Mobile Augmented Reality. A step towards this we have successfully ported the ARToolkit to consumer mobile phones running on the Symbian platform and present results around this. We also present three sample applications based on UMAR and future case study work planned.

Categories and Subject Descriptors

H.5 [Information Interfaces and Presentation]: Multi-media Information Systems

General Terms

Human Factors, Experimentation, Design

Keywords

Augmented Reality, Pervasive Computing

1. INTRODUCTION AND BACKGROUND

In this paper we present exploratory research in the area of mobile computer graphics and interaction using augmented reality and context aware environments. Currently, mobile devices have limited processing power and screen size due to their small size and dependency on batteries with limited life times. However, the mobile device is the most ubiquitous device and a part of most peoples everyday life. Hence, as mobile devices become more advanced, the development of 3D information systems for mobile users is a growing research area [22]. Being mobile means that the context changes and that information, and computation can follow

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MUM 2004 October 27-29 2004 College Park, Maryland, USA

Copyright 2004 ACM 1-58113-981-0/04/10 ...\$5.00.

the person. We would like to exploit the features of a mobile device while trying to work around its limitations in order to study its potential to enhance the experience of the users. We have chosen to work with smartphones since they have the capability of rendering and displaying graphics such as video and 3D animations. They are connected to data enabled networks such as GPRS and UMTS and many of them now feature built in cameras (with some models having GPS built in).

The idea behind augmented reality (AR) is to track the position and orientation of the users head in order to enhance his or her perception of the world by mixing a view of it with computer generated visual information relevant to the current context. AR is useful when we have to solve a real world problem using information from another domain such as a printed manual or a computer screen. By projecting the information onto a view of the real world there is no need for distracting domain switching. The information is fetched from a virtual representation of the real world environment to be augmented, using tracking, or it could also be derived from direct object recognition. We, however, will only discuss the case where tracking is involved. Tracking can be performed with a wide variety of sensors such as GPS, optical, inertia trackers, ultrasonic trackers etc. Since the smartphones we are working with has built in cameras we have chosen to work with optical tracking where we track the orientation and translation of the camera.

AR has traditionally been reserved for high-end computers while mobile augmented reality (MAR) has used custom-built hardware setups. Many of these consist of a laptop mounted on a frame carried as a backpack [20]. They often feature head mounted displays (HMDs) and as such, the majority are research platforms inaccessible to average consumers. Instead, in this paper, we have focussed on consumer level mobile phones where devices such as PDAs and smartphones have developed rapidly and have enough rendering power to do 3D graphics. As concluded in [13], the smartphone meets all of the requirements posed by AR to some extent.

We now present a background into relevant research regarding AR, mobile devices and tracking. The AR-PDA [16] setup consists of a PDA with a camera. The PDA is connected to an AR-server using WLAN and sends a video stream to the server for marker-less tracking and rendering. The augmented video stream is returned to the PDA for display after processing on the server. Similarly, AR-Phone [14] and PopRi [7] have still images sent to a server running ARToolkit [1] for augmentation. PopRi also allows contin-

uous augmentation when run from a video enabled phone. The drawback of these client-server setups is that the user is dependent on a fast connection to a server. In a wide area application where no WLAN is accessible the images must be sent over a network such as GSM/UMTS that usually comes with a cost per byte or minute for data traffic and, obviously, latency issues. An implementation of the AR-ToolKit on the PocketPC platform was performed by [15]. They analyzed the performance of different functions in the ARToolkit and identified the most computationally heavy. These have been implemented using fixed-point arithmetic for significant speed up. However the PDA still requires WLAN or Bluetooth for communication, which limits its use in dynamic wide area applications where these networks are not available. The Real in Real project from Japanese operator NTT [9] uses a Tablet PC as the viewing device. It tracks a sensor cube with sides that consist of ARToolkit markers. The sensor cube is equipped with sensors that measure the incoming light. This information is then used to produce realistic renderings. Video See-Through AR and Optical Tracking with Consumer Cell Phones project [18] does 6DOF tracking on a smartphone at interactive frame rates. It tracks a 3D marker onto which the three coordinate axes are printed. By detecting these lines in the image the camera transformation can be estimated and used for rendering. The disadvantage is that it requires a marker with more complex topology than the simple 2D markers of for example, the ARToolkit. The SpotCode platform [4] uses markers to turn a camera-enabled phone into a virtual mouse pointer for interaction with digital content displayed on a screen. The camera phone identifies the marker and sends its id and relative position and orientation to a nearby computer using Bluetooth. The markers consist of data rings with sectors each of which encodes a single bit of information. Thus there is no need for the system to know the particular marker in advance and the markers are of known complexity. They can be used for real world hyperlinks where a marker id is mapped to an URL for instance.

IDeixis [21] uses images taken with a camera phone to recognize the location and provide context related information using a hybrid image-and-keyword searching technique. It works by using an image captured by a camera phone for content-based image retrieval (CBIR) in a limited image database. If there is a hit it continues by extracting a keyword that is used to search Google to find related images. These are matched to the captured image to determine relevance. Siemens has developed an AR game called Virtual Mosquito Hunt [12] where the real time video stream from the onboard camera is augmented by virtual mosquitoes for the player to kill.

An environment for context aware mobile services is SmartRuuaari [19]. It is based around PDAs and uses WLAN for communication and positioning. The context information also includes weather data. It provides services such as map-based guidance, mobile ads and interactive 3D visualization of historical data. Currently, there is no augmented reality in use.

Optical tracking can be marker-based or marker-less. In the case where we have markers the system calculates the orientation and translation of the camera in a coordinate system where the origin is in the center of a marker. In marker-less tracking we have to rely on features in the environment such as edges to track the camera. At a first

glance marker-less tracking seems more beautiful but having visible markers has the advantage of telling the user where environment has been augmented. The problem is analogous to that of hyperlinks in MPEG-4 video.

Marker-based optical tracking gives accurate results but is limited by the visibility of the marker. The markers have to be scaled by distance so that they are identifiable by the system. Here the complexity of the marker plays an important role. A low frequent pattern (large white and black regions) is easier to recognize but simple patterns limit us to a small set of distinct markers. Some marker-based systems need to learn a pattern before it can be used [1] while others contain information that can be translated into an identification number [4].

However, the benefits of camera tracking include, but are not limited to: 1) Virtual screen: By tracking a single marker it is possible to pan over a virtual screen up to 4 times bigger than the physical screen; 2) Positioning: Information tied to location. Tracking the camera in the environment yields accurate position information that can be used to adapt and configure services; 3) Camera phone as optical mouse pointer: Interact by moving the phone relative to marker.

The research we are conducting is to investigate how AR can be used to enhance the limited user interface of current smartphones and transform them into tools for interacting with the real world (through mobility). Mobility means that the context switches according to e.g. position, climate etc. By estimating the context a device or a service can be configured to better suit its user. The current context can be estimated by a wide variety of sensors e.g. GPS for positioning. In order to configure services as such, the system needs to know the user's personal preferences, which might also depend on context. The paper is organized as follows: Section 2 we present our framework for ubiquitous mobile AR. Section 3 describes the implementation of ARToolkit on a smartphone and also our implementation of context aware video as an example of a non-augmented context aware mobile application. Section 4 presents the three application and results. Section 5 examines future work planned.

2. UMAR FRAMEWORK

UMAR is a conceptual framework that consists of research areas required to perform Ubiquitous Mobile Augmented Reality where we bridge the digital and real domains with context aware information. For an arbitrary context we want to fetch the relevant information and display it using appropriate techniques depending on the spatial relationship between context and retrieved information. If there is a close spatial relationship we would prefer to use AR. If the spatial relationship is weaker we could use a 2D map similar to [19]. If there is no spatial relationship the information could be presented e.g. as a web page or as audio using text-to-speech depending on the user preferences. In contrast to specialized HMD configurations, the smartphone can easily switch between modalities.

Information retrieval based on context awareness and personalization is an important part of the overall framework but not a primary focus. Work in this area has been done as part of the Context Aware Pervasive Networking program [2] and also in the area of MPEG-7 when it comes to indexing media [5]. Future semantic web technologies [8] will be needed as we ultimately will need to search the entire web

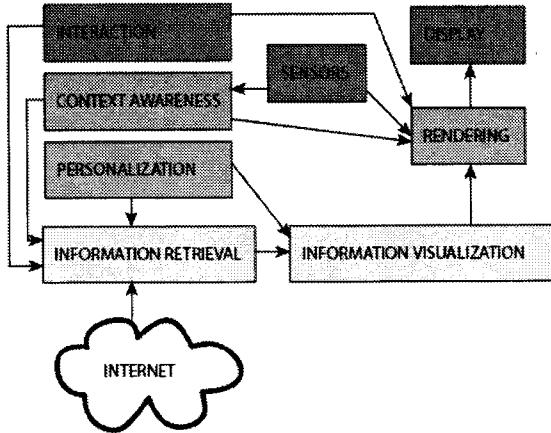


Figure 1: Overview of the UMAR Framework.

for relevant information and not be limited to custom-built databases. The hybrid search of [21] was able to search among 425 million indexed images and related web pages using a single image as input.

The retrieved information would need to be classified according to the mentioned spatial relationship to the real world scene surrounding the user and converted into graphical representations. This later is a problem of information visualization. Based on the user preferences, other media types e.g. web pages and audio could be synthesized. In the AR case we render a virtual object using the estimated camera before compositing it into the frame where the marker was detected. Ideally we would like to use context information for the rendering similar to [9] in order to produce photorealistic images. The rendering technique will depend on the object representation e.g. polygonal and must be adapted to the device in order to meet QoS demands such as interactive frame rate. The overview of the UMAR framework is shown in Figure 1. The arrows can be seen as data flow between possible outcomes from different areas but they can also represent research questions. Darker color means that the issues are more likely to be studied on the client side.

The goal for UMAR is to perform as much as possible on the client to reduce the data traffic and avoid being dependent on fast network access. While the sensors, display and interaction UI is tied to the device itself the remaining issues can be server-based if necessary. The information retrieval and visualization issues where information is retrieved and converted to graphical representation will most likely be studied on the server side for non-trivial scenarios.

3. IMPLEMENTATION

We have chosen to work with the ARToolkit [1] which is an open source toolkit for optical tracking. Besides a main library for tracking and marker identification, it also contains camera calibration software. It works by identifying markers in a video stream and calculate the orientation and translation of the camera in a reference coordinate system centered at a marker image. It performs the following steps:

- 1) Turn captured images into binary images;
- 2) Search the

binary image for black square regions;

- 3) For each detected square the pattern is captured and matched against templates;
- 4) Use the known square size and pattern orientation to calculate the orientation and translation of the camera;
- 5) Draw the virtual objects.

We have implemented two applications based on this conceptual framework. First we implemented a simple context-aware video service based on a subset of the UMAR framework in order to have a simple case study (see Figure 2). Secondly, we ported the ARToolkit 2.65 on to a smartphone running Symbian [11] and Series60 [10] for the purpose of evaluating marker-based optical tracked AR on a consumer device. ARToolkit consists of an AR library foremost responsible for marker identification and tracking, calibration software and an application for learning new markers. We treated ARToolkit as a black box that takes camera parameters and an image as input and returns a camera transformation matrix. The camera matrix consists of intrinsic and extrinsic parameters. The intrinsic parameters relates to the properties of the camera such as focal length. Together with distortion parameters they are used to link a pixel coordinates with the corresponding coordinates in the camera reference frame. These parameters are calculated once in a calibration process. To do this we modified the calibration application to take still images captured by the camera phone. Five images containing 81 points each were used. The extrinsic parameters are estimated each frame in an iterative process and consist of the orientation and translation of the camera in the world coordinate frame centered at the marker. When rendering, the extrinsic parameters correspond to the view transformation in the rendering pipeline and the intrinsic parameters correspond to the perspective transformation.

4. EXPERIMENTS AND RESULTS

To test the performance we built a test program for the mobile device (a Nokia 6600) with a 104 MHz ARM9 CPU, 6 MB of volatile memory and a screen size of 176*208 pixels with 16 bits per pixel. It also has a 0.3 megapixel camera. We used images at a resolution of 160*120 pixels with 12 bits per pixel color depth. For our tests we used a single marker with low complexity (size was 8 by 8cm). As an UMAR application example we placed a marker on top of a tram route map similar to those found at tram stops. The idea was to augment the map with an animation showing the current position of the trams trafficking the line. The stations were identified on the map and by using the arrival/departure time for the end stations and the current system time, the relative position of tram could be calculated. The absolute positions in mm relative to the marker center could then be calculated. The tram route was drawn by a polyline and the trams represented with sprites. Since each station can have a unique marker, the users position can be estimated accurately and location-based information could easily be added. In a more sophisticated case all the above steps, now done manually, would likely be made automatically by using appropriate web semantics to tag the timetable data (see Figure 3).

The second UMAR application involved context-aware mobile video. We developed a minimalistic setup consisting of a client and a combined context and video server. The setup targets parts of the UMAR framework e.g. sensors, context awareness and information visualization. The client

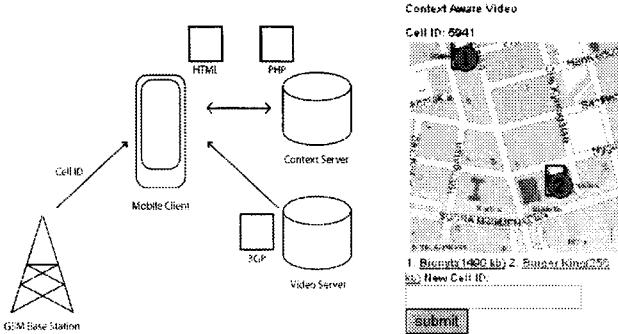


Figure 2: Context Aware Video Architecture based on UMAR Framework

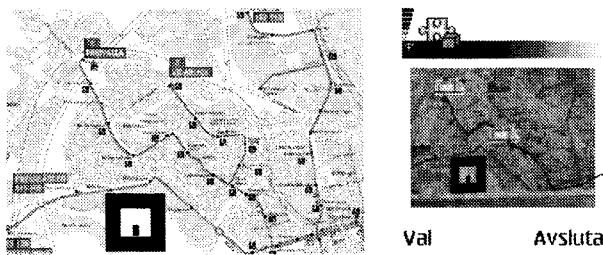


Figure 3: Map with marker and screenshot from mobile device.

was a Sony Ericsson P800 smartphone and our implementation involved integrating three commercial applications: Psiloc miniGPS, Opera browser and PacketVideo pvPlayer; plus our own research software. Context retrieval is performed by miniGPS, an application that displays the cell ID number of the GSM base station used. The application can alert the user when a cell is entered or exited. The GUI consists of a web page displaying an approximal map of the current cell. It has a form where the cell ID is entered manually. Available videos are marked with numbered legends and are presented as hyperlinks. When clicked, the user is presented a dialog box where she or he can choose to open the video file. Lacking a compatible video server we settled with download of the entire clip prior to playing. If the video is downloaded it will automatically be played by pvPlayer. When the video is finished the user can return to the GUI with the click of a button. The client communicates with a context server which is a web server running a PHP script to generate the HTML document to be displayed. In our current implementation the cell IDs, maps and legends are static. The cell ID:s were obtained by walking around the town of Norrköping to which the application is limited and the maps have been obtained from Gula Sidorna [3]. Thirdly, as a hybrid between the two applications we also implemented the tram animation using a bitmap route map as background image, i.e. a context-aware video rendered on the client. Since we assumed no real-world map to augment in this case, we have a weaker spatial relationship between the animation and the users context and are thus using a simpler visualization technique.

A closer look at the implementation reveals that the AR-Toolkit uses floating points with double precision for the representation and calculation of the camera matrix. Since smartphones lack FPU all floating-point arithmetic is emulated in software with a big computational overhead compared to hardware (performance is hundreds of times slower than integer performance). Because of this the full 6DOF tracking could not be done at interactive frame rate but closer to about one frame per second. However 3DOF tracking (2D translation + area to estimate z-value) and marker identification can be done at interactive frame rates with no perceived performance degradation compared to the video frame rate provided by the camera. We experimented with fixed point implementations based on the analysis in [15]. However since there is no corresponding fixed-point library freely available for the Symbian platform we were not able to achieve satisfying results using simple implementations. The maximum range for marker detection using our setup was close to 1.5 m and the Nokia 6600 performed better than previously tested smartphones [13]. The limited range is not a big problem since the quality of the real-time video stream is fairly low and the screen size is small which makes long range tracking less attractive.

5. CONCLUSIONS AND FUTURE WORK

We have successfully ported the ARToolkit to the Symbian platform though some performance issues remain to be solved. We have shown that smartphones can be used for AR without server assistance. We have presented UMAR, a conceptual framework for further research in Mobile Augmented Reality in Context Aware Pervasive Environments. We have implemented simple UMAR applications to show different visualization techniques depending on the level of spatial relationship between information and context. The current platform is limited to the close proximity of a marker in order to provide AR. We would like to expand the platform to incorporate wide area tracking using GPS or cell ID. To extend the marker tracking we would like to look at feature tracking such as [17]. It would also be interesting to study to what extent optical flow measurements could assist the optical tracking. On the rendering side we need to incorporate an efficient 3D renderer (we need to be able to define the view and perspective matrices) or an implementation of OpenGL ES [6]. It would be desirable to have a standard for content representation. We need to implement fixed-point arithmetic with variable precision in order to solve the performance issues. We will also see if we can further adapt the ARToolkit to the smartphone platform. The next generation of smartphones features in some cases both GPS and megapixel cameras. There are also phones with a tilt sensor and digital compass which will open up new possibilities when it comes to tracking and will be an obvious research platform. Over the next phases of the research, field trials and user evaluations will take place, giving us both qualitative and quantitative results.

6. ACKNOWLEDGEMENTS

The first author is supported by a department grant from the Department of Science and Technology at Linköping University. This project was partially funded by a grant from HomeCom. We also acknowledge the support of Prof. Anders Ynnerman.

7. REFERENCES

- [1] Artoolkit. www.hitl.washington.edu/artoolkit/.
- [2] Capnet. www.mediateam.oulu.fi/projects/capnet/.
- [3] Gulasidorna. www.gulasidorna.se.
- [4] High energy magic. www.highenergymagic.com.
- [5] Mpeg/7 overview. www.chiariglione.org/mpeg/standards/mpeg-7/mpeg-7.htm.
- [6] Opengl es. www.khronos.org/opengles.
- [7] Popri. www.popri.jp/real_in_real/PopRi.htm.
- [8] Rdf. www.w3.org/TR/REC-rdf-syntax/.
- [9] Real in real. www.contents4u.com.
- [10] Series 60. www.series60.com.
- [11] Symbian. www.symbian.com.
- [12] Virtual mosquito hunt. w4.siemens.de/en2/html/press/newsdesk_archive/2003/foe03111.html.
- [13] H. Anders and O. Mark. Augmented reality on smartphones. In *2nd IEEE International Augmented Reality Toolkit Workshop*, 2003.
- [14] D. J. C. Dan Cutting, Mark Assad and A. Hudson. Ar phone: Accessible augmented reality in the intelligent environment. In *OZCHI2003*, Brisbane, 2003.
- [15] W. Daniel and S. Dieter. Artoolkit on the pocketpc platform. In *2nd IEEE International Augmented Reality Toolkit Workshop*, Waseda University, Tokyo, Japan, 2003.
- [16] C. Geiger, B. Kleinnjohan, C. Reiman, and D. Stichling. Mobile ar4all. In *2nd IEEE and ACM International Symposium on Augmented Reality (ISAR 2001)*, New York, USA, 2001.
- [17] M. B. Hirokazu Kato, Keiachiro Tachibana and M. Grafe. A registration method based on texture tracking using artoolkit. In *2nd IEEE International Augmented Reality Toolkit Workshop*. Waseda Univ., Tokyo, Japan, 2003.
- [18] C. L. Mathias Möhring and O. Bimber. Video see-through ar on consumer cell-phones. In *In proceedings of International Symposium on Augmented and Mixed Reality (ISMAR'04)*, 2004.
- [19] T. Ojala. Smartrotuaari - context-aware mobile multimedia services. In *2nd International Conference on Mobile and Ubiquitous Multimedia*, Norrkoping, Sweden, December 2003. ACM Press.
- [20] T. H. S. Feiner, B. MacIntyre and T. Webster. A touring machine: Prototyping 3d mobile augmented reality systems for exploring the urban environment. In *Proc. ISWC '97 (First IEEE Int. Symp. on Wearable Computers)*, Cambridge, MA, 1997.
- [21] K. T. Tom Yeh and T. Darrell. Searching the web with mobile images for location recognition. In *CVPR 2004.*, 2004. To appear.
- [22] T. Vainio and O. Kotala. Developing 3D information systems for mobile users: some usability issues. In *Proceedings of the second Nordic conference on Human-computer interaction*, pages 231–234. ACM Press, 2002.

The GapiDraw Platform: High-Performance Cross-Platform Graphics on Mobile Devices

Johan Sanneblad and Lars Erik Holmquist

Future Applications Lab, Viktoria Institute

Horselgangen 4, SE-41756 Goteborg, SWEDEN

www.viktoria.se/fal

{johans, leh}@viktoria.se

ABSTRACT

The *GapiDraw* platform supports the creation of high-performance graphical applications across a variety of handheld hardware configurations, including Palm, Symbian and Windows Mobile devices. Handheld computers makes it possible to create applications and services not possible with stationary computers, thus there is a need for a high performance development platform for rapid prototyping on mobile devices. Unlike desktop computers there has not yet evolved a single standard for graphics on handheld devices. Typically, handheld computers only provide direct frame buffer access, and there are major differences in implementation details across different hardware configurations, making it difficult to use mobile devices for prototyping. Using *GapiDraw*, developers can re-use the same code across a variety of devices and do not have to focus on device-specific implementation details. *GapiDraw* is actively used as an enabler platform in numerous research labs, and has also been used in over one hundred commercial games. We give an overview of the platform, and highlight some new mobile application concepts made possible through the use of *GapiDraw*.

Keywords

Handheld computers, mobile phones, mobile games, graphics framework, graphics API, graphics middleware, prototyping

1. INTRODUCTION

Handheld computers are rapidly gaining in popularity – and the capabilities of such devices are increasing at an accelerated pace. From Apple's *Newton* and Palm's *Pilot*, with their monochrome screens and processor speed in the single-digit megahertz range, current handhelds outstrip the performance of a stationary computer only a few years old. A typical PocketPC such as the *Toshiba e805* has a screen with 65.535 colors and a resolution of 480 x 640 pixels, a processor running at 400 megahertz, and 128 megabytes of RAM. This evolution means that handheld computers will see a shift from low-resolution grayscale graphics and text-based interfaces, to interactive applications that take full advantage of these new possibilities – including high-resolution, full-color graphics and full-motion

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MUM 2004, October 27-29, 2004 College Park, Maryland, USA.
Copyright 2004 ACM 1-58113-981-0/04/10... \$5.00

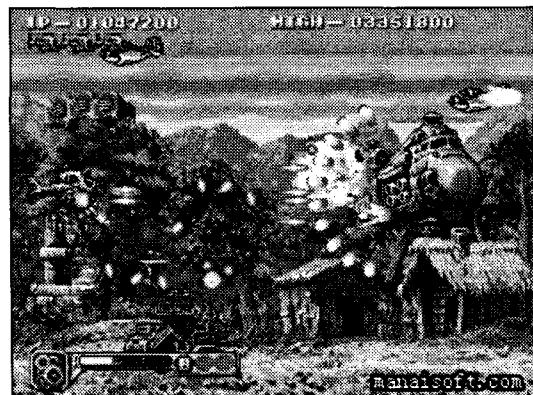


Figure 1: *FirePower - Onrush* by ManaiSoft is one notable game that uses the *GapiDraw* platform.

video.

Unfortunately, while for stationary PCs there has been a convergence in graphics hardware and software towards a common developer platform (the Microsoft DirectX platform [3]), this has not yet been the case for handheld computers. Because of the heterogeneous hardware specifications of mobile devices, developers who want to create high-performance graphic applications on one handheld computer have to create new optimized graphics routines for each new device they want to target. Important hardware variations include frame buffer orientations, cache memory sizes, memory access latencies and timer granularities. Very few handheld computers currently offer graphics acceleration in hardware, which means that the speed of the graphics routines is crucial in the performance of software applications.

We believe that this lack of a common development standard is seriously hindering the creative potential of mobile devices. As a solution, we introduce *GapiDraw*, a freely usable graphics platform for handheld computers. Originally developed specifically for games programming (hence the name, which stands for Game Application Programming Interface for Drawing), *GapiDraw* supports all applications that require high performance graphics. It borrows many features from DirectX, easing the transition for programmers already familiar with games programming on stationary PCs. *GapiDraw* supports development over a wide range of handheld devices, including Palm handheld computers, Symbian mobile phones and all Windows Mobile devices (Pocket PCs and Smartphones). For development and testing purposes it also runs natively on stationary PCs. *GapiDraw* was first used in education and is supported by a lively developer community and an active online forum. *GapiDraw* is used

extensively in commercial applications, making it something of a current de-facto standard for mobile application development. One of the more notable releases so far is *FirePower - Onrush* by ManaiSoft (see Figure 1).

2. HANDHELD COMPUTER GRAPHICS

The sheer number of different mobile devices and mobile operating systems has made it quite difficult to create mobile applications that run on more than one device. In fact, the uniqueness of the various operating systems has made it difficult to create applications for just one device, since mobile application development in many cases is very different from developing applications for stationary computers. Thus, there have been several software platforms released to assist with application development for mobile devices. We will now highlight some of the platforms for creating high-performance graphic applications on mobile devices, and then introduce our GapiDraw platform and relate it to these platforms.

2.1 Graphic Hardware APIs

With *Graphic Hardware APIs* we mean Application Programming Interfaces that expose the video hardware of a device, with much of the functionality available only if the appropriate video hardware is available. *OpenGL ES* is such an API and is an embedded subset of the industry-standard OpenGL graphics platform [11]. OpenGL ES has been designed for today's mobile microprocessors (e.g. replacing floating point arithmetic with fixed point), while still providing a feature set matching that of stationary PCs just a few years ago. The numerous differences to the stationary OpenGL API however may make it difficult to create cross-platform OpenGL / OpenGL ES applications using OpenGL ES (the lack of a GLUT [11] for OpenGL ES is one such thing), and it is today not clear what devices will ship with OpenGL ES compatible hardware in the near future. Devices that will not support OpenGL ES are Microsoft Windows Mobile devices, and Microsoft ships a product called *Direct3D mobile* [3] with their operating system Windows CE 5.0. Direct3D Mobile has a similar API to DirectX 8.0, but lacks most of the more advanced features used in many game titles for stationary PCs, and also has several API changes such as replacing floating point arithmetic with fixed point. When creating applications for mobile devices that take advantage of graphics hardware through APIs such as these, developers have to adapt their applications to the feature sets of each graphics library individually for each device they want to support.

2.2 Java-Based Graphic APIs

Java-enabled mobile phones are many, and so is the number of *Java-Based Graphic APIs*. Java on mobile devices (e.g. Java2 Micro Edition, J2ME) is in most cases a small subset of Java on stationary PCs, with performance most suitable for simple visualizations and puzzle games. To improve visual performance on Java for mobile devices, some mobile phone manufacturers ship native rendering libraries with Java APIs with their devices. Examples are *Mophun* by Syngenix, *Brew* by Qualcomm and *JBlend* by Aplix. Typically, Java-Based Graphic APIs such as Brew or JBlend only work on mobile phones, making it difficult to target other devices such as handheld computers using the same code base. Being designed for a small footprint, these graphic

libraries provide a minimum of functionality, making them difficult to use for prototyping.

2.3 Frame-Buffer APIs

Many handheld computers and Smartphones do not ship with a graphics API suitable for high-performance interactive applications such as games. Examples of such devices are Palm, Symbian and Windows Mobile devices. Thus there has evolved a market for *Frame-Buffer APIs*, which provide software-based rendering operations that draw graphics directly into the frame buffer of the device, using only the CPU as a graphics engine. Currently there are a few such platform-specific graphic libraries available, for example PocketFrog [4] on Windows Mobile, and Razor [12] on the Palm platform. One library worth mentioning is PocketHAL [4], which provides a unified way to access the frame buffer on Windows Mobile and Symbian devices, but does not by itself provide any drawing tools.

2.4 GapiDraw

The GapiDraw API is a mix between a Frame-Buffer API and a Graphic Hardware API. Most graphics operations provided by GapiDraw use graphics hardware if available (through DirectX or other APIs such as TwGfx on the TapWave Zodiac device), but every graphic operation is also implemented in software using techniques such as template meta programming and platform-specific assembler code. GapiDraw differs from the previously mentioned APIs in three ways. Firstly, GapiDraw supports mobile application prototyping through a high level framework that abstracts device-specific issues such as event handling, interfacing with the operating system, stylus control, image handling and file management. By using the GapiDraw framework, developers only have to consider the actual logic of their applications, without dealing with device-specific tasks. Secondly, GapiDraw was initially designed for handheld computers, and then later adapted to work on stationary PCs. There is no "downscaling" involved as with OpenGL ES or Direct3D Mobile. Thirdly, GapiDraw is currently the only graphic API that works across all mobile devices using the Palm, Symbian or the Windows Mobile operating systems.

3. DESIGNING A MOBILE GRAPHICS PLATFORM

GapiDraw was designed from the beginning with mobile devices as the prime target, and thus the implementation of GapiDraw is strongly influenced by current mobile hardware. During the design process, companies such as PalmOne, Microsoft and TapWave assisted with both device hardware and implementation details to aid with development. Below we highlight some of the design challenges that had to be considered when creating GapiDraw.

3.1 Frame Buffer Color Depths

Optimizing the performance of direct frame-buffer access has been a regularly visited research topic ever since the first raster-scan displays became publicly available almost three decades ago (e.g. the SUN display [1]). Different approaches for alternative frame buffer configurations (such as the 8 by 8 display [15]) were introduced for performance improvements, but the format that remained was the linear frame buffer format that was introduced with the first displays.

Typical mobile display depths are 12-bit (4096 color) or 16-bit (65535 color). To draw images to the display, developers have to manually convert RGB color values into the current native screen format and write them as bytes to the frame buffer of the device. If the display uses a 16-bit color depth, each pixel occupies 2 bytes. This representation is shown in Figure 2 (in the figure, the Palm OS 5 display is stored in Little Endian format where the rest of the operating system uses Big Endian, which explains why it might look odd). To convert an RGB value to native screen format, the processor has to perform up to ten operations for each pixel (four AND, three SHIFT and three OR for the Palm device). Since this is a costly operation, images should be pre-rendered to match the native format of the frame buffer before being used in any graphics operations. One difference between current mobile devices and stationary PCs with regards to pixel formats is that the color depth of the frame-buffer cannot be changed, where stationary PCs can re-initialize the frame-buffer to use any color depth in any resolution. Thus, supporting multiple pixel formats on mobile devices also means that several graphics routines have to be created, one for each display type, for optimal performance. One way to achieve this is to use *template meta programming*, which is described later.

3.2 Frame Buffer Orientations

Implementing high performance graphics on mobile computers, developers need to reconsider hardware aspects that were a common research topic almost three decades ago on stationary PCs. One such optimization is cache optimizations when reading and writing pixel data to the display frame buffer. For design reasons, displays are internally aligned differently on various mobile devices. Form factor is important, and the display and its connector are often rotated 90 or 180 degrees to decrease the physical size of the device. Currently there are mobile devices available with all of the four possible frame buffer orientations. Two such examples are seen in Figure 3. In the figure, the same four colors are displayed in the top left corner of the display. Depending on how the display is physically oriented in the device, the location in memory of the colors varies between devices.

The variation in frame buffer orientation is one of the more difficult issues to target when creating high-performance graphics for mobile devices. On stationary PCs, displays always present themselves to the developer in one format on all computers (where *xPitch*, the number of bytes to add to step to the next pixel, is always the size of one pixel, and *yPitch*, the number of bytes to add to step to the next row, is the 32-bit aligned width). If the display is internally stored in another format, video hardware will transform the display pixels to this format with a minimum of overhead (called a *swizzle*). The advantage of using a single display format is related to how memory is accessed using a *data cache*. Reading a single byte from any location in memory will automatically cause a computer to read an additional number of consecutive bytes (a *cache line*) and place that in the data cache. On a handheld computer using the ARM CPU, the size of the cache line is 32 bytes. The data cache stores a specific number of

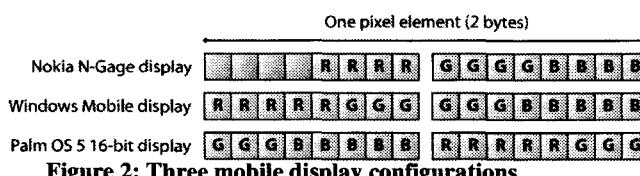
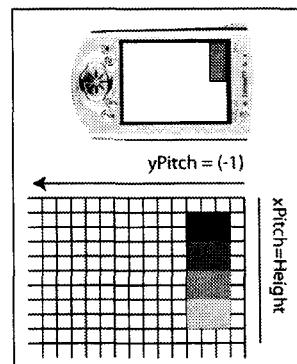


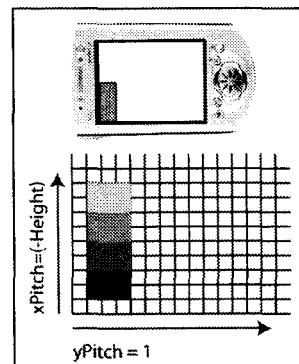
Figure 2: Three mobile display configurations.

1:IPAQ 3600 DEVICES



Pixel Origin = (Height-1)

2:IPAQ 3800 DEVICES



Pixel Origin = (Width-1) * (-xPitch)

Figure 3: Two frame buffer orientations.

cache lines (the ARM CPU stores 256 cache lines, which equals 8kb of data cache), its contents depending on what data that was last requested. The advantage of using a data cache is that data retrieved from the cache is accessed significantly faster from it than data from the main system memory.

Using a display format such as the one used on stationary computers, data is read and written to the display in a format that matches the data cache. Reading the first pixel will cause the cache to automatically contain the second, third, and up to the 32nd pixel of that same row (depending on the pixel format). On displays that are rotated, such as those used on many handheld computers, the alignment of the display must be analyzed so that pixels are read in a way so that the cache memory is always optimally used. For example, in Figure 3, eight pixels should be copied three times (3-1), and three pixels should be copied eight times (3-2). While issues related to reading and writing non-cache-aligned data to the frame buffer is not new (e.g. the 8 by 8 display [15]), it was a long time ago developers had to consider aspects such as these when developing applications for stationary PCs.

3.3 Display Orientations

While the displays on mobile devices are stored internally in various orientations, they are typically presented to the developer in a portrait orientation (where screen width is less than screen height). Many applications however require a rotated landscape mode to operate correctly (where the screen height is less than screen width). Many video cards on stationary PCs support display rotations in hardware, and simply present the developer with a display with a different aspect ratio – the display is accessed using the same display format, and the video card does the final rotation transparently. Supporting display rotations on mobile devices however requires taking both the visual display rotation and the internal frame buffer orientation into consideration. This means that all image pre-rendering and all graphics operations must consider both of these aspects for optimum performance.

4. THE GAPIDRAW PLATFORM

GapiDraw was initially created for educational use. In the fall of 2001, the primary author was responsible for a course in mobile application development. The goal of the course was to teach

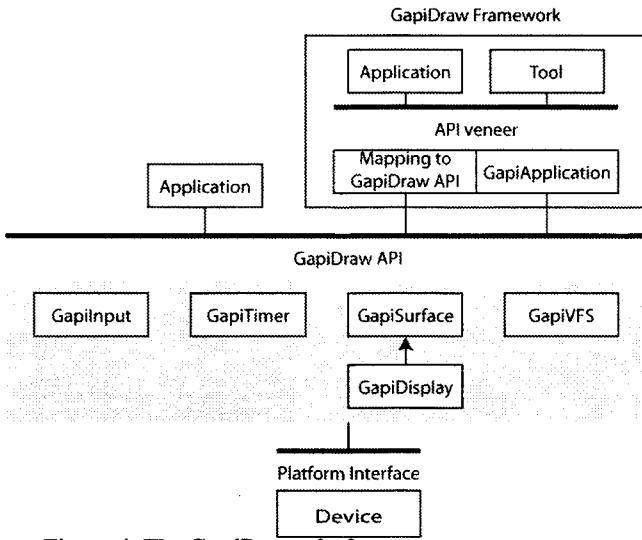


Figure 4: The GapiDraw platform.

students write networked interactive applications for handheld devices. To assist the students in their work two software platforms were created: the networking platform *OpenTrek* [13], and the graphics platform *GapiDraw*. It was not until March 2002 that *GapiDraw* was released on the Internet, and it was instantly picked up by a variety of software developers.

GapiDraw has two main components (as seen in Figure 4): a cross-platform Application Programming Interface (API) and a cross-platform Framework. The *GapiDraw API* comprises several classes for accessing the display, loading and manipulating images, and applying real-time effects such as opacity. The *GapiDraw API* was designed to be similar to the DirectX API in use, so that existing applications for stationary PCs could be transferred to mobile devices with a minimum of work, and vice versa. Some of the commercial games that have been ported from a stationary PC to a mobile device using *GapiDraw* are *Warlords II* and *Atlantis – Redux*. The *GapiDraw Framework* contains a single base class *GapiApplication*, which initiates full screen mode, enables application switching, captures messages from the operating system, and forwards button and stylus input to the application. Using *GapiApplication*, all device-specific logic is hidden beneath a cross-platform interface.

4.1 GapiDraw Components

The various components of *GapiDraw* are packaged as classes in a dynamically or statically linked library, depending on the target device. We will now describe the various components of *GapiDraw* and motivate the design choices made during the implementation process.

4.1.1 GapiApplication

GapiApplication is a cross-platform application shell that is used as an interface between the physical device and applications created with the *GapiDraw* platform. *GapiApplication* contains code for retrieving necessary information about the hardware configuration and pass this on to the necessary functions in the *GapiDraw API*. An application created with *GapiApplication* does not have to consider issues related to device hardware or operating systems, and *GapiApplication* can thus be used to enable cross-

platform prototyping on mobile devices. *GapiApplication* captures button, keyboard and stylus input and passes them on to the application in a cross-platform format. *GapiApplication* initializes full screen mode on the device and locks all hardware keys for exclusive access. *GapiApplication* also captures operating system events (such as those received when the device is switched on and off) and correctly notifies the operating system when the application requests to be minimized or otherwise wants to interact with the operating system.

The use of an application framework such as *GapiApplication* is commonly used by other graphic libraries for cross-platform development, such as GLUT [11] for Open GL development. *GapiApplication* however is the first application framework for creating cross-platform applications for mobile devices such as Palm, Symbian and Windows Mobile, not including interpreting language environments such as Java.

4.1.2 GapiInput

GapiInput captures all button events and forwards them to the application. Key codes are transformed to a common format across all devices, and they are also mapped to the current display rotation of the device. For example, if the display is rotated 90 degrees counter clockwise, if the user presses the “Up” key it will be reported to the application as the “Right” key.

Even though *GapiInput* uses a cross-platform API, there are still several cross-platform considerations that have to be done by the application developer. Many mobile devices do not have touch displays (e.g. most mobile phones), some devices do not have any buttons (e.g. the Sony Ericsson P900 only have a navigation stick in full screen mode), and some more advanced devices have it all – lots of buttons, touch displays, and even analog button devices (such as the Tapwave Zodiac). The variations in input options can in some cases make it impossible to create cross-platform versions of a mobile application, even though it is technically feasible.

4.1.3 GapiSurface

Images in *GapiDraw* are stored in *surfaces*. A surface in *GapiDraw* is an object, comprising a set of functions for initialization and image manipulation, and a memory area located in either system memory or video memory (depending on the hardware capabilities of the device). Images can be loaded from files (either physical files or files stored in the *GapiVFS* virtual file system), an image resource embedded in the executable file, or from an image located in system memory. To simplify development and to enable prototyping, the surface object in *GapiDraw* comprises a wide range of features, such as copying surfaces with variable opacity (alpha blend), drawing tools such as rectangles and lines, font tools, and real time rotation and scaling. If the proper video hardware is available to accelerate surface operations, *GapiDraw* will use this video hardware seamlessly. *GapiDraw* surfaces support clipping in all operations (see e.g. [7]), either to the entire surface or to a specified rectangle. *GapiDraw* stores all surfaces in a native format matching the frame buffer color depth, the frame buffer orientation and the display rotation of the device – all images are automatically color-converted and rotated as necessary when they are loaded.

One design feature of *GapiSurface* is that it is possible to subclass it to re-use its features. One such subclass is *GapiDisplay*, allowing all drawing tools available in *GapiSurface* to be used to draw graphics directly to the device frame buffer. Other examples of subclasses available in the *GapiDraw API* are

GapiMaskSurface, *GapiBitmapFont* and *GapiCursor* (not shown in figure 4). *GapiMaskSurface* is a surface specifically created for Z-buffer based collision masks. This type of collision masks are typically used in 2D adventure games. *GapiBitmapFont* is a bitmapped font class. Since *GapiBitmapFont* is a *GapiSurface* subclass, developers can change the font in real time using all of the drawing tools available. *GapiCursor* is an animated, alpha blended cursor class for applications that run on devices without a touch screen.

4.1.4 *GapiDisplay*

GapiDisplay is a subclass to *GapiSurface* and provides a cross-platform interface to the frame buffer of the device. Where on stationary PCs it is possible to configure the display mode such as color depth and double buffer mode, mobile devices all use different display modes and it is in most cases not possible to change the display configuration on a specific device programmatically. *GapiDisplay* thus supports all variations in display configurations and simply reports back to the application what modes are available (such as direct frame buffer access). *GapiDisplay* also manages surfaces stored in video memory, and provides support for many of the more proprietary device features, like the possibility to automatically resize the display from 320x320 to 320x480 on the Palm Tungsten T3 handheld computer.

4.1.5 *GapiTimer*

GapiTimer serves two purposes – first it can be used to synchronize the updates to the display to the vertical blanking period of the device, secondly it can be used to limit the number of frame updates each second for the application. Limiting the number of frame updates each second is necessary to improve battery life of the mobile device. If a mobile application continuously updates the display as many times as possible each second, the processor usage would run at a constant 100% and the batteries would be drained quickly. On devices having a timer with a resolution of 1 millisecond or better, *GapiDraw* provides the option to limit the maximum number of frame updates each second to save batteries. The *GapiTimer* implementation is similar to the timer included with the *SDL* graphic library [16], in that it analyzes the time all previous frames have taken to render and then calculates how long it should wait in milliseconds before rendering the next frame. If the last frame took longer to render than allowed, *GapiTimer* notifies the application, skips the next frame and resets the frame time history.

4.1.6 *GapiVFS*

Some mobile devices (such as devices manufactured by Palm and Tapwave) do not include a hierarchical file system. On other devices with a “real” file system, file management differs greatly between devices, making it difficult to distribute applications on multiple hardware platforms. *GapiVFS* is a cross-platform virtual file system that stores all images, sounds and other resources in one single database file. This database can then be included either as a resource (on Palm, the database is automatically split into multiple 32kb resources that are automatically merged by *GapiVFS*) or as an actual file, depending on the device. *GapiVFS* supports folders and subfolders, and files stored in the virtual file system can be individually compressed using zip compression to

```
// One loop for all flag combinations
<loop through all rows in surface>
{
  <loop through all pixels in current row>
  {
    <if predicate(pixel)>
      <pixeloperation(sourcecopy(pixel))>
    }
  }
}
```

Figure 5: Syntax code for a pixel loop in *GapiDraw*.

save memory storage (*GapiVFS* automatically decompresses these files when they are loaded).

5. IMPLEMENTATION

The *GapiDraw* source code is written in C++ and assembler. The source code is split into two parts: one large common part for all devices, and one minor, customized part for each unique operating system. *GapiDraw* uses an optimization technique called *template meta programming*. Template meta programming is a C++ programming technique that allows optimizations such as loop-expansion and automatic means of creating temporary variables in optimized code. Template meta programming is a recently introduced technique [17], and has previously been used in class libraries such as *Blitz++* [2]. *GapiDraw* uses the template meta programming technique for loop-expansion. In most graphics operations, *GapiDraw* accepts several flag parameters that changes how pixels are to be copied between surfaces. Examples are transparency, color mask and colorization. Implementing support for three flags, a developer can write either one loop that covers all flags, or eight individually optimized loops, where each single loop is optimized for one unique flag combination. Writing eight different loops to support three flags, means having to write another eight loops to support an additional frame buffer color depth. With *GapiDraw* accepting up to eight different flags as parameters to some operations, and supporting four different frame buffer color depths (12-bit, 15-bit, 16-bit and 16bit PalmOS5), writing individually optimized loops for each unique flag combination proved to be impossible. Writing one loop that supports all possible flag combinations by itself however imposes significant performance issues, in that all flag combinations have to be checked for every single pixel being copied.

GapiDraw’s pixel loops are created with template meta code and comprise three virtual function calls that are inlined by the pre-compiler: *<predicate>*, *<pixel operation>*, and *<source copy>*. By adding a unique call to the same loop for each flag combination (and changing the predicate function, pixel operation function, and source copy function for each call), the pre-compiler expands the code for all unique flag combinations before passing them on to the compiler. The end result is that each single pixel loop only needs to be written once, and is then automatically expanded into $(2^n)^i$ unique loops, where n is the number of unique flag variations, and i is the number of frame buffer formats supported. A typical *GapiDraw* loop is shown in Figure 5. *Predicate* determines if the source pixel should be copied or not (for example, if pixels matching a certain color should be ignored). *Pixel operation* declares how the source pixel should be blended with the destination (for example using a pixel shader such as an alpha blend). *Source copy* defines what will be copied to the destination, such as the source pixel or a pre-defined color

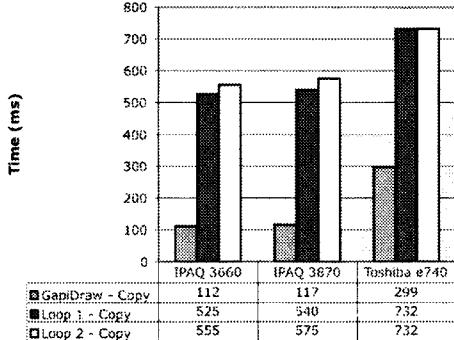


Figure 6: Time to copy 1000 images to the display.

value. If the loop in Figure 5 takes eight flags as an argument and supports two different color depths (12-bit and 16-bit), the pre-compiler expands it into 512 separately optimized loops at compile-time.

6. PERFORMANCE TEST

To test the graphics performance of GapiDraw we wrote a simple test application using code samples from the book Pocket PC Game Programming [8]. The samples in the book do not consider frame buffer rotations and do not use template meta programming. We wanted to test two aspects of drawing graphics to the display: First, what difference in performance can one expect from not considering the frame buffer orientation of a mobile device? Secondly, how much does the use of template meta programming affect performance?

The test application we implemented copies a bitmapped logo (128x40 pixels) to the display, once using GapiDraw and then using simple pixel-by-pixel loops. The application was tested on three different Pocket PC devices – all with different frame buffer orientations (an IPAQ 3630, an IPAQ 3870, and a Toshiba e740). The display of the Toshiba e730 is not rotated, the IPAQ 3630 display is rotated 90 degrees counter clockwise, and the display of the IPAQ 3870 is rotated 90 degrees clockwise.

The GapiDraw part of the test application was created by extending the GapiDraw Framework. The image logo was loaded into a surface, and then simply copied 1000 times to the display using the GapiDraw API. The test was run ten times on the three devices to get an average time to copy the images. To create the pixel-by-pixel loops the image was first loaded into a memory area and converted to the frame buffer color depth. Two separate loops were then implemented. The first loop copies the image to the display pixel-by-pixel. The second loop takes a Boolean value as a parameter. If the value is true, the image will be copied to the display, if the value is false, nothing will be done. The second loop will show the performance difference of using template meta programming for loop-unrolling, when comparing it in performance to the first loop. Pixels are copied on a row-by-row basis: for every row ($0 \leq n < \text{height}$), all pixels are copied ($0 \leq n < \text{width}$).

6.1 Test Result

The test result is seen in Figure 6 and shows the performance difference in ms of copying an image aligned to the frame buffer orientation, versus not considering the rotation of the frame buffer. Simply by pre-rotating all images to match the frame



Figure 7: Students playing the game *Pac-Man Must Die*, created in three weeks with GapiDraw.

buffer orientation, GapiDraw can copy up to two pixels simultaneously using 32-bit data reads and writes. The performance difference of copying an image aligned to the display using GapiDraw is up to 4.7 times faster than copying the image unaligned (using the first pixel-by-pixel loop on the IPAQ 3630 device). The option flag introduced in the second loop decreases performance up to 6.5% as compared to the first loop when the test application is run on the IPAQ 3870 device.

The performance decrease of adding an option flag and not using template meta programming in this test might not seem that significant (the difference was up to 6.5%). This performance decrease however scales exponentially with the number of flags added – meaning that four flags would slow down loop 1 up to 104%. By using template meta programming, the performance in GapiDraw is not affected by the number of flag combinations exposed, which in some operations can include up to eight variations.

7. GAPIDRAW AS AN ENABLER PLATFORM

GapiDraw was designed to enable cross-platform application prototyping on mobile devices. As such, it has been widely used in numerous university courses, research projects, and commercial game development projects.

7.1 Education

For two years we held a course in software development on mobile devices at a local university. Using our platforms GapiDraw and OpenTrek [13], the students were instructed to create games that required people to collaborate to succeed. The students were assigned only three weeks to implement the games. A total of 12 games were created each year, where the students worked in groups of 2-4 people on each game.

One of the games created by the students was *Pac-Man Must Die*. The game is played similar to the traditional game *Pac-Man*, where the player needs to collect dots in a labyrinth while avoiding enemies. However, some of the dots are located on the displays of other players' devices! The player can enter another person's handheld display by using "doors" at the edges of the map. When a player has entered the display of another computer she has to look at the other user's display to control her ghost (as seen in Figure 7). The game was recently tested in a use study [14], where players quickly found out that they could run away

with their displays, preventing other people from controlling their ghosts when they are at another person's display.

7.2 Research

GapiDraw is used by several researchers for application prototyping on mobile devices. Examples of such applications are *Tilt and Feel*, *PlaceMemo*, *Slide Scroller*, and *Total-Recall*. The Tilt and Feel project [10] explores the use of a mobile device augmented with a tilting sensor and a vibrotactile transducer. Using GapiDraw, the project group created several sample applications during the design phase of the project, including a tilt-driven maze game, an address book and a map application. PlaceMemo [5] runs on mobile devices and uses GPS positioning to allow road inspectors to connect voice notes to a geographical location. Using GapiDraw, the researchers implemented a navigational tool that allowed real-time zooming and manipulation of a map of the user's current location. The Slide Scroller prototype [6] uses a mobile device augmented with an optical mouse sensor. By "scrolling" the device, it is possible to navigate large documents such as web pages on the small display. In the project, researchers used GapiDraw to test several application concepts on the actual device in the design phase. Finally, in the Total Recall project [9], handheld computers are augmented with an ultra-sonic positioning system to introduce a new way to view captured whiteboard annotations – in place, where they were drawn. Using GapiDraw, the first Total Recall test prototype running on an actual mobile device was created in just one day.

7.3 Commercial games

GapiDraw has been used in more than 100 commercial games for handheld computers, including titles such as *EverQuest for the Pocket PC* by Sony Online Entertainment, *Warlords II* by Pocket PC Studios, and *Atlantis – Redux* by TetraEdge / DreamCatcher Europe. Due to the increased number of companies that used GapiDraw commercially, the GapiDraw project was moved to a separate company in May 2004 to manage sales and support.

8. CONCLUSION AND FUTURE WORK

We have presented the GapiDraw platform, which supports the creation of cross-platform applications with high-performance graphics on mobile devices. GapiDraw implements numerous optimization techniques for the heterogeneous hardware designs of mobile devices, while at the same time providing an easy to use cross-platform API and an extensible framework suitable for prototyping. Based on the extensive use of the platform in commercial applications, educational settings and research projects, we argue that GapiDraw can play an important role as an enabler platform to implement and evaluate new application concepts for mobile devices, on actual mobile devices. Future work related to the platform has already begun, with current focus on exploring new ways to make mobile 3D graphics more accessible for prototyping and educational use.

9. ACKNOWLEDGEMENTS

This research is funded by the Swedish Research Institute for Information Technology (SITI), and the Mobile Services project financed by the Foundation for Strategic Research (SSF). "Pac-Man" is a registered trademark of Namco, Inc.

10. GAPIDRAW DOWNLOAD

The GapiDraw platform and documentation can be freely downloaded from the web at: www.gapidraw.com

11. REFERENCES

1. Bechtolsheim, A. and Baskett, F.: High performance raster graphics for microcomputer systems. *Computer Graphics* 14, 3 (July 1980), pp 43-47.
2. Blitz++, <http://www.oonumerics.org/blitz/>
3. DirectX, <http://www.microsoft.com/directx/>
4. Droneship Software, <http://www.droneship.com/>
5. Esbjörnsson, M. and Juhlin, O. PlaceMemo - Supporting Mobile Articulation in a Vast Working Area through Position Based Information. In proceedings of the *European Conference on Information Systems*, 2002, Gdańsk, Poland.
6. Fallman, D., Lund, A., and Wiberg, M.: Inside-Out Interaction: An Interaction Technique for Dealing with Large Interface Surfaces such as Web Pages on Small Screen Displays, to be presented at *SIGGRAPH 2004, Sketches program*, Los Angeles, USA.
7. Foley, J. D., van Dam, A., Feiner, S. K., and Hughes, J.: *Computer Graphics: Principles and Practice*, second edition, Addison-Wesley, 1990.
8. Harbour , J. S.: *Pocket PC Game Programming*, Muska & Lipman / Premier-Trade, 2001
9. Holmquist, L. E., Sanneblad, J., and Gaye, L.: Total Recall: In-place Viewing of Captured Whiteboard Annotations. In *Extended Abstracts of CHI 2003*, Fort Lauderdale, Florida, United States.
10. Oakley, I., Angesleva, J., Hughes, S., and O'Modhrain, S.: Tilt and Feel: Scrolling with Vibrotactile Display, in *Proceedings of EuroHaptics 2004*, Munich, Germany
11. Open GL, <http://www.opengl.org/>
12. Razor Graphics Engine, <http://www.tilo-christ.de/razor/>
13. Sanneblad, J. and Holmquist, L. E.: OpenTrek: A Platform for Developing Interactive Networked Games on Mobile Devices, in *Proceedings of Mobile HCI 2003*, Udine, Italy
14. Sanneblad, J. and Holmquist, L. E.: Why is everyone inside me?! Using Shared Displays in Mobile Computer Games, to be presented at the *3rd International Conference on Entertainment Computing, 2004*, Eindhoven, The Netherlands.
15. Sproull, R. F. and Sutherland, I. E.: The 8 by 8 Display. *ACM Transactions on Graphics*, Vol. 2, No. 1, January 1983, pp 32-56.
16. The Simple DirectMedia Layer, <http://www.libsdl.org/>
17. Veldhuizen, T. and M. E. Jernigan. Will C++ be faster than Fortran? In proceedings of the *International Scientific Computing in Object-Oriented Parallel Environments*, 1997, Marina del Rey, California, United States.

Middleware Design Issues for Ubiquitous Computing

Tatsuo Nakajima, Kaori Fujinami, Eiji Tokunaga, Hiroo Ishikawa
Department of Computer Science
Waseda University
tatsuo@dcl.info.waseda.ac.jp

ABSTRACT

Our daily lives will be dramatically changed by embedded small computers in our environments. The environments are called *ubiquitous computing environments*. To realize the environments, it is important to reduce the cost to develop ubiquitous computing applications by encapsulating complex issues in middleware infrastructures that are shared by various applications.

In this paper, we describe three middleware infrastructures for supporting ubiquitous computing, that have developed in our projects. Our infrastructures have tried to hide some complexities to make it easy to develop ubiquitous computing applications in an easy way. We also show some lessons learned in our projects.

Keywords

Ubiquitous Computing, Middleware Design

1. INTRODUCTION

Our daily lives become more and more complex every day. Information technologies have been increasing these complexities, because a large proportion of our daily lives is currently spent in analyzing various sorts of information. Ironically, present ubiquitous computing technologies will increase the amount of such information dramatically, and increase complexities in our daily lives. A variety of appliances surrounding us rapidly become commodities. Today, it is very difficult to create an appliance that offers special, distinctive features. For example, we cannot distinguish among different vendor's televisions. Therefore, it is important to take into account pleasurable experiences when a user uses the appliances[14].

These devices and appliances should be integrated to work together, and it is important to develop many attractive services and applications. However, it is not easy to develop ubiquitous computing applications on existing software infrastructures currently since we need a variety of knowledge

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MUM '2004 October 27-29, College Park, Maryland, USA
Copyright 2004 ACM 0-58113-981-0/04/10 ...\$5.00.

to develop them. We believe that middleware infrastructures for ubiquitous computing that hide a variety of complexities such as distribution and context-awareness are important to make it easy to develop ubiquitous computing applications.

In this paper, we present overviews of three middleware infrastructures that we have developed. We have considered the following issues during the design and implementation of our systems.

- Which abstraction is appropriate ?
- How to hide complexities in ubiquitous computing environments?
- How to reduce the development cost of middleware ?

Our middleware infrastructures offer high level abstraction for building specific application domains to hide complexities such as distribution and context-awareness. We report how to offer high level abstraction and to implement non functional properties hidden in the middleware infrastructures. We also discuss how to implement them on standard infrastructure software and protocols to make it easy to develop our systems. Finally, we show what we have learned during their design and implementation.

The remaining of the paper is structured as follows. Section 2 describes three middleware infrastructures for ubiquitous computing. In Section 3, we show six lessons learned for building our middleware, and Section 4 concludes the paper.

2. MIDDLEWARE INFRASTRUCTURE FOR UBIQUITOUS COMPUTING

This section describes three middleware infrastructures that have developed in our project. These middleware infrastructures do not offer generic services for building ubiquitous computing applications. They support to develop applications for specific domains to realize ubiquitous computing visions.

2.1 Middleware for Mixed Reality

2.1.1 Design Issues

The middleware infrastructure described in this section allows us to build distributed mixed reality applications in an easy way. When designing the middleware, we take into account the following issues.

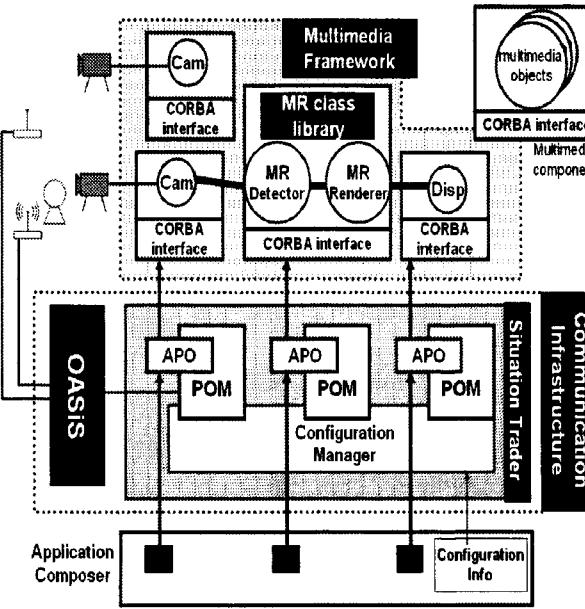


Figure 1: Overview of MiRAGE Architecture

- An application programmer should not take into account complex algorithms for implementing mixed reality applications.
- Distribution should be hidden in the middleware.
- Dynamic reconfiguration according to the current situation should be hidden from a programmer.

Our middleware infrastructure makes it dramatically easy to develop mixed reality applications for ubiquitous computing by composing existing multimedia components, and the connections among the components are reconfigured when the current situation is changed, but the change is not aware from programmers.

2.1.2 Basic Architecture

Our middleware infrastructure called MiRAGE[16] consists of the *multimedia framework*, the *communication infrastructure* and the *application composer*, as shown in Figure 1. The multimedia framework is a CORBA(Common Object Request Broker Architecture)-based component framework for processing continuous media streams. The framework defines CORBA interfaces to configure multimedia components and connections among the components. In the figure, each circle means a multimedia component that can be controlled through the CORBA interface.

Multimedia components supporting mixed reality can be created from the MR class library. The library contains several classes that are useful to build mixed reality applications. By composing several instances of the classes, mixed reality multimedia components can be constructed without taking into account various complex algorithms realizing mixed reality.

The communication infrastructure based on CORBA consists of the *situation trader* and *OASiS*. The situation trader is a CORBA service that supports automatic reconfiguration, and is colocated with an application program. It contains Adaptive Pseudo Objects(APO), Pseudo Object Man-

agers(POM), and a configuration manager that are used for dynamic configuration of multimedia components, that are described in Section 2.1.4. Its role is to manage the configuration of connections among multimedia components when the current situation is changed. OASiS is a context information database that gathers context information such as location information about objects from sensors. Also, in our framework, OASiS behaves like a naming and trading service to store objects references. The situation trader communicates with OASiS to detect changes in the current situation.

Finally, the application composer, written by an application programmer, coordinates an entire application. A programmer needs to create several multimedia components and connect these components. He specifies a policy on how to reconfigure these components to reflect situation changes. By using our framework, the programmer does not need to be concerned with detailed algorithms for processing media streams because these algorithms can be encapsulated in existing reusable multimedia components. Also, distribution is hidden by our CORBA-based communication infrastructure, and automatic reconfiguration is hidden by the situation trader service. Therefore, developing mixed reality applications becomes dramatically easy by using our framework.

2.1.3 Multimedia Framework

The main building blocks in our multimedia framework are software entities that internally and externally stream multimedia data in order to accomplish a certain task. We call them *multimedia components*.

A multimedia component consists of a CORBA interface and one or more *multimedia objects*. Our framework offers the abstract classes **MSource**, **MFilter** and **MSink**¹. Developers extend the classes and override the appropriate methods to implement functionality. Multimedia objects need only to be developed once and can be reused in any components. For example, Figure 1 shows three connected components. One component contains a camera source object for capturing video images, one component contains the **MRDetector** and **MRRenderer** filter objects for implementing mixed reality functionality, and one component contains a display sink object for showing the mixed reality video images.

In a typical component configuration, video or audio data are transmitted between multimedia objects, possibly contained by different multimedia components, running on remote machines. Through the CORBA interface defined in MiRAGE connections can be created in order to control the streaming direction of data items between multimedia objects.

2.1.4 Communication Infrastructure

A *configuration manager*, owned by the situation trader, manages stream reconfiguration by updating connections between multimedia objects. Complex issues about automatic reconfiguration are handled by the situation trader and they are hidden from application programmers. The situation trader is linked into the application program.

In our framework, a proxy object in an application composer refers to APO, managed by the situation trader. Each

¹In Figure 1, *Cam* is an instance of **MSource**, **MRDetector** and **MRRenderer** are instances of **MFilter**, and *Disp* is an instance of **MSink**.

APO is managed by exactly one POM that is responsible for the replacement of object references by receiving a notification message from OASiS upon a situation change.

A reconfiguration policy needs to be set for each POM. The policy is passed to OASiS through the POM, and OASiS selects the most appropriate target object according to the policy. In the current design, we can specify a location parameter as a reconfiguration policy. The policy is used to select the most suitable multimedia component according to the current location of a user.

A configuration manager controls the connections among multimedia components. Upon situation change, a callback handler in the configuration manager is invoked in order to reconfigure affected streams by reconnecting appropriate multimedia components.

2.1.5 Mixed Reality Class Library

The **MR class library** is a part of the MiRAGE framework. The library defines *multimedia mixed reality objects* for detecting visual markers in video frames and superimposing graphical images on visual markers in video frames. These mixed reality multimedia objects are for a large part implemented using the ARTToolkit[2]. Application programmers can build mixed reality applications by configuring multimedia components with the mixed reality objects and stream data between them. In addition, the library defines data classes for the video frames that are streamed through the mixed reality objects.

MRFiler is a subclass of **MFilter** and is used as a base class for all mixed reality classes. The class **MVideoData** encapsulates raw video data. The **MRVideoData** class is a specialization **MVideoData** and contains a **MRMarkerInfo** object for storing information about visual markers in its video frame. Since different types of markers will be available in our framework, the format of marker information must be defined in a uniform way.

The class **MRDetector** is a mixed reality class and inherits from **MRFiler**. The class expects a **MVideoData** object as input and detects video markers in the **MVideoData** object. The class creates a **MRVideoData** object and adds information about detected markers in the video frame. The **MRVideoData** object is transmitted as output. The class **ARTkDetector** is a subclass of **MRDetector** that implements the marker detection algorithm using the ARTToolkit.

The **MRRenderer** class is another mixed reality class derived from **MRFiler**. The class expects an **MRVideoData** as input and superimposes graphical images at positions specified in the **MRMarkerInfo** object. The superimposed image is transmitted as output. The **OpenGLRenderer** is a specialization of **MRRenderer** and superimposes graphical images generated by an OpenGL program.

2.1.6 An Application Scenario

In a typical mobile mixed reality application, our real-world is augmented with virtual information. For example, a door of a classroom might have a visual tag attached to it. If a PDA or a cellular phone, equipped with a camera and an application program for capturing visual tags, the tags are superimposed by a schedule of today's lecture.

We assume that in the future our environment will deploy many mixed reality servers. In the example, the nearest server stores information about today's lecture schedule and provides a service for detecting visual tags and superim-

posing them by the information about the schedule. Other mixed reality servers, located on a street, might contain information about what shops or restaurants can be found on the street and until how late they are open.

To build the application, an application composer uses components for capturing video data, detecting visual markers, superimposing information on video frames and displaying them. The application composer contacts a situation trader service to retrieve a reference to a POM managing references to the nearest mixed reality server to a user. When he moves, a location sensor component notifies sensed location information to OASiS, and OASiS notifies the situation trader to replace the current object reference to the reference of the nearest mixed reality server currently. In this way, the nearest mixed reality server can be selected dynamically according to his location, but the automatic reconfiguration is hidden from an application programmer.

2.2 Middleware for Interaction Devices

2.2.1 Design Issues

Future ubiquitous computing applications will use a variety of interaction devices to control them. These devices are distributed and are changed according to a user's current situation. Since we already have many interactive applications that adopt existing GUI toolkits, we like to reuse these applications in ubiquitous computing environments. To realize the goal, we take into account the following issues when designing the second middleware.

- Existing interactive applications can be controlled by various interaction devices.
- Interaction devices can be changed according to a user's current situation.

An application programmer needs not to consider which interaction device is appropriate by hiding the complex decision into the middleware infrastructure, and he can use existing GUI toolkits to develop ubiquitous computing applications by adopting our middleware.

2.2.2 Basic Architecture

Figure 2 shows an overview of the architecture of our middleware infrastructure that is called Unit[11]. In the architecture, an application generates bitmap images containing information such as control panels, photo images and video images. These applications can receive keyboard and mouse events to be controlled. The user interface middleware receives bitmap images from applications and transmits keyboard and mouse events to the applications. The role of the middleware is to select appropriate interaction devices by using context information. Input/output events and mouse/keyboard events are converted according to the characteristics of respective interaction devices.

The application implements graphical user interface by using a traditional GUI toolkit such as the GTK+ or Qt. The bitmap images generated by the user interface system are transmitted to our middleware. On the other hand, mouse and keyboard events captured by the middleware are forwarded to the toolkit. The protocol between the middleware and the user interface system are specified as a standard protocol called a *universal interaction protocol*.

Our system consists of the following four components.

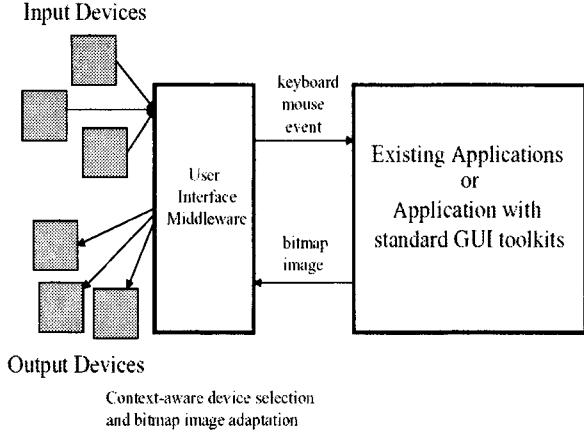


Figure 2: Basic Architecture

- Interactive Application
- Unit Server
- Unit Proxy
- Input/Output Interaction Devices

Interactive applications generate graphical user interface written by using traditional GUI toolkits. In our system, we can use any existing GUI based interaction applications, and they are controlled by a variety of interaction devices that are suitable for a user's current situation.

The *Unit server* transmits bitmap images generated by a GUI toolkit using the universal interaction protocol to a Unit proxy. It forwards mouse and keyboard events received from a Unit proxy to the GUI toolkit.

The *Unit proxy* is the most important component in our system. The Unit proxy converts bitmap images received from a Unit server according to the characteristics of output devices. The Unit proxy converts events received from input devices to mouse or keyboard events that are compliant to the universal interaction protocol. The Unit proxy chooses a currently appropriate input and output interaction devices for controlling appliances. To convert interaction events according to the characteristics of interaction devices, the selected input device transmits an input specification, and the selected output device transmits an output specification to the Unit proxy. These specifications contain information that allows a Unit proxy to convert input and output events.

The last component is *input and output interaction devices*. An input device supports the interaction with a user. The role of an input device is to deliver commands issued by a user to control interactive applications. An output device has a display device to show graphical user interface to control the interactive applications.

In our approach, the Unit proxy plays a role to deal with the heterogeneity of interaction devices. Also, it can switch interaction devices according to a user's situation or preference. This makes it possible to personalize the interaction between a user and appliances.

2.2.3 Unit Proxy

The current version of Unit proxy is written in Java, and the implementation contains four modules. The first module is the universal interaction module that executes the universal interaction protocol to communicate with a Unit server. The second module is the plug and play management module. The module collects currently available interaction devices, and builds a database containing information about respective interaction devices. The third module is the input management module. The module selects a suitable input interaction device by using the database contained in the plug and play management module. The last module is an output management module. The module also selects a suitable output interaction device. Also, the module converts bitmap images received from the universal interaction module according to the output specification of the currently selected output interaction device.

2.2.3.1 Management of Available Interaction Devices:

The plug and play management module detects currently available input and output devices according to context information. The module implements UPnP(Universal Plug and Play) to detect currently available interaction devices. An interaction device transmits advertisement messages using SSDP(Simple Service Discovery Protocol). When a Unit proxy detects the messages, it knows the IP address of the interaction device. Then, the Unit proxy transmits an HTTP (HyperText Transfer Protocol) GET request to the interaction device. We assume that each interaction device contains a small Web server, and returns an XML(eXtensible Markup Language) document.

The XML document contains information about the interaction devices. If the interaction device is an input device, the document contains various attributes about the device, which are used for the selection of the most suitable device. For an output device, the document contains information about the display size and the attributes for the device. The plug and play management module maintains a database containing all information about currently detected interaction devices.

2.2.3.2 Adaptation of Input and Output Events:

The role of the input management module and the output management module is to determine the policies for selecting interaction devices. As described in the previous paragraph, all information about currently available interaction devices are stored in a database of the plug and play management module. The database provides a query interface to retrieve information about interaction devices. Each entry in the database contains a pair of an IP address and a list of attributes for each interaction device, then the entry whose attributes are matched to a keyword provided in a query is returned.

The output management module converts bitmap images received from the universal interaction module according to the display size of an output device. The size is stored in the database of the plug and play management module. When an output device is selected, the display size is retrieved from the database. The bitmap image is converted according to the retrieved information, then it is transmitted to the selected output device.

2.2.4 An Application Scenario

An example described in this section is a ubiquitous video phone that enables us to use a video phone in various ways. In this example, we assume that a user speaks with his friend by using a telephone like a broadband phone developed by AT&T Laboratories, Cambridge. The phone has a receiver like traditional phones, but it also has a small display. When the phone is used as a video phone, the small display renders video streams transmitted from other phones. The display is also able to show various information such as photos, pictures, and HTML(Hypertext Markup Language) documents that are shared by speakers. Our user interface system makes the phone more attractive, and we believe that the extension is a attractive application in ubiquitous computing environments.

When a user needs to start to make a dinner, he will go to his kitchen, but he likes to keep to talk with his friend. The traditional phone receiver is not appropriate to continue the conversation with his friend in the kitchen because his both hands may be busy for cooking. In this case, we use a microphone and a speaker in the kitchen so that he can use both hands for making the dinner while talking with his friend. In the future, various home appliances such as a refrigerator and a microwave have displays. Also, a kitchen table may have a display to show a recipe. These displays can be used by the video phone to show a video stream. In a similar way, a video phone can use various interaction devices for interacting with a user. The approach enables us to use a telephone in a more seamless way.

Our system allows us to use a standard VoIP application running on Linux. The application provides a graphical user interface on the X window system, but our system allows a user to be able to choose various interaction styles according to his/her situation. If his/her situation is changed, the current interaction style is changed according to his preference.

2.3 Middleware for Home Computing

2.3.1 Design Issues

In the future, there are many home applications in our home environments. It is important to control these appliances in an easy way. The third middleware infrastructure called a *personal home server*[12] allows us to aggregate them by using a personal device. While designing the middleware, we take into account the following issues.

- We like to control home appliances from various presentation documents such as HTML and Flash.
- A way to control home appliances is changed according to a user's preference.

Our middleware offers high level abstraction to specify appliances that a user likes to control, and each user's personal device contains rules for personalizing the control of appliances. Since a personal home server can be carried with a user, he can aggregate home appliances by using the same preferences in a seamless way anytime anywhere.

2.3.2 Basic Architecture

A personal home server that is carried by a user is implemented in a personal device like a cellular phone, a wrist watch, or a jacket. Thus, the server can be carried by a user

anytime anywhere. The personal home server collects information about home appliances near a user, and creates a database storing information about these appliances. Then, it creates an HTML-based presentation document containing the attributes of appliances and the commands to control them. A display near the user also detects the personal home server, and retrieves the presentation document containing the automatically generated user interface. The display shows the presentation document on the display. The document contains URLs(Uniform Resource Locator) embedding the attributes of appliances and their commands specified by using our URL-based naming scheme. Also, the presentation document is customized according to the user's preference. When the user touches the display, a URL containing the attributes of an appliance and its command is transmitted to his/her personal home server via the HTTP protocol. The server translates the URL to a SOAP command by accessing a database containing information about the appliance that s/he likes to control. Finally, the SOAP(Simple Object Access Protocol) command is forwarded to the target appliance.

2.3.3 Spontaneous Appliance Detection

2.3.3.1 URL-based Naming Scheme:

Our framework allows a user to access one or more appliances through a personal home server. We introduce a URL-based naming convention for specifying and controlling appliances. In our approach, by embedding the attributes of appliances and commands in URLs, an HTML-based presentation document can be used to control home appliances. The convention is defined within the standard URL but the path elements of the URL form can contain some additional information.

The URL definition is very flexible because we can specify various attributes to identify a target home appliance. We can also use attributes that represent context information such as location. A personal home server can select an appliance in a context-aware way.

2.3.3.2 Service Management:

In our system, the service management module in a personal home server knows respective appliances via SSDP, and retrieves service specification documents represented as RDF(Resource Description Framework).

The service database in a personal home server contains all service specification documents detected currently. It contains a link to a WSDL(Web Service Description Language) document identifying commands that can be accepted. If an appliance contains several functionalities, its specification document may contain several links to WSDL documents. Also, attributes of the document are used to identify a target appliance.

2.3.3.3 Personalization Management:

A personal home server customizes a presentation document according to a user's preference encoded in a preference rule. Now, a personal home server detects several types of light appliances. We assume that a rule to filter light alliances whose type is not a ceiling light is stored in the personal home server. The presentation document contains only ceiling lights that reside in a room where a user is. A personal home server omits information about other

types of light appliances. The preference rule is encoded in a tag, and it can be registered in a personal home server by closing the tag to a user's personal device[13].

2.3.4 An Application Scenario

In this scenario, we assume that surrounding various objects will embed tags. Since these tags contain different preference rules, the customization is changed according to objects near a user. For example, if a child holds a stuffed animal that contains tags, the rules encoded in the tags are registered in the child's personal coordination server. S/he can customize how to control information appliances by changing stuffed animals that s/he holds currently.

For example, we assume that a child is holding a stuffed toy dog. The dog contains a tag including a rule for selecting televisions because the child believes that the dog likes to watch televisions. Thus, a display shows a user interface to control televisions. On the other hand, when the child holds a stuffed toy rabbit, the display shows a user interface for music players because she believes that the rabbit likes to listen to music.

The tags can also be embedded in our daily goods like clothes, accessories, and shoes. Especially, a young person usually wants to put on these goods according to his/her feeling or emotion everyday. The goods reflect his/her preferences and current mental condition either consciously or unconsciously. Customizing services depending on what a user puts on today makes his/her daily lives more pleasurable.

3. DISCUSSIONS

The section describes some experiences while building our middleware infrastructures, and identifies six lessons learned from the experiences.

3.1 High Level Abstraction and Middleware Design

One of the most important design issues for building middleware infrastructures for ubiquitous computing is what properties the middleware infrastructures should hide. In our first middleware, distribution and automatic reconfiguration are hidden from a programmer. In the second middleware, the automatic selection of interaction devices is hidden from an application programmer. The last middleware hides device discovery and personalization from a programmer. Our experiences show that hiding these complex issues makes application programming dramatically easy. However, achieving complete distribution transparency is very hard to be implemented because different abstractions require different ways to hide the distribution. Each abstraction may also have different assumptions and constraints to hide dynamic reconfiguration. It is not easy to hide these properties in a common infrastructure that can be shared from various middleware for ubiquitous computing, and we need to carefully consider how to hide the properties in each middleware infrastructure.

High level abstraction supporting a specialized application domain is very useful to develop ubiquitous computing applications easily. The similar conclusion is discussed in a middleware infrastructure for supporting synchronous collaboration in an office environment[15], and a middleware infrastructure for building location-aware applications[6]. In our approach, the first middleware focuses on supporting

continuous media applications. The second middleware supports interactive applications that are controlled by a variety of interaction devices. The third middleware makes it easy to aggregate home appliances in a spontaneous way. These middleware supports only specialized application domains, and their functionalities do not overlap each other. Thus, they can coexist to develop one application. For example, we can use the second middleware to control an mixed reality application implemented on the first middleware. Also, the third middleware can be used to discover multimedia components implemented on the first middleware dynamically.

Lesson 1: It is important to develop specialized high level abstraction for supporting one specific domain, but the abstraction should be generic to cover a wide range of applications.

Middleware infrastructures for ubiquitous computing need to offer various non-functional properties such as context-awareness, timeliness, reliability and security. For example, the first and second middleware infrastructures automatically change the configurations according to a user's situation, and the third middleware infrastructure generates a graphical user interface automatically according to the currently available appliances. However, it is not easy to offer a common service for supporting context-awareness because modeling our real world for building any ubiquitous computing applications require to define ontologies in a complete way. Therefore, we suspect that we can implement a generic and reusable high level service to support context-awareness that is used in various middleware for ubiquitous computing. On the other hand, we believe that a low level support for handling context information like [4] can be used uniformly in many middleware infrastructures. This means that it is desirable to hide the details of the behavior of sensors in a component to develop middleware infrastructures in an easy way, but the middleware should not interpret the meaning of the value retrieved from sensors because there is no common consensus how to model our world in a standard way.

Also, we found that it is not easy to offer a common service for adding security in each middleware infrastructure. In the second system, a system needs to register their interaction devices before using them manually, but after registering them, the devices can be changed according to a user's situation automatically. In the third system, a very light-weighted security support is implemented to make the system simple[12]. We found that each middleware infrastructure requires a customized security support because each application domain may have different requirements for supporting security.

We believe that future middleware infrastructures should offer a variety of non-functional concerns such as security, timeliness, privacy protection, trust relationship among people, and reliability. The generic support of the concerns may make the infrastructures too big and complex. Especially, when multiple middleware infrastructures are integrated, the concerns are cross-cut across them. Therefore, it is important to support only minimum supports, and customized supports for non-functional properties should be implemented respectively using software structuring techniques like an aspect-oriented programming technique[9] on the minimum supports.

Lesson 2: Common generic services that offer non-functional concerns may make it difficult to develop practical ubiquitous computing applications. The service should be customized in respective middleware infrastructures.

3.2 Development of Middleware Infrastructures

The third middleware has adopted standard protocols such as SOAP as underlying protocols. This makes it easy to adopt commercial products in our experiments, and the approach is very important for incremental evolution of ubiquitous computing environments. However, devices and appliances in smart environments may want to change their interfaces independently. We need a more spontaneous approach to collaborate them. We believe that it is desirable not to fix interface between appliances, but to determine the message format to communicate with each other. The format may adopt the XML-based representation. In the approach, each appliance may add extra tags that offer additional functionalities. Let us assume that an appliance receives the message. If the message contains some tags that cannot be understood by the appliance, the tags can simply be ignored. Therefore, each appliance can extend the message format independently. We believe that the approach is desirable in ubiquitous computing environments to support communication among appliances.

In the first middleware, we have adopted CORBA to compose multimedia components. The middleware offers multiple interfaces to communicate between an application composer and components. The approach is useful to support multiple versions of the interfaces, and components can extend its functionalities by adding new interface.

Lesson 3: Each program should extend its interface independently without considering other programs if they are loosely coupled.

Our project has adopted Linux, CORBA, OSGi and Java as underlying infrastructures, and they reduce the development cost dramatically. We think that it is not a good approach to extend existing commodity software because this makes it difficult to replace software platforms. Our middleware exploits to use existing software and appliances. For example, in the first middleware, we use a CORBA system as an underlying infrastructure. However, we have not modified the existing CORBA runtime to support dynamic configuration. Therefore, we can use any commercial CORBA runtimes for executing our middleware. The second middleware allows us to use existing GUI-based applications as ubiquitous computing applications without modifying them. Thus, the approach allows us to use existing interactive applications in ubiquitous computing environments. Also, the third middleware adopts OSGi[3] as its component framework. Therefore, we can use standard services provided by OSGi, and we can use any OSGi components to develop new services on personal home servers.

We believe that the approach to use traditional commodity software without modifying them is very desirable to migrate to new platforms easily when old platforms will become obsolete. If we adopt a modified version of commodity software, it is difficult to promote our middleware on the new platforms. The approach is very desirable to migrate from the current environments to ubiquitous computing environments in a seamless and incremental way. For example, in

[10], we have extended CORBA to support dynamic transport protocol selection. However, we need to rewrite applications to select the most appropriate transport protocol. The approach is very useful to optimize the performance of applications on a specific environment, but the modification cost is not cheap to use existing large applications.

Lesson 4: We should not extend traditional standard middleware infrastructures to support advanced ubiquitous computing services if possible.

3.3 Human Factors

In our middleware infrastructures described in the paper, dynamic reconfiguration is hidden from a user. However, these properties are closely related to human factors. For example, if an interaction device is switched in an unexpected way, a user may surprise the context change. This means that middleware infrastructures that hide dynamic reconfiguration will need to take into account human factors when designing middleware[5]. Our experiences show that automatic selection of interaction devices is not good approach. Instead, we use a token to choose the most suitable interaction device in an explicit way. For example, let us assume that a user is using a telephone in a living room. When s/he moves to a kitchen, s/he may use a speaker and a microphone in the kitchen. In this case, the user brings a token attached to the telephone, and put it into a base in the kitchen. Our system detects the event, and changes the interaction devices for the user.

However, implicit changes may be attractive for realizing pleasurable services. For example, if an environment detects that a user and his girl friend are together in a room, it is desirable to make the room's lighting strategy more romantic automatically. Also, implicit changes are desirable if a user utilizes services without being aware. In this case, the services should not interrupt the user's current activity that he is focusing on. We believe that it is better to control the strategies for dynamic reconfiguration should be customized in each application. The programming interface to control dynamic reconfiguration should be clearly separated from other programming interface to make the structure of an application clear.

When representing personal information on a display, we need to take into account how to protect privacy information of a user. However, the information is useful to offer better services customized for the user. We found that it is important to take into account the tradeoff between the quality of services and the amount of privacy information. When a user cannot trust an application, s/he will not offer his/her personal information, but if s/he wants better services and trusts the services, s/he can offer more his/her personal information. Also, when multiple users share a display, it is desirable to abstract information represented on the display to protect privacy information. The abstraction level of the representation is determined according to how the information is secret. Ambient displays or informative art[7] is a first step towards information abstraction that allows us to protect privacy information and to reduce information overload in our society.

Lesson 5: Middleware infrastructures should be flexible to implement dynamic reconfiguration and information representation according to the requirements of respective applications.

In the near future, designers for ubiquitous computing middleware should learn psychology, and we need to consider that the adaptation of software should not contradict our mental model. We believe that how to implement the real world model and the mental model in middleware infrastructures is an important research topic for building practical middleware for ubiquitous computing. For example, in our second middleware, if choosing a suitable interaction device is not consistent with a user's mental model, the user will confuse which interaction device s/he should use. In the first and second middleware, if the dynamic configuration is not consistent with a user's mental model, the user may surprise the dynamic changes. However, the implementation of a real world model and mental model requires to represent ontologies in a standard way to access them from a variety of middleware for ubiquitous computing.

We also need to consider social aspects and cultural aspects when designing applications interacting with the real world. For example, it is important to take into account trust and privacy in future ubiquitous computing environments, but we need to learn sociology and anthropology to know whether our understanding is enough or not. We believe that it is important to consider how to model psychological, social, and anthropological concepts into our programs to interact with the real world properly when designing middleware infrastructures for ubiquitous computing. For example, in our middleware infrastructures, we use a user's location to offer context-aware services, but it depends on a user's situation to offer location information to our middleware. These information related to privacy should be treated very carefully, and traditional concepts about privacy and trust are very naive to offer practical ubiquitous computing services.

We also believe that the designers for ubiquitous computing middleware should know aesthetics to provide pleasurable services[8] or to abstract information. To develop pleasurable services, we may need to take into account emotion, peak experience, and unconsciousness to develop software. For example, our third middleware supports a mechanism to design pleasurable experiences by encoding preference rules in RF tags. Our system infrastructure allows us to embed tags into various objects and places, and controls our experiences by changing the behavior of applications[13], and we found that the approach is very useful to offer pleasurable services.

Lesson 6: Middleware infrastructures should incorporate a model for psychological, sociological and anthropological concepts explicitly.

4. CONCLUSION AND FUTURE DIRECTION

This paper has described three middleware infrastructures that have developed in our project. We have also presented several experiences and future directions for building middleware for ubiquitous computing. We believe that there are new requirements to develop the middleware infrastructures for ubiquitous computing. Especially, we believe that it is important to take into account human factors to develop them.

One of the most important future topics in our project is to develop a pattern language[1] for building middleware infrastructures for ubiquitous computing. The language will support to consider what abstraction should export, which

properties should be hidden and how to offer non-functional properties. Also, the language will help to consider how to implement middleware infrastructures in an easy way and how to use legacy software.

5. REFERENCES

- [1] C.Alexander, "A Pattern Language: Town, Building, Construction", Oxford Press, 1977.
- [2] ARToolkit, http://www.hitl.washington.edu/research/shared_space/download/.
- [3] K.Chen, L.Gong, "Programming Open Service Gateways with Java Embedded Server", Addison-Wesley, 2001.
- [4] A.Dey, G.D. Abowd, D.Salber, "A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Applications", Human-Computer Interaction, Vol.16, No.2-4, 2001.
- [5] Edwards, K., Bellotti, V., Dey, A.K., Newman, M. "Stuck in the Middle: The Challenges of User-Centered Design and Evaluation for Middleware", In the Proceedings of CHI 2003, 2003.
- [6] A.Harter, A.Hopper, P.Steggles, A.Ward, P.Webster, "The Anatomy of a Context-Aware Application", In Proceedings of Mobicom 2000, 2000.
- [7] L.E.Holmquist, T. Skog, "Informative Art: Information Visualization in Every day Environments", In Proceedings of GRAPHITE 2003, 2003.
- [8] P.W.Jordan, "Designing Pleasurable Products", CTI, 2000.
- [9] G.Kiczales, J.Lamping, A.Memdheker, C.Maeda, C.V. LOpes, J-M. Loingtier, J.Irwin, "Aspect-Oriented Programming", In Proceedings of ECOOP'97, 1997.
- [10] T.Nakajima, "Practical Explicit Binding Interface for supporting Multiple Transport Protocols in a CORBA system", In Proceedings of ICNP'00, 2000.
- [11] Tatsuo Nakajima, et. al., "Making Existing Interactive Applications Context-Aware", In Proceedings of Europar 2003, 2003.
- [12] T.Nakajima, I.Satoh, "Personal Home Server: Enabling Personalized and Seamless Ubiquitous Computing Environments", In Proceedings of Percom2004, 2004.
- [13] T.Nakajima, "A Personalization Framework in a Personal Home Server: System Infrastructure for Designing Pleasurable Experiences", To be submitted, 2004.
- [14] B.J.Pine II, J.H. Gilmore, "The Experience Economy", High Bridge Company, 1999.
- [15] P.Tandler, "The BEACH Application Model and Software Framework for Synchronous Collaboration in Ubiquitous Computing Environments", Journal of Systems and Software, October, 2003.
- [16] Eiji Tokunaga, Tatsuo Nakajima, et. al., "A Middleware Infrastructure for Building Mixed Reality Applications in Ubiquitous Computing Environments", In the Proceedings of MobiQuitous2004, 2004.

Plug-and-Play Application Platform: Towards Mobile Peer-to-Peer

Erkki Harjula¹, Mika Ylianttila¹, Jussi Ala-Kurikka¹, Jukka Riekki², Jaakko Sauvola¹

¹ MediaTeam Oulu Group, ² Intelligent Systems Group,

Department of Electrical and Information Engineering, Erkki Koiso-Kanttilankatu 3,
FIN-90014 University of Oulu, Finland
Tel. +358 8 553 1011

erkki.harjula@ee.oulu.fi, mika.ylianttila@ee.oulu.fi, jussi.ala-kurikka@ee.oulu.fi,
jukka.riekki@ee.oulu.fi, jaakko.sauvola@ee.oulu.fi

ABSTRACT

While peer-to-peer (P2P) has emerged as a new hot communication concept among the Internet users, mobile usage of P2P applications is still taking its first steps. This article first elaborates the evolutionary process that P2P architectures are going through. Challenges and requirements for mobile P2P are then identified, followed by a definition of a novel Plug-and-Play Application Platform (PnPAP). This platform enables dynamic selections between diverse P2P and session management protocols while preserving the best available network connectivity through Holistic Connectivity (HCon) management. On-the-fly reconfiguration and run-time parameter optimization can be done with a lightweight interpretable state machine. The concept enables flexible and seamless communications for mobile devices in P2P networks.

Categories and Subject Descriptors

C.2.4 [Computer-Communication Networks]: Distributed Systems - *Distributed applications*.

General Terms

Design, Management, Experimentation

Keywords

Peer-to-peer, Mobile P2P, Plug-and-Play, Holistic Connectivity

1. INTRODUCTION

Recently, P2P networking has become a hot topic in the world of information technology. P2P challenges the traditional client-server model that has dominated the computer networks since it superseded the early peer-to-peer networks, e.g. ARPANET, during the 80's. The storage model of the Internet is changing from a "content located in the center" model to a "content located on the edge" model [1]. The evolution of P2P architecture comprises development from the first generation (first implementations in late 90s) via the second generation

"Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee."

MUM 2004, October 27-29, 2004 College Park, Maryland, USA.
Copyright 2004 ACM 1-58113-981-0 /04/10... \$5.00"

(commenced in 2000) to the third generation (from 2002 to present). This evolutionary process is introduced in this paper as a basis for suggestions to further development. Figure 1 categorizes the protocols introduced in this article.

The new wave of P2P architectures took place in the late 90s. New freeware file sharing systems, like Napster, were published. Also the SIP protocol [2], which can loosely be considered as the first generation of P2P, for creating and controlling real-time sessions, was standardized in 1999. These first new-wave P2P systems are, however, more like a mixture of client-server and peer-to-peer models. This denomination comes from the fact that, in these systems, many of the functions related to peer and resource discovery were made in centralized servers or server pools. In this paper, applications of this type are called first-generation P2P systems.

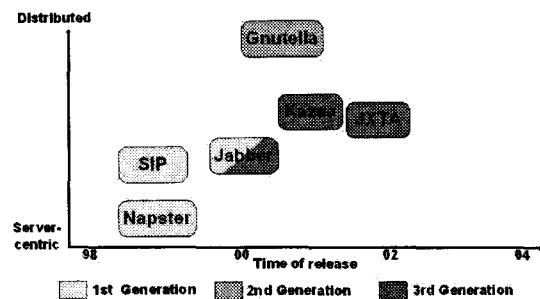


Figure 1. The evolution of P2P.

The next step in the evolution of P2P was the development of pure P2P systems, where every peer had equal functionality without any centralized servers at all. These types of P2P systems are called the second generation of P2P. In this paper, we use Gnutella [3] as an example of this generation. Second-generation P2P systems strived to solve many problems of first-generation server-centric systems. Many improvements were successful, but the new model suffered from major overhead generated by the binding messages and queries propagating around the Internet.

The current generation, the third generation of P2P, is a mixture of the first and the second generation. In third-generation systems, some of the peers are so-called super-peers. These super-peers are organized dynamically. Unlike in the earlier generations, only super-peers are used in peer and resource discovery, which significantly decreases the stress caused to the network. Also several binding and routing optimization methods are used to decrease the overhead. In this paper, we use JXTA [4] as an example of 3rd generation P2P systems.

The growing usage of mobile devices – smart phones and PDAs – and thus a need for mobile P2P applications bring up new challenges to peer-to-peer networking. Compared to desktop computers, mobile devices have many restrictions: network bandwidth, memory, processing capacity and battery life are limited. Also the network barriers, NATs (Network Address Translator) and firewalls bring challenges to mobile P2P networking. However, the evolution of mobile network technologies and -devices decreases hardware limitations, and thus makes it possible to use P2P applications also in mobile devices. Mobile environment also opens new creative ways to use P2P technology, which many seem to forget. In this article, we introduce ways to develop P2P towards a new mobile generation.

2. EVOLUTION OF P2P

2.1 The 1st generation of P2P

The first generation of P2P architectures consisted of peers and a dedicated server or a server pool for maintaining an index of the connected peers and their resources. Connections between peers were peer-to-peer connections. Napster is a typical first-generation P2P application, invented in 1999 by Shawn Fanning. Later it became very popular around the world in sharing music files between millions of users. Napster was, however, forced to shut down in 2001 due to legal issues. Later, Napster has reopened its service as a legal music net store.

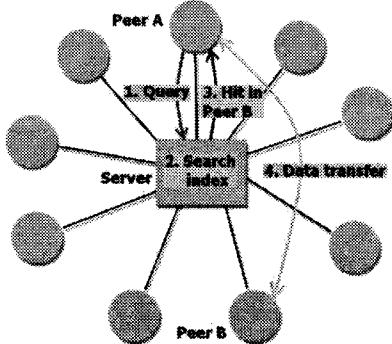


Figure 2. The operating principle of Napster.

Figure 2 illustrates the operating principle of Napster. In Napster, there was a large cluster of dedicated servers maintaining an index of the users and their shared files. After connecting to the server, peer (for example A) maintains the connection to server cluster. When the user of peer A is querying for resource, A sends the query message to the server cluster (1.). Servers in the cluster then co-operate to find the resources (2.) and return the list of matching resources to A (3.). After this, when the user chooses the resource to be downloaded, a peer-to-peer connection between A and B is established to transfer the resource (4.).

The benefits of Napster and other first-generation P2P architectures are efficiency of query processing and low overhead. However, the server-centricity leads to several problems:

- Need of maintenance
- Cost
- Vulnerability
- Poor scalability
- Legal issues

The first-generation architecture itself is feasible and a working model for mobile devices, since peer applications can be very light. However, the centralized architecture needs maintenance, which somebody has to pay for. The architecture is also vulnerable to network or hardware failures and denial-of-service attacks. Since, for example, Napster has a single point of entry, the network can completely collapse if this point goes down. The server-centric architecture is also poorly scalable for different networks. Napster, like other first-generation architectures, does not offer a NAT traversal functionality, which would be essential to enable usage in mobile devices, since mobile networks are usually in NATted networks behind firewalls.

2.2 The 2nd generation of P2P

The second-generation P2P architectures can be described as pure P2P architectures, since every peer in these architectures has equal functionalities and no dedicated servers are needed. The peer discovery and query delivery functions are distributed to all peers. Descriptively, peers of these architectures are called *Servents*. The word *servent* is constructed from words *server* and *client*. Gnutella [3] protocol is our example of second-generation architectures. It was developed by Nullsoft in the early 2000 but the project was buried on the day following its release mainly because Nullsoft's parent company AOL urged it. However, the protocol was soon reverse-engineered and published as open-source. After that, it has been widely used in many applications for several years.

When Gnutella servent (for example A) is started, it begins searching for neighbor servents. When another servent is found, A announces its availability to it, which again broadcasts A's availability to its neighbor servents. This pattern will continue recursively with each new level of nodes announcing A's availability to their neighbors. For restricting this propagation, Gnutella uses Time-to-live (TTL) fields in these broadcast messages. After every node, TTL is decreased by one.

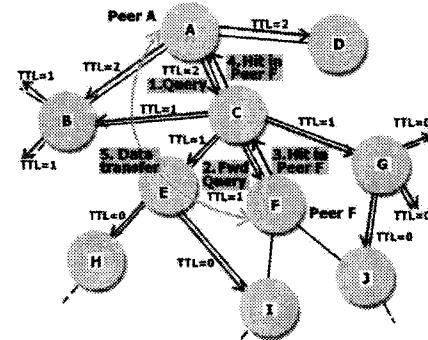


Figure 3. The operating principle of Gnutella.

Figure 3 illustrates the operating principle of Gnutella in the case of searching resources. In the beginning, A sends a query message to all its neighbor servents, B, C and D (1.). If the queried resource is not found, they again broadcast the query to all their neighbor servents, in C's case, to E, F, and G (2.). If the resource is again not found, this pattern will continue recursively with each new level of nodes, until the resource is found or TTL reaches zero. In this case, the resource is found from F. F responds to C with a hit message containing its location information (3.). C then delivers this hit message to A (4.). File

transfer is then made between A and F (5.), just like in Napster's case. In other peers, E and G, the query is forwarded again. In this case the search ends in peers H, I, and J because the TTL reaches zero in them.

Gnutella's totally decentralized, pure P2P architecture overcomes some problems of Napster's architecture. The most important advantages of the Gnutella architecture are its independency of central servers, and thus cost-effectiveness, fault tolerance and ability to work without maintenance. However, since the Gnutella protocol is based on propagating broadcast messages, it stresses networks heavily [5]. The amount of transferred broadcast messages increases exponentially with a linear increase of the depth of the search. Gnutella is also worse than Napster in search efficiency.

The second-generation protocols with their multi-function servants are obviously too heavy for use in mobile devices, since they waste the limited network capacity. Also the need of these multifunction servants for processing capacity is probably too heavy for mobile devices. NAT and firewall traversal is not typically used in second-generation architectures. However, not all second-generation protocols use the broadcast method for query propagation. There are many newer second-generation protocols, with more sophisticated search methods, for example CAN, Chord, Pastry/Tapestry and Kademlia [6]. Several models have also introduced mobile agents to enable mobile use of second-generation P2P architectures [7, 8]. This development, alongside the movement in trying to solve the other problems pointed above, has led to the development of the 3rd generation P2P architectures.

2.3 The 3rd generation of P2P

The third-generation P2P architectures strive to solve the problems of the earlier generation architectures. Fundamentally it is a mixture of the two first generations, using the structured, hierarchical peer architecture. Unlike in the first and the second-generation solutions, the third-generation architectures have two kinds of peers: *peers* and *super-peers*. Peers are light end-user peers, whereas super-peers are on a higher level in the hierarchy, working as relays for peers and other super-peers. The role of the super-peers is quite similar to the servants in the 2nd generation architectures, but the functionality is very different. Several optimization methods are used for decreasing the overhead, which was the main problem of the second-generation architectures. The regular peers, so-called *edge-peers* (peers) use super-peers as a gateway to the P2P network. As many functions as possible are left to be handled in super-peers. Super-peers are also used for NAT and firewall traversal.

Our example of 3rd generation P2Ps is open-source JXTA development framework [4]. JXTA platform was originally conceived by Sun Microsystems Inc. and designed with the participation of experts from academic institutions and industry. The development of JXTA started in 2001 and is still going on. In a nutshell, JXTA establishes a virtual network on top of the IP or non-IP networks, hiding the underlying protocols. It uses XML messages, and is thus independent of the software and hardware platform. Each JXTA peer has its own logical, network-independent ID. Peers organize automatically or manually into peer groups that are either protected private- or public groups of peers that are visible to each other. Peer group is the base unit of JXTA, and basically everything happens within them, which also considerably limits the load to underlying networks.

JXTA dynamically uses either TCP or HTTP protocols to traverse network barriers, like NATs and firewalls. A JXTA network consists of peers (*edge-peers*) and super-peers (*rendezvous peers* and *relay peers*). A peer with enough privileges can become a super-peer depending on its location in the network. When a peer joins a JXTA network, it finds, either manually or automatically, the closest rendezvous peer and creates a special relationship with it. From this moment, the edge peer starts using the rendezvous peer as a gateway to the P2P network. The rendezvous peer maintains a list of its edge peers and their shared resources. Rendezvous peers organize themselves into a loosely coupled network, delivering queries and peer information between each other. Rendezvous peers use DHTs (Distributed Hash Table) for optimizing peer and service discovery. In this respect, JXTA differs a lot from Gnutella's style of broadcasting queries to neighbor peers. Actually, DHTs were already used in some advanced second-generation protocols, like CAN [6]. JXTA introduces also the *relay peers* that can route JXTA messages and data between peers that have no direct connection between each other. Relay peers are used also in spooling messages for unreachable or temporarily unavailable peers. With the functions mentioned above, JXTA in practice allows any peer to reach any other JXTA peer independently of its network location.

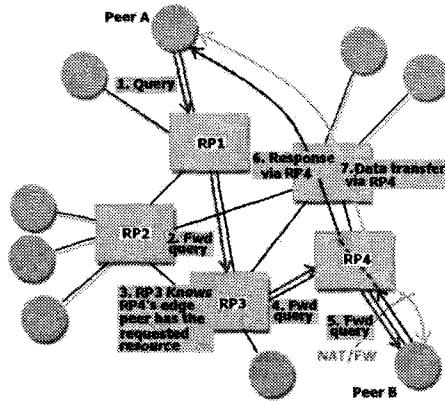


Figure 4. The operating principle of JXTA.

Searching for resources in JXTA is illustrated in Figure 4. A peer (for example A) is requesting a resource, which is in this case in another peer located in the network behind NAT. When querying a resource, A sends a query to its rendezvous peer (for example RP 1) (1.). RP1's index does not contain the requested resource, so it relays the query to its own rendezvous peer RP3 (2.). RP3's index contains the requested resource with the information that the resource is available some of in RP4's edge-peers (3.), so RP3 relays the query to RP4 (4.). RP4 knows the resource is in its edge peer B, so it relays the query to B (5.). Because B is in a network using NAT, it sends the response to A via its relay (and rendezvous) peer RP4 (6.). Then, the data is transferred between A and B using RP4 as a relay (7.).

The third generation has solved many of the problems of the earlier generations. The hierarchical model has effectively decreased the stress caused to the underlying network protocols when compared to the second generation. Third-generation protocols also provide new services, like peer groups and NAT/firewall traversal.

2.4 Current P2P architectures in mobile environment

In desktop environment, the current 3rd generation P2P architectures have matured to the point where they work rather well and the overhead inflicted to the network has decreased from the earlier generation architectures. However, this applies only for desktop and laptop environments with wideband Internet connections, and high processing and memory capacity. All the currently available P2P protocols have been designed with a desktop environment in mind, and thus there are no any well-known third-generation protocols designed especially for mobile devices.

The principle of third-generation architectures as such is suitable for mobile use. Although there are many advantages in using 3rd generation protocols in a mobile environment, there are also drawbacks. Current third-generation protocols, like JXTA, are too heavy for effective mobile use.

As a response to this problem, the JXTA community has developed a light version of JXTA for mobile devices, called JXME (JXTA for J2ME). It works in all MIDP devices, e.g. Nokia's Series 60 phones. JXME has two versions: proxyless and proxied. The proxyless version works similarly to native JXTA, whereas the proxied version needs a native JXTA peer to be set up as its proxy. The proxied version is also lighter than the proxyless one, since it uses binary communication with its proxy, whereas the proxyless version uses XML-based communication. The proxied version cannot work as a super-peer either. Unfortunately, JXME works currently only in Java environment and there are no JXTA versions yet ported to native Symbian C++ language.

Table 1. Table Comparison of P2P protocols and frameworks

Main Feature	SIP	JXTA	JXME *	XMPP
Inherently P2P	No	Yes	Yes/Yes	No
Server-centric	Yes	No	No/No	Yes
Suitability for thin peer (lightweight)	Good	Poor	Good / Average	Good
Built-in resource discovery	No	Yes	Yes/Yes	Yes
Support for real-time data streaming	Good	Average	Poor / Average	Good
Interdomain mobility awareness	Yes	No	No/No	No
Built-in NAT and firewall traversal	No	Yes	Yes/Yes	No
Interoperability with other protocols	Good	Poor	Poor/Poor	Good
Programming language independent	Yes	Yes	No/No	Yes

* JXME itself is not a protocol, but a lightweight implementation for mobile devices, using only a part of JXTA protocols.

The 1st generation P2P protocols, like SIP, are light and therefore seem to be even better suited for mobile use, but they still have the drawbacks that are characteristic to first-generation architectures, as pointed out earlier in this paper: server-centric architecture, difficulties in NAT/firewall traversal, cost etc. Jabber, which uses XMPP protocol [9,10], is another XML-based P2P framework. Its architecture is close to the 1st generation of P2P, but also has features from the 3rd generation. The Jabber network consists of a dynamic network of servers and clients using them. Actually these servers have the same kind of role as super-peers in the JXTA/JXME system. It works like an e-mail system but its functionality is close to real-time. Jabber is lighter than JXTA and therefore seems to suit mobile use better.

Table 1 shows a comparison of the four P2P protocols or frameworks mentioned above. It can be seen that each of them has its advantages and disadvantages. All of them are good in many areas, but also lack something important. When thinking about the restrictions and the potential of mobile devices, desirable features for mobile P2P seem to consist of a combination of the features in Table 1. These desirable features are at least:

- Decentralization
- Lightness
- Awareness of interdomain mobility
- Ability to traverse NATs and firewalls
- Interoperability between P2P protocols
- Peer and resource discovery
- Support for real-time and non-real-time data transfers
- Independence of programming languages

3. TOWARDS MOBILE P2P

For further development of P2P towards the mobile world, we have identified two possible approaches. A new protocol, optimized especially for mobile devices, is the most obvious option. Another, and more generic, approach is to develop a platform that would lie between the P2P protocol layer and the application layer. This platform would hide the underlying protocols from the applications and users by dynamically using the existing protocols, and providing an interface for applications using it. We have named our proposal *Plug-and-Play Application Platform (PnPAP)*.

3.1 Requirements for Mobile P2P

As the mobile environment brings new challenges to P2P computing, mobile P2P must offer methods for coping with them. When analyzing the benefits and drawbacks of the existing P2P technologies, the third-generation P2P model would be a good starting point for mobile P2P. This architecture is a good compromise between the best features of the server-centric first generation and totally decentralized P2P second generation. Peer and resource discovery, group management, and NAT/firewall traversal are the most beneficial functions specific to third-generation protocols. However, in first-generation P2P architectures, e.g. SIP, there are many good features that mobile P2P should comprise. One of these is SIP's awareness of mobility: SIP's redirect function is very beneficial in the handovers between network domains [11].

Third-generation P2P protocols have more functionality, so it may be difficult to make the peers as light as for example SIP end-user clients. But there are many things that can be optimized. The hierarchical architecture enables using super-peers as agents

for mobile peers, by moving as many functions as possible to be handled by super-peers. The proxied version of JXME does this by using JXTA super-peer as its proxy. Jabber's approach is more modular: whereas JXTA protocols cover a wide variety of functions, the base protocols of Jabber specify only the basic session management functions, and the more advanced functions can be added as plugins to the system. Mobile P2P should afford the same kind of modularity.

Interoperability with other P2P protocols is essential. Mobile P2P should not be an isolated "island" among P2P networks, where peers cannot connect to peers outside the protocol and vice versa. Instead, mobile P2P should be interoperable with as many other P2P and messaging protocols as reasonably possible. For example, JXME interoperates with any other JXTA peer without modifications, but JXTA itself is not natively interoperable with other P2P protocols. Due to the modular structure and the availability of plugins, Jabber and SIP are easier to make interoperable with other P2P and messaging protocols.

Nowadays, the need for real-time applications is growing fast, so it is important that mobile P2P supports them as well as possible. SIP is an example of a P2P protocol made especially for real-time applications. There are also third-generation P2P applications supporting real-time data. One well-working example of these is Skype[†].

Security is another important area to be taken into account in the development of mobile P2P. When compared to the first generation's server-centric model, the hierarchical model is less vulnerable to denial-of-service attacks, but also raises the importance of trust management. A peer must, for example, trust that the super-peer to which it is attached is what or who it says it is.

In short, mobile P2P should combine the best and the most suitable features of the existing P2P protocols to best achieve its goals.

3.2 Plug-and-Play Application Platform

In the future, fixed and mobile devices communicate seamlessly in ubiquitous All-IP networks using several protocols. Instead of one dedicated mobile P2P protocol, it seems more feasible to develop a platform which could use the existing protocols dynamically. To ensure the compatibility with other peers, the platform should also be able to download and take in use new communication protocols on the fly. On the other hand, this platform would give the application developers a more generic way to develop applications, which would not be stuck with one protocol.

Currently, developers have to make a decision on using older protocols that often have less admirable features but are more widely used than newer protocols. The final decision is always a compromise between the protocol's features and its compatibility with the most commonly used applications and protocols. This makes also the end-user decide which application to use, since all of them have advantages and disadvantages depending on the protocols used. The situation with Instant Messaging protocols is exceptionally bad with a huge amount of different service providers, all equipped with different protocols. In the mobile environment these problems are even emphasized due to the

restrictions of the network and the device itself. PnPAP is our proposal for a platform, which will give mobile applications an efficient and flexible way to communicate with each other using different communication protocols and connection technologies dynamically.

3.2.1 PnPAP definition and analysis

PnPAP is an API on top of a device's operating system, through which applications can access many kinds of networks over various protocols with basically only one method call. In other words, PnPAP provides an abstraction layer for networked applications, hiding the complexity of various underlying P2P protocols including for example JXTA, SIP and Jabber. PnPAP gives a developer the plug-and-play functionality on the protocol level, which gives the developer more freedom to concentrate on the application's core functionality. PnPAP can also be coupled with the *Holistic Connectivity (HCon)* management layer [11] plugin, which can dynamically use many different network connection technologies, such as WLAN, Bluetooth, EDGE and GPRS. Together, the PnPAP and HCon form an effective platform especially for mobile devices. Figure 5 illustrates the structure of PnPAP and HCon.

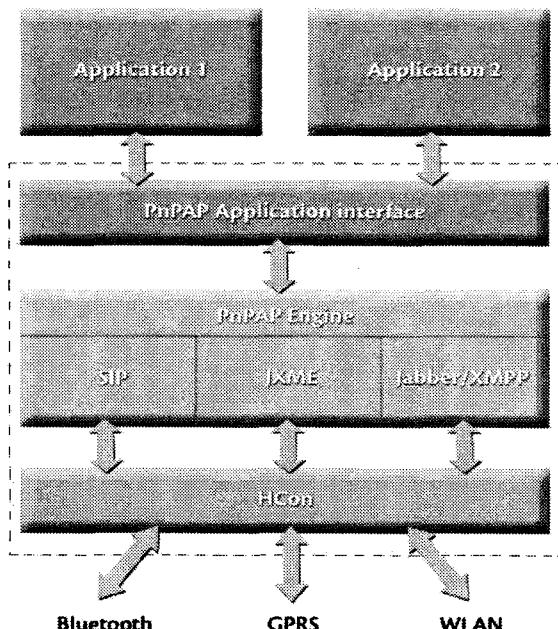


Figure 5. PnPAP architectural overview.

PnPAP consists of an interface for applications and an engine behind it, which reacts to requests received through the interface. The engine can communicate with the outside world using several P2P protocols. If the application does not specify a protocol, the messages can go out using the most suitable protocol at hand. This does not only ensure that the most optimal protocol can be used at all times, but also the number of devices that the application can connect to increases substantially. If the only protocol known by the other end is for example SIP, it could still be used. Protocols can be switched on the fly, or multiple protocols can be used simultaneously. When needed, protocols

[†] Skype is a real-time VoP2P (Voice over Peer-to-Peer) conference application, which is based on Kazaa P2P protocol.

can be downloaded from a special protocol repository, or from some other peer with PnPAP in the network.

PnPAP enables considerably faster and easier application development, since application programmers do not need to go into low-level protocol coding. This also helps debugging and leaves the developer more time to concentrate on the application itself. In fact, a user interface and an application controller communicating through the PnPAP engine are the only things needed to create a new application. See more about implementation-related issues in [12]. PnPAP is also planned to offer services to all networkable programming languages that can be run on the mobile device. This gives developers more freedom to select the language based on their preferences. On the other hand, nowadays a developer must often decide between programming languages based on the requirements of a particular application, since most of their mobile versions of APIs have artificially hindered functionalities. For example, J2ME CLDC/MIDP does not provide access to the mobile phone's file system without external components because of potential security hazards. This prevents e.g. creating MIDP applications that could transfer files between peers. Some other programming language APIs, such as Symbian C++, provide almost unlimited access to the mobile's resources. These differences in restrictiveness seem strange and regular end-users rarely even know about this.

PnPAP provides good compatibility with different applications, protocols, devices and networks. Many communication protocols have different plugins that balance the functionalities between the protocols. There are, for example, drafts specifying peer-group-like functionality for SIP: SIMPLE and XCAP [11]. There are also projects in the JXTA community to enable voice communication over a P2P network. Since the functionalities are getting even, it even emphasizes the benefits of PnPAP. There no longer has to be many applications that are committed to using only one protocol each but only one application that can use them all.

Addressing also needs attention in the case of PnPAP. Since basically every P2P protocol has its own method for peer naming and addressing, there is a need for address translation. PnPAP should contain methods for resolving an address of one protocol from an address of any other P2P protocol PnPAP is using.

We acknowledge the need of some security constraints. However, using PnPAP instead of lower level operating system APIs, many security hazards are easier to handle. PnPAP should include authentication methods for applications to increase the security level. Also the exchange of control and status messages between PnPAP engines in different peers should be secure. The hazard of fake PnPAP peers can be prevented this way. These fake peers could, for example, send PnPAP engine an erroneously working protocol when requested. This erroneously working protocol could then send some personal data to a third party that could benefit from it.

3.2.2 Holistic Connectivity

HCon, which has knowledge and control of different network connection technologies, enables simultaneous use of multiple different network connections. PnPAP makes it possible to switch connectivities on the fly without the user or even application noticing it. Network peers can be contacted by using the most suitable technology for the task at hand. For example, a Bluetooth-equipped cell phone could connect to another that has both Bluetooth and WLAN connections. This way the

communication could be forwarded from the Bluetooth-connected device into a WLAN network and back. The combination of PnPAP and HCon provides far more extended possibilities for communication of peers than today's solutions.

HCon eases the co-operation of network technologies, especially with today's technologies, but also with new beyond-3G technologies. Adding new connectivities and protocols to the platform is straightforward due to well-defined interfaces. The optimization is more comprehensive since PnPAP together with HCon can now see both up and down in the protocol stack. Thus, the criteria of optimization can vary depending on the applications' requirements and hardware resources. These criteria can be regarding for example:

- Running applications and sessions
- Protocol compatibility
- Data transfer rate
- Overall delay
- Processing load
- Battery consumption
- Other QoS criteria

3.2.3 State Machine

The behavior of the PnPAP engine can be controlled by a *state machine* (SM). The SM decides, for example, what protocols and connectivities to use and when to switch between them. In this model, the engine and the SM communicate using events that the engine can send to the SM to trigger state transitions. A transition can, in turn, generate an event that defines an action to be performed in the engine. For example, the engine can generate an event such as "SIP available". This can trigger a state transition that generates the event "Switch to SIP" to the engine.

The novelty of this approach is in representing the SMs explicitly using an RDF-based (Resource Description Framework) representation. RDF is represented using XML. The SM is executed by a state machine interpreter that actually communicates with the engine. The advantage of this approach is that the behavior of an engine can be changed without modifying the code - hence no compilations or installations of new versions are needed. Instead, the engine passes the SM description to the interpreter as data. A description is passed when the engine is initialized and new descriptions can be passed even when applications are running.

Furthermore, SMs are beneficial in representing asynchronous behavior, and this is just the type of behavior encountered in this application area. Representing SMs explicitly facilitates rapid prototyping and even switching to new SMs, while applications are still running. For example, a new P2P protocol might be defined as an SM. The protocol SM might be drawn with a graphical tool; the resulting RDF description could be downloaded into the engine, and then modified on the basis of test results. The feasibility of interpreting RDF-based SMs has already been verified, see [13].

4. DISCUSSION

As seen, PnPAP is planned to be very rich in features. This sets challenging requirements for optimizing the consumption of memory and CPU power. PnPAP is meant to be running all the time that the mobile device is on, which means that especially the memory consumption should be optimized well. The overhead and the performance of different protocols can vary significantly.

SIP, for example, is a very light protocol when compared to JXTA. PnPAP should take this into account especially when the only available connection is slow, e.g. GPRS [11]. This means that PnPAP and HCon should co-operate to find the optimal protocol based on the knowledge of the needed features and the available hardware resources, including network connections. The state machine used in the PnPAP engine brings undeniable benefits, but the limited resources of the mobile platforms present challenges also to their implementation. Especially the RDF processing is better to be left to more capable peers that, in turn, send the SM definitions to mobile devices in binary format. The need of updating the SM is not very frequent, so it would not stress the network significantly.

Overall, PnPAP will give a new type of freedom to create applications, since they do not have to be done using a single protocol. PnPAP is a more generic solution than developing a new mobile P2P protocol. However, this does not rule out developing mobile P2P protocols; as a matter of fact, a mobile P2P protocol could be one of the protocols PnPAP uses.

To sum up, SM controlled PnPAP with HCon has the following advantages over the traditional model:

- Uniform environment for all P2P applications
- Easier application development and maintenance
- Always the most optimal P2P protocol-connectivity pair in use
- Better interoperability between protocols, devices and networks
- The behaviour of the PnPAP can be changed on the fly thanks to RDF-based interpretable SMs

5. CONCLUSIONS

This paper has provided an analysis of the P2P architecture evolution toward mobile P2P. First, it has identified the challenges and requirements for mobile P2P, followed by a definition of a novel Plug-and-Play Application Platform (PnPAP). It enables dynamic selections between diverse P2P and session management protocols while preserving the best available network connectivity through Holistic Connectivity (HCon) management. On the fly reconfiguration and run-time parameter optimization can be done with a lightweight interpretable state machine. The concept enables flexible and seamless communications for mobile devices in P2P networks. Future work will contain developing and implementing the PnPAP towards its first prototype.

6. ACKNOWLEDGEMENTS

The acknowledgments are due to the project team of Application Supernetworking/All-IP project. The authors would like to thank National Technology Agency TEKES, Nokia, Elektrobit, TeliaSonera, Serv-IT and IBM for supporting this project financially.

7. REFERENCES

- [1] Z. Li, D. Huang, I. Liu, J. Huang, "Research of peer-to-peer network architecture", *Proceedings of the International Conference on Communication Technology (ICCT 2003)*, Vol. 1, pp. 312 – 315, Beijing, China, April, 2003.
- [2] M. Handley, H. Schulzrinne, E. Schooler, J. Rosenberg, "SIP: session initiation protocol", RFC 3261, Internet Engineering Task Force(IETF), June 2002.
- [3] M. Ripeanu, "Peer-to-peer architecture case study: Gnutella network", *Proceedings of the First International Conference on Peer-to-Peer Computing (P2P 2001)*, pp. 99–100, Linköping, Sweden, Aug. 2001.
- [4] N. Maibaum, T. Mundt, "JXTA: a technology facilitating mobile peer-to-peer networks", *Proceedings of International Mobility and Wireless Access Workshop (MobiWac 2002)*, pp. 7-13, Fort Worth, TX, USA, Oct. 2002.
- [5] S.K. Sai Ho, S.M. Lui, R. Cheung, S. Chan, CC. Yang, "Searching behaviour in peer-to-peer communities", *Proceedings of International Conference on Information Technology: Computers and Communications (ITCC 2003)*, pp. 130-134, Las Vegas, NV, USA, April 2003.
- [6] M. Castro, M.B Jones, A-M. Kermarrec, A. Rowstron, M. Theimer, H. Wang, A. Wolman, "An Evaluation of Scalable Application-level Multicast Built Using Peer-to-peer Overlays", *Proceedings Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2003)*, Vol. 2, pp 1510–1520. San Francisco, USA, March-April 2003.
- [7] T. Hsin-ting Hu, T. Binh, A. Seneviratne, "Supporting mobile devices in gnutella file sharing network with mobile agents", *Proceedings of Eighth IEEE International Symposium on Computers and Communication (ISCC 2003)*, pp. 1035–1040, Kemer-Antalya, Turkey, June-July 2003.
- [8] P. Dasgupta, "Agent based peer-to-peer systems", *Proceedings of The 2002 45th Midwest Symposium on Circuits and Systems (MWSCAS 2002)*, Vol. 1, pp. 663-666, Tulsa OK, USA, August 2002.
- [9] P. Saint-Andre et al. "Extensible Messaging and Presence Protocol (XMPP): Core", Internet-Draft (work-in-progress), IETF XMPP Working Group, 87 p, May 2004.
- [10] P. Saint-Andre et al. "Extensible Messaging and Presence Protocol (XMPP): Instant Messaging and Presence", Internet-Draft (work-in-progress), IETF XMPP Working Group, 109 p, April 2004.
- [11] D.Howie, M. Ylianttila, E.Harjula, J. Sauvola, "State-of-The-Art SIP for Mobile Application Supernetworking", *Proceedings of Nordic Radio Symposium, including Finnish Wireless Communications Workshop (NRS/FWCW 2004)*, Oulu, Finland, August 2004.
- [12] J. Ala-Kurikka, M. Ylianttila, E.Harjula, O. Kassinen, "Empirical Aspects on Implementing Mobile Application Supernetworking", *Proceedings of Nordic Radio Symposium, including Finnish Wireless Communications Workshop (NRS/FWCW 2004)*, Oulu, Finland, August 2004.
- [13] T. Mäenpää, T. Tikanmäki, J. Riekki, J. Röning, "A Distributed Architecture for Executing Complex Tasks with Multiple Robots", M. International Conference on Robotics and Automation (ICRA 2004), pp. 3449-3455, New Orleans, LA, USA, April-May 2004.

Survey of Requirements and Solutions for Ubiquitous Software

Eila Niemelä

VTT Technical Research Centre of Finland

Embedded Software

P.O.Box 1100

FIN-90571 Oulu, Finland

Tel +358 551 2111

Eila.Niemela@vtt.fi

Juhani Latvakoski

VTT Technical Research Centre of Finland

Embedded Software

P.O.Box 1100

FIN-90571 Oulu, Finland

Tel +358 551 2476

Juhani.Latvakoski@vtt.fi

ABSTRACT

Ubiquitous computing embeds computer technology in our everyday environment, providing a human with information services and applications through any device over different kinds of networks. Ubiquitous computing can be seen as a prerequisite for pervasive computing that emphasizes mobile data access and the mechanisms needed for supporting a community of nomadic users. Ubiquitous software is the software required in ubiquitous computing environments. This paper surveys the challenges and state-of-the-art software technologies applicable to ubiquitous computing environments. Ubiquitous wireless world systems trigger a set of requirements, e.g. interoperability, adaptability and mobility, for ubiquitous system and software technologies. The main challenges of ubiquitous software are a uniform and adaptive middleware technology, interoperability of services and networks, and the enabling technologies required in their development. Furthermore, guaranteeing secure transactions between service providers, content providers and users is essential in worldwide pervasive computing environments. Although standards, reference architectures and generic software technologies provide the basis for future ubiquitous software development, new kinds of micro architectures and software technologies, and development methods are needed.

Keywords

Adaptability, Interoperability, Software architecture, Ubiquitous computing, Pervasive computing.

1. INTRODUCTION

Ubiquitous computing enhances the use of computers by making computers effectively available throughout the physical environment and, at the same time, making them invisible to the

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided such copies are not made or distributed for profit or commercial advantage and they bear this notice and the full citation on the first page. To otherwise copy, republish, post on servers or redistribute to lists requires prior specific permission and/or a fee.

*Mobile Ubiquitous Computing Conference'04, October 2004,
Washington DC, State, USA*

Copyright 2004 ACM 1-58113-000-0/00/0004...\$5.00.

user. Mark Weiser [1] expressed this goal as achieving the most efficient technology and making computing as ordinary as electricity. In ubiquitous computing, the focus was first on small special-purpose devices, network protocols, interaction substrates and new styles of applications. Later, new directions were identified: wireless communications, disconnected operation, location and resource discovery, privacy, and power consumption.

Pervasive computing is another term used in the same context but from different points of view. Pervasive computing emphasizes mobile data access, smart spaces and context awareness. Pervasive computing also focuses on nomadic users and the way they apply devices and interact with the environment; the best ways to deploy new functions on a device and exploit interface modalities for specific tasks. Thus three major focus areas of pervasive computing are:

- the way people see mobile and wireless computing devices and use them;
- the way applications are created and deployed to end users; and
- how ubiquitous services enhance the environment.

However, in this paper we use the terms 'ubiquitous' and 'pervasive' interchangeably.

As a vision for the ubiquitous computing paradigm, the Wireless World Research Forum (WWRF) introduces the MultiSphere model, which consists of several levels: personal area network (PAN), immediate environment, instant partners, radio accesses, interconnectivity, and cyberworld [2]. These levels set new requirements for computing platforms and software technologies. For example, innovative solutions for energy-efficient radio accesses are required. However, one of the most essential challenges will arise from the complexity and intelligence required in software embedded in ubiquitous computing devices. Therefore, the contribution of this paper is to survey the challenges of ubiquitous software technologies.

The paper is organized into five sections. After the Introduction, the requirements of ubiquitous computing systems are presented. Section 3 provides an overview of state-of-the-art ubiquitous software technologies. Section 4 discusses the issues to be investigated, and concluding remarks close the paper.

2. REQUIREMENTS

The requirements of ubiquitous computing systems are classified into three categories: system, software, and business and organization (Figure 1). The term system refers to computing platforms. Ubiquitous software refers to services and components of a ubiquitous system realized by software technologies. Business refers to the network of actors who provide software components and services for the system development and deployment. Organization includes the development methods and processes needed for the development and integration of services provided by multi-actor business networks. The requirements of ubiquitous computing environments are interoperability, heterogeneity, mobility, adaptability, security and privacy, self-organization, and augmented reality and content scalability. Most of these requirements have to be coupled with the system, software and business and organization levels.

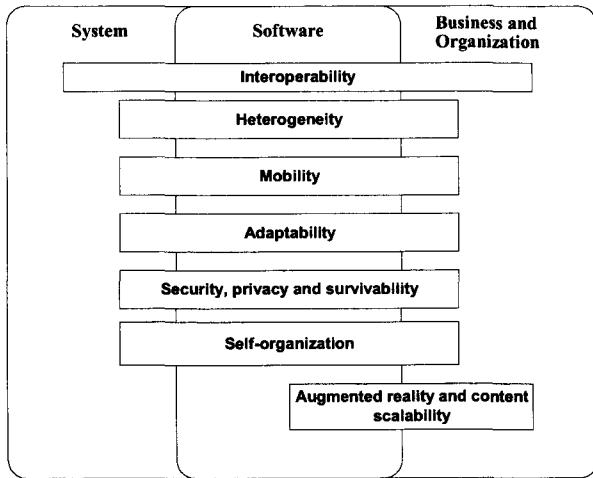


Figure 1. Special requirements of ubiquitous computing environments.

2.1 Interoperability

Due to the increasing need for networked products from different vendors, interoperability is required at all levels of ubiquitous computing. The increasing number of microprocessors and amount of networking highlights the need for distributed software platforms applicable to wireless computing [3] and the standards used in their development. Interoperability is the ability of software to understand the exchanged information and to provide something new originating from the exchanged information. It is considered when components and their interactions are defined in detail and finally observed in executable models, simulations and running systems. [4].

Integration architecture is a software architectural description of the overall solution to interoperability problems between various component systems. The focus of integration architecture is on a pre-integration assessment of the architecture. The purpose of this assessment is to discover inherent interoperability problems by an architecture mismatch analysis that reveals the underlying reasons for interoperability problems among software components [5]. Software components should be built to be independent of the

context in which they are used, as this allows their use in different computing environments and applications. If a uniform description language is used for component specification, this description can be used for binding components dynamically. Put the other way around, when the context of a component changes, the interconnections between components are changed according to the new context.

2.2 Heterogeneity

In the ubiquitous world, heterogeneous networks will form integrated networks that connect devices with different screen resolutions, user interaction methods, radio capabilities, memory, power and processing capabilities, as well as mobility. Services can be accessed with widely varying transport capability, quality and usage cost, and they may have different requirements for bandwidth, real-time capabilities, and input output methods. User interaction with computers is performed by various mechanisms that enable the users to interact with devices using multi-modality such as speech, hand movement, screens, buttons, etc. This kind of interaction requires novel solutions in the form of embedded sensors.

Heterogeneity of software is expressed by a diversity of software structures, component models, interface technologies and languages. The component model depends on the selected middleware, which sets requirements and constraints for the architectural structure and, in many cases, also for the implementation languages used in application development. The software may consist of components, subsystems and services provided by different actors, and the value chain may be a complex value network between these actors.

2.3 Mobility

Mobility is an important characteristic of a ubiquitous system. There are different kinds of mobility schemes, such as terminal mobility, personal mobility, session mobility and service mobility. Users should be supported in such a way that they can move from one place or terminal to another and still get a personalized service [6]. Networks may also be mobile and dynamic, and full mobility is the requirement of the ubiquitous wireless world [7].

From the software perspective, mobility can be divided into actual, virtual and physical mobility [7]. Actual mobility is an extension to the capability of an autonomous software agent that dynamically transfers its execution, i.e. its code, data and execution state, towards the nodes where the resources it needs to access are located. Exploitation of actual agent mobility can save network bandwidth and increase the reliability and efficiency of the execution. Agent mobility can effectively be exploited by nomadic users who are assisted by personal software agents capable of following them in their activities. Virtual agent mobility is the ability to be aware of the multiplicity of networked execution environments. When agents are aware of the distributed nature of the target, and explicitly locate and access Internet resources in the environment, this allows virtual mobility of agents across execution environments. Physical agent mobility means mobile and wireless computing devices connecting to the Internet from dynamically changing access points. An active space extends the physical space, i.e. physical objects, networked devices and users with well-defined physical boundaries, adding coordination via a context-based software infrastructure [8].

2.4 Survivability and Security

Survivability is the ability of a system to fulfill its mission timely and in the presence of attacks and failures. Survivable systems require a self-healing infrastructure with improved qualities, such as security, performance, reliability, availability and robustness [9]. A key characteristic of a survivable system is its ability to deliver essential services even in the face of attack, failure or accident. Thus it is important to define minimum levels of quality attributes associated with the essential services. Survivability is often expressed in terms of trade-offs among multiple quality attributes, such as performance, security, reliability, availability, and modifiability. Because quality attributes represent broad categories of related requirements, a quality attribute may contain other quality attributes [10], [11]. The definition and analysis of survivability requirements is the first step to be addressed, not only by the functional requirements but also by the requirements for software usage, development, operation and evolution [10].

An application may employ security mechanisms, such as passwords and encryption, yet may still be fragile by failing when a server or network link dies. On the other hand, an application must be able to survive malicious attacks and must, therefore, support security. There are two kinds of survivability: survival by protection (SP) and survival by adaptation (SA). In SP, security mechanisms like access control and encryption attempt to ensure survivability by protecting applications from harmful (accidental or malicious) changes in the environment. In SA, the application can survive by adapting itself to the changing conditions. These two kinds of survivability are not mutually exclusive; an application may utilize security mechanisms in SA as well. For example, it may start using access control or increase the key length when it perceives the threat of an intrusion. [12].

2.5 Adaptability

Software services must adapt to different kinds of terminals and networks. They also have to handle dynamically emerging and evolving contexts and user preferences. A ubiquitous system may have different kinds of radio capabilities (multi-radio). Due to mobility, dynamically changing conditions make adaptability a big challenge.

There are several adaptation strategies. In the *laissez-faire approach*, the responsibility of adaptation is left to individual applications and no system support is provided for adaptations. However, this approach lacks the central arbitrator to resolve the incompatible resource demands of different applications and to enforce limits on resource usage. Even though the system support for adaptation can be avoided in this approach, the applications become more difficult to implement and the size of the applications increases because each application needs to implement its own adaptation functionality individually. [13].

The other extreme of adaptation strategies is the *application-transparent approach*, where adaptation does not require any changes in the applications and is fully left as the responsibility of the underlying system. Even when providing backward compatibility with existing applications, this approach has its drawbacks; there may be situations where the adaptation performed automatically by the system is inadequate or even harmful.

Between these two extremes of adaptation strategies lay a number of solutions that are collectively referred to as application-aware adaptation. Application-aware adaptation emphasizes the collaborative partnership of applications and the system in the adaptation functionality. This approach allows applications to determine the best adaptation behavior for the situation but preserves the ability of the system to monitor resources and enforce allocation decisions. Application-aware adaptation also reduces the application size compared with the laissez-faire adaptation approach because part of the adaptation functionality is provided by the system and each application does not have to have embedded adaptation functionality [14].

Agility is a set of combined quality properties, sensitivity to varying resources (e.g. battery power and bandwidth) and sensitivity to changes in resource availability (e.g. data sharing in intermittent connections) [13]. In the case of dynamically changing resources, agility is mapped to the execution of a software and its ability to manage changes that are timely unpredictable but their characteristics are predictable. In addition to execution agility, all software systems also embody evolvability - the ability to handle changes in the long-term, considering the life cycle of a system [15].

2.6 Ability of Self-Organization

Self-organization is the ability of a system to spontaneously increase its organization without the control of the environment or an encompassing and external system. Self-organizing systems not only regulate or adapt their behavior but also create their own organization. Self-organization applies the concepts of self-learning, expert systems, chaotic theory and fuzzy logic. Self-organization may be applied in communication networks, such as ad hoc networks, to reach improved performance and efficiency, to minimize cost and to increase reliability and survivability [16].

Ad hoc networks consist of dynamically connected devices and wireless media [17]. Ad hoc networks are automatically organized without any static configuration and centralized management, i.e. self-organization of communication networks. From a user's perspective, these systems can be called spontaneous systems [7] [18]. From a software perspective, the ability to self-organize refers to the ability to dynamically re-organize the structure of the software - in other words, having dynamic software architectures.

2.7 Augmented Reality and Scalable Content

New ways of looking at the content, like augmented reality, are emerging. In augmented reality, the human awareness is augmented by using a virtual context in parallel with the context sensed by a human [19]. Therefore, the requirements for augmented reality and scalable content include many perspectives, such as digital rights management, self-organization and semantic awareness.

The term fidelity has been used as a property of a system that defines the degree to which data presented at a client matches the reference copy at the server [13]. Fidelity includes three dimensions: consistency, the type of data, and tradeoffs made by applications. When network connectivity is poor or non-existent, data provided to applications may be stale but still useful for achieving appropriate functionality in a system. Data consistency means high availability of shared data in intermittent networked systems. Data types are based on time, state and frequency.

Sampling rate and timeliness are quality properties of the telemetry data type. The size and resolution of data is considered the fidelity of spatial data, e.g. topographical maps. Frame rate, in addition to the image quality of each frame, is the key issue in video streaming.

3. UBIQUITOUS SOFTWARE TECHNOLOGIES

3.1 Technology map

Figure 2 visualizes software technologies for ubiquitous systems. Ubiquitous computing consists of three main software layers: applications (services), middleware [20] and system infrastructure. Standards and reference architectures are required to enable ubiquitous computing system(s). In addition, a set of generic software technologies and specialized advanced software technologies are required.

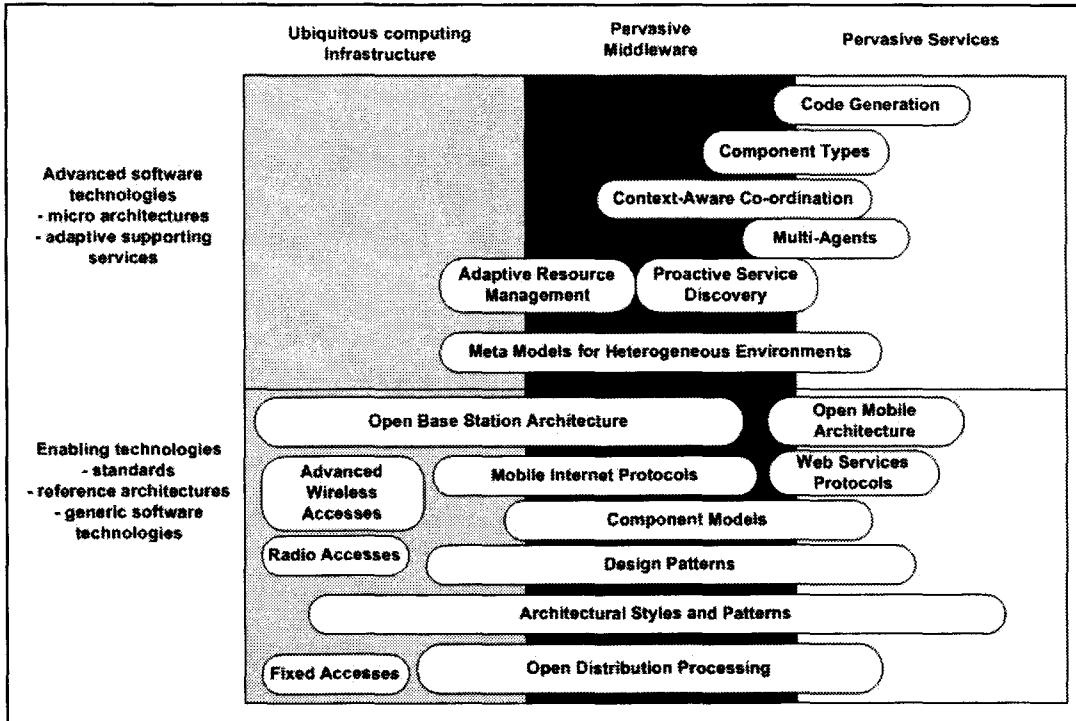


Figure 1. Software technology map for a Ubiquitous system.

Generic software technologies concerning development methods, architectural styles and patterns, as well as component models and appropriate languages, are used at the application level as well as at the middleware and infrastructure level. Thus generic software technologies provide standard-based technologies that software should be based on. Programming and interface languages such as Java, IDL (Interface Definition Language) and XML (eXtensible Markup Language) are technologies commonly used for defining and implementing software components and services modeled by UML (Unified Modeling Language). Interface technologies are even more important in the service development, in which software developed in multi-organizational settings is composed and used together. COM+ (Component Object Model) and EJB (Enterprise Java Beans) are the most promising component models that will also be applied to pervasive software.

Architectural styles and patterns, such as an implicit invocation style and an observer pattern as well as architectural viewpoints, will play a key role in the pervasive software development. For

example, QADA^{SM1} is an architecture development method that focuses on the quality requirements and keeps them as driving factors for achieving a stable and reliable architecture that can be shared by a wide community of software developers.

Open Distributed Processing, which introduced middleware based on remote procedure calls, is also used in pervasive computing systems. However, the communication of mobile and wireless services is more often based on asynchronous messages than on synchronous procedure calls.

¹ QADA (Quality-driven Architecture Design and quality Analysis) is the service mark of VTT Technical Research Centre of Finland.

3.2 Advanced Software Technologies

3.2.1 Adaptive Resource Management

The aim of adaptive resource management is to provide software solutions that assist in resolving the adaptability required in pervasive environments. Odyssey [13], a lightweight middleware, monitors resources such as bandwidth, CPU (Central Processing Unit) cycles and battery power, and interacts with each application to best exploit them. For example, when connectivity is lost due to a radio, Odyssey detects the change and notifies the interested applications. Reaction to the notification depends on the application.

Odyssey is realized as a new VFS (Virtual File System) connected to the NetBSD² (Net Berkeley Software Distribution) kernel. Odyssey uses two component types: a viceroy that is responsible for the centralized resource management, and wardens that are data type-specific pieces of software that provide the system-level support to clients to effectively manage data types. The applied architecture follows the data-centric and event-driven architectural styles. The data-centric style defines the fidelity levels for each data type and factors them into the resource management. Action-centric communication is used between middleware and applications for providing applications with control over the selection of the fidelity levels supported by the wardens.

The range of adaptation is defined by two end points: laissez-faire and application-transparency. The best solution for a particular ubiquitous system is somewhere between these two extremes because laissez-faire means that the applications themselves have to take care of adaptation and application-transparent adaptation does not support the diversity of applications.

3.2.2 Proactive Service Discovery

The Proactive Discovery Service (PDS) concept allows clients transform from controlling pull-based passive interfaces to trade control for performance because message traffic is only generated when updates occur [21].

PDS is message-based, as opposed to W3C's service discovery that is heavily based on RPC (Remote Procedure Call), although the necessity for a message-based interaction model has already been identified [22]. The PDS concept uses three component types: PDS clients, servers and object owners. Clients discover available objects in the environment and become aware of any change in them that could affect their functionality and/or performance. Object owners publish their objects through the directory service. Servers act as mediators. Related information is organized into well-defined collections called entities, where each entity represents an instance of an actual object type in the environment. Each object has an associated set of properties with particular values. Entities may be bound to names in different contexts and each context contains a list of name-to-entity bindings. Contexts may be bound to names in other contexts, building an arbitrary directed naming graph.

Dividing the global naming space into sub-spaces and assigning these sub-spaces to domains, each with a single root context,

obtains the scalability of the naming space. PDS is a general directory service, but shares a number of architectural ideas with GMA (Grid Monitoring Architecture) [23]. The PDS concept is also intended to be applied to widely distributed services through flexible replication strategies, dynamically adaptive server hierarchy management and automatic failure recovery. Dynamic service discovery with self-adaptation [24] is also considered a promising approach for a generic service discovery in which all kinds of services can interoperate with each other. This idea is based on Apple's Rendezvous, which focuses on easily connecting computers and devices through multicast interaction models and trader-like mechanisms. Rendezvous applies to software as well as to hardware, and Apple uses it for its iChat instant messenger system and file sharing.

3.2.3 Context-Aware Co-ordination

The MARS (Mobile Agent Reactive Spaces) architecture [25] introduces the coordination medium (CM) that is associated with an Internet node or a local administrative domain of nodes and is in charge of acting as a mediator for all coordination activities in that site. CM, through a specific API, provides the ability both to access the local resources of a site and to interact with other local agents and application agents (i.e. inter-agent coordination). Any coordination model, such as meetings, event-channels and tuple spaces, can provide the API for agents. The applied tuple spaces model can easily be integrated with the current web scenario.

The MARS architecture applies many styles and patterns. The intra-agent architecture depends on the task to be performed, but the main style is the independent-components style. Interactions between agents and between agents and environment define the main styles to be applied - the peer-to-peer style for networking and the rule-based style for defining global rules and environment-specific coordination laws. Moreover, reactions can be combined in a pipeline and a hierarchical Pipes-and-Filters style can be applied to install and uninstall reactions on a tuple space. The code and behavior of a reaction can also be changed without changing the agent and/or the other reactions, because the agents and reactions are implemented separately.

An adapter with the API running on top of middleware is a solution for runtime binding of components [26]. This binding model is based on two types of interactions: 1) adapters communicate with each other through a mediator or a facilitator, or 2) adapters communicate with each other directly. The first approach reduces implementation complexity at the expense of increased interaction overhead. Adapters invoke components based on a component's name and the operation name. The interface needs to be agreed on by the communicating components, by registering interfaces to the associated adapters of these components. The biggest drawback to the adapter API could be overhead, which can make this concept inapplicable in real-time ubiquitous systems and in some mobile ubiquitous systems.

3.2.4 Multi-Agents

A G-net system [27] includes a number of G-nets, each representing a self-contained module or an object. A G-net is composed of two parts: a special place called Generic Switch Place (GSP) and an Internal Structure (IS). GSP provides the abstraction of the module and serves as the only interface between

² <http://www.netbsd.org/>

the G-net and other modules. IS represents the design of the module.

The Planner module is the heart of an agent; it can ignore an incoming message, start a new conversation or continue with the current conversation. Through the Planner module, the Goal, Plan and Knowledge-base modules of an agent are updated after the processing of each communication that defines the type and the content of a message, or if the environment changes. The Planner module is both goal-driven and event-driven, because the transition sensor may fire when any committed plan is ready to be achieved or any new event happens. The Planner is also message-triggered because certain actions may initiate whenever a message arrives.

IS consists of incoming message, outgoing message and private utility called by the agent itself. The incoming/outgoing message section defines a set of Message Processing Units (MPU), which correspond to a subset of communicative acts. A new mechanism, called Message-passing Switch Place (MSP), is introduced for asynchronous message passing. When a token reaches an MSP, it is removed and deposited into the GSP of the called agent and the calling agent continues with its execution.

A mix of several styles is applied: IS is similar to the Pipes-and-Filters style and Planner acts as a broker, including independent modules. The dispatcher pattern has been used several times. The major benefit of the multi-agent architecture is the number of hot spots to be applied for extensions, modifications and reusability.

3.2.5 Models for Heterogeneous Platforms

Meta Models. Gaia [8] is a metaprogramming environment that exports a service to query, uses existing resources and context, and provides a framework for developing user-centric, resource-aware, multi-device, context-sensitive mobile applications. Gaia's main contribution is the functionality it provides for the interaction of individual services. This interaction provides users and developers with an abstract ubiquitous computing environment as a single reactive and programmable entity, instead of a collection of heterogeneous individual devices. Gaia has been applied in a ubiquitous computing environment similar to digitally augmented meeting rooms.

Gaia provides five basic services based on top of CORBA middleware: event manager, context service, presence service, space repository and context file system. Although similar kinds of names are used as patterns in Gaia, it does not have a clear connection to any style. However, the system is a mix of the event-driven style, the independent-components style and the rule-based style. The proxy pattern is used on behalf of the applications, services, devices and persons in the physical-entity presence subsystem. Periodic notification is sent as heartbeats of the present services and applications by using the observer pattern. The space repository stores information on all software and hardware entities in the space as XML descriptions, including the properties of services. The context file system (CFS) uses application-dependent properties and environmental context information to simplify many of the tasks that are traditionally performed manually or require additional programming. For the context model, Gaia uses first-order logic and Boolean algebra, which allow easy writing of rules to describe the context information.

The application framework consists of a distributed component-based infrastructure (derived and refined from the Model View Control (MVC) pattern), a mapping mechanism for customization and a group of policies that defines sets of rules for customization. The mapping mechanism defines two application description files: an application generic description and an application customized description that defines how the component is assembled, i.e. a list of components and how they are allocated and initialized.

Component Types. The main goal of the Aura architecture is to maximize the use of the available resources and to minimize user distraction [28]. This simplified model encompasses four component types:

- Task Manager, which embodies the concept of personal Aura.
- Context Observer, which provides information on the physical context and reports relevant events in the physical context to Task Manager
- Environment Manager, which embodies the gateway to the environment
- Suppliers, which provide the abstract services of which the tasks are composed

Each environment has one instance of Task Manager, Context Observer and Environment Manager that cooperates with the corresponding components in another environment. Task Manager strives to minimize user distraction in the face of a user moving to another environment or changing the environment, tasks and context.

For task migration, the service status is provided as a markup representation. The suppliers of a given service type share a vocabulary of tags and the corresponding interpretation. Existing applications can also be wrapped to the Aura architecture. Context Observers in each environment may have different degrees of sophistication, depending on the sensors deployed in that environment. Examples of sophistication dimensions are user recognition, location, activity and other people in the vicinity. When Suppliers are installed in an environment, they are registered with the local Environment Manager. The registry is the base for matching requests for services and it also keeps a record of the available capacity.

Aura supports dynamic reconfiguration in a transparent way and hides the variation of low-level interaction mechanisms from one environment to the next, implemented by connectors. Although the styles and patterns used have not been defined, the main architectural style seems to be an independent components style, using at least the observer and bridge patterns. However, self-awareness and adaptability of the environment is addressed at two levels. The infrastructure level monitors the availability and performance of components and communication (i.e. coarse-grained adaptation). At the lower level, the system components themselves are endowed with the ability to adjust their operation to the available resources.

Code Generation. The architecture of the generative approach [26] includes the Interface Manager and one or more generators defined in the Generator Database. While generating code, the selected generator uses information about the workspace context from the Context Memory. Code generation requires information

about services, appliance, and workspace. Generation is produced in the following way: services send a beacon about their presence, including a service description; an appliance supplies its appliance description; the workspace context is stored in a central datastore called the context memory.

The generative model uses proxies that are applied to achieve flexibility for different appliances. The main architecture is an event-based, blackboard and rule-based system that uses thoroughly defined service and device descriptions as the input for the code generation integrated as part of supporting services. Thus the framework also supports evolvability. Although the current realization is event-driven, the communication can be changed to remote procedure call, remote method invocation or message-based communication.

4. DISCUSSION

Although the approaches mentioned in the previous section give some technologies and solutions for ubiquitous systems to meet the new requirements set for them, there are still challenges to be solved. Meta models, component types and code generation assist in developing interoperable software but software services still need uniform description languages and standardized ontologies that provide seamless interoperability and ease the evolution of networked systems and services. Resource discovery should be automatic, and wireless access networks should be able to configure themselves based on built-in knowledge. However, these topics are still research issues for the future.

Multi-agent platforms, adaptive resource management and proactive service discovery provide prototype-like solutions for pervasive middleware. However, heterogeneous ubiquitous systems also require lightweight distribution platforms that support mobility, adaptability and self-organization. Thus an interface standard for managing the resources of a heterogeneous environment is needed. Seamless use of personal and home networks also requires automatic and scalable service discovery, adaptive resource management and proper authentication of service users. Although some steps have been taken in the development of survivable and self-organizing systems, new algorithms, strategies and negotiation techniques have to be studied and developed. Self-services and intelligent self-configuration may exploit the concepts and constructs of dynamic architectures, which is the issue considered a key enabling software technology in the future.

5. CONCLUSIONS

Future ubiquitous computing systems trigger new challenges for software development that concern software and hardware, as well as their development. Interoperability and adaptability are the most important, but new technologies such as meta models are needed to bridge the gap between heterogeneous technologies. Mobility of services is crucial for nomadic users that would like to use similar services in a similar way, regardless of their location.

In this survey, these challenges were approached from the software architecture point of view, presenting a set of enabling technologies and advanced technologies that can be applied to meet the identified requirements. Although several software technologies applicable to ubiquitous computing environments

already exist, new micro and macro architectures are required for the advanced services required for dynamic resource management, optimized co-ordination, dynamic binding of services and hiding the heterogeneity of execution platforms. Future development efforts will be used for the development of advanced software technologies. It will become apparent in the near future whether the challenges presented in this paper can be solved in different kinds of ubiquitous systems, from personal area networks to cyberworld applications.

6. REFERENCES

- [1] M. Weiser, "The computer for the twenty-first century," *Scientific American*, pp. 94-104, 1991.
- [2] Wireless World Research Forum, *The Book of Visions*, 2001.
- [3] G. Colouri, J. Dollimore, and T. Kindberg, *Distributed Systems Concepts and Design*: Addison-Wesley, 2001.
- [4] A. Taulavuori, E. Niemelä, and M. Matinlassi, "Evaluating the Integrability of COTS Components - the product family viewpoint," in *Building Quality into COTS Components - Testing and Debugging*, S. Beydeda and V. Gruhn, Eds.: Springer-Verlag, 2004, pp. 25.
- [5] D. Garlan, R. Allen, and J. Ockerbloom, "Architectural Mismatch or Why It Is So Hard to Build Systems out of Existing Parts," presented at The 17th International Conference on Software Engineering, Seattle, Washington, USA, 1995.
- [6] H. Schulzrinne and J. Rosenberg, "Application Layer Mobility Using SIP," *ACM Sigmobile. Mobile Computing and Communications Review*, vol. 4, 2000.
- [7] J. Latvakoski, D. Pakkala, and P. Pääkkönen, "A Communication Architecture for Spontaneous Systems," *IEEE Wireless Communications Magazine*, Jun 2004.
- [8] M. Roman, C. Hess, R. Cerqueira, A. Ranganathan, R. H. Campbell, and K. Nahrstedt, "A Middleware Infrastructure for Active Spaces," in *IEEE Pervasive Computing*, 2002, pp. 74-83.
- [9] M. Amin, "Toward Self-Healing Infrastructure Systems," *IEEE Computer*, pp. 44-53, 2000.
- [10] R. J. Ellison, D. A. Fisher, R. C. Linger, H. F. Lipson, T. A. Longstaff, and N. R. Mead, "An Approach to Survivable Systems," CERT Coordination Center, Software Engineering Institute, Carnegie Mellon University 1999.
- [11] M. Matinlassi and E. Niemelä, "The Impact of Maintainability on Component-based Software Systems," presented at Euromicro 2003, Antalya, Turkey, 2003.
- [12] P. P. Pal, J. P. Loyall, R. E. Schantz, J. A. Zinky, and W. F., "Open implementation toolkit for building survivable applications," presented at DARPA Information Survivability Conference and Exposition, 2000. DISCEX '00. Proceedings., 2000.

- [13] B. D. Noble, D. Narayannan, J. E. Tilton, J. Flinn, and K. R. Walker, "Agile Application-Aware Adaptation for Mobility," presented at The 16th ACM Symposium on Operating Systems Principles, Saint Malo, France, 1997.
- [14] D. Pakkala, "Lightweight Distributed Service Platform for Adaptive Mobile Services," VTT Technical Research Centre of Finland, Espoo, Finland, VTT Publications 2004.
- [15] E. Niemelä and T. Vaskivuo, "Agile Middleware of Pervasive Computing Environments," presented at Middleware Support for Pervasive Computing Workshop, Orlando, Florida, USA, 2004.
- [16] F. Heylighen and C. Gershenson, "The Meaning of Self-Organization in Computing," IEEE Intelligent Systems, 2003.
- [17] C. E. Perkins, Ad Hoc Networking, 1st ed: Addison-Wesley, 2001.
- [18] [T. Kindberg and A. Fox, "System Software for Ubiquitous Computing," in IEEE Pervasive Computing, 2002.
- [19] P. Antoiniac, P. Pulli, T. Kuroda, D. Bendas, S. Hickey, and H. Sasaki, "Wireless User Perspectives in Europe: HandSmart Mediaphone Interface," Wireless Personal Communications, vol. 22, pp. 161-174, 2002.
- [20] J. Charles, "Middleware moves to the forefront," Computer, vol. 22, pp. 52-, 1999.
- [21] F. E. Bustamante, P. Widener, and K. Schwan, "Scalable Directory Services Using Proactivity," presented at Supercomputing 2002, 2002.
- [22] S. Vinoski, "Putting the "Web" into Web Services," in IEEE Internet Computing, 2002, pp. 90-92.
- [23] B. Tierney, R. Aydt, D. Gunter, W. Smith, V. Taulor, R. Wolski, and M. Swany, "A Grid Monitoring Architecture," Global Grid Forum - Performance Working Group 2002.
- [24] S. Vinoski, "Service Discovery 101," in IEEE Internet Computing, 2003, pp. 69-71.
- [25] G. Cabri, L. Leonardi, and F. Zambonelli, "Engineering Mobile Agent Applications via Context-Dependent Coordination," IEEE Transactions on Software Engineering, vol. 28, pp. 1039-1055, 2002.
- [26] S. R. Ponnenkanti, B. Lee, A. Fox, P. Hanrahan, and T. Winograd, "ICrafer: A Service Framework for Ubiquitous Computing Environments," 2001.
- [27] H. Xu and S. M. Shatz, "A Framework for Model-Based Design of Agent-Oriented Software," IEEE Transactions on Software Engineering, vol. 29, pp. 15-30, 2003.
- [28] J. P. Sousa and D. Garlan, "Aura: An Architectural Framework for User Mobility in Ubiquitous Computing Environments," presented at The 3rd working IEEE/IFIP Conference on Software Architecture, Montreal, Canada, 2002.

Utilizing Context-Awareness in Office-Type Working Life

Marika Tähti
University of Oulu
P.O. Box 3000
90014 University of Oulu,
Finland
marika.tahti@oulu.fi

Ville-Mikko Rautio
University of Oulu
P.O. Box 4500
90014 University of Oulu,
Finland
ville-mikko.rautio@ee.oulu.fi

Leena Arhippainen
University of Oulu
P.O. Box 3000
90014 University of Oulu,
Finland
leena.arhippainen@oulu.fi

ABSTRACT

This paper presents a context-aware mobile application for office-type working purposes. Via the user tests we have evaluated if this kind of application can support worker's daily life. In the experiments we had real users in real working environment. Our investigation illustrates that workers could utilize context-aware features to ease up their working routines such as keeping presentations. The results of this paper can help designers and developers to envision and implement future ubiquitous devices and environments.

Categories and Subject Descriptors

J.0 [Computer Applications]: General

General Terms

Experimentation

Keywords

Context-aware, mobile application, user experience

1. INTRODUCTION

Weiser [13] has introduced his grand vision of ubiquitous computing over ten years ago. Still we can argue that it is just a vision. Several developers and researchers have tried to envision such a system or environment. So did we. We developed an application which is ubiquitous for a user. It brings services near to user. It brings documents near to user. It decreases disruptions when user is not available. However, is ubiquitous device useful, can it support users' daily working activities? That is what we have investigated.

There are several studies of context-aware applications for guidance purposes mainly. So, Weiser's vision has been evaluated before, for instance, Bellotti et al. [3] have researched what impacts ubiquitous computing systems have to real visitors in actual environment, museums. They have investigated can pervasive computing support a museum-like

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MUM 2004 October 27-29, 2004 College Park, Maryland USA
Copyright 2004 ACM 1-58113-981-0 /04/10 ...\$5.00.

experience. Likewise, Fleck et al. [6] have introduced an electronic guidebook for an interactive museum, called the Exploratorium.

Aittola et al. [1] have developed the SmartLibrary system for the University of Oulu. It is a WLAN-based location-aware (Wireless Local Area Network) mobile library service, and it offers a map-based guidance to find books and collections on a PDA (Personal Digital Assistant) device and mobile phone. The purpose of the service is to support both students of the University and the personnel of the library.

Already many researchers in the field of ubiquitous and context-aware computing have created various demonstrations of applicability of context-awareness to support people's daily lives. Part of this work closely resembles the work that we have done. For example, the CybreMinder by Dey and Abowd as described in [4] or the ComMotion by Marmasse and Schmandt in [8] share some of the concepts with and are closely related to the context-aware prototype we have created for supporting business users in their daily work. However, we are not aware of any work with equally wide set of application components utilizing context-awareness in co-operation in one single prototype, which is tested with real test users.

This paper describes how office-type working life can be supported via context-aware application. By office-type work we mean sort of work were a person has meetings, he uses services like printers and data projectors and he keeps presentations etc.

The paper is organized as follows. The first section introduces reader to research and development area. In the section 2 we describe the application prototype from user perspective not from technical approach. In the section 3 and 4 we want to give the reader a good picture of the conducted experiments by describing them in detail and walking the user through the test phases. The section 5 presents the results of the evaluation. The section 6 summarizes the results and outlines some future work. Finally, the paper is concluded.

2. PROTOTYPE APPLICATION

The prototype is realized as context-aware personal assistant application, which consists of several context-aware services. The components that were implemented and tested with real test users were personal assistant, service browser, context-aware file browser, calendar, context-aware profile manager, and projector service. The prototype has been built over CAPNET architecture which is an architecture developed for distributed context-aware and pervasive ap-

plications. The architecture has been under development now for more than two years and the further development work is still being carried out in CAPNET-project (<http://www.mediateam.oulu.fi/projects/capnet/?lang=en>).

The most important application views used in the testing are shown in Figure 1. Calendar, context-aware file browser and service browser are complex application components containing internally multiple views to provide the functionality, whereas the personal assistant and profile manager contain only a single view. The screenshots, presented in figure 1, are captured from the screen of the PDA running the prototype software.

The personal assistant presents to the user static list of the most important services offered by the application and acts as a shell for accessing different service components in the system. In the user tests personal assistant was configured to present to the user the services which were mentioned above apart from the projector service, which is accessed dynamically through the service browser.

With the **service browser** the user can find service components in a dynamic fashion. The user can either search services by parameters such as location, name and service description out of the set of all services registered within the system, or he can set the service browser to dynamic discovery mode, which means that he is presented with constantly updated list of services which are in his vicinity. If user enters a place recognized by the system, say for example 'meeting room TS335', he is presented a list with the services which have been registered in the specified place. From the service list user may choose and launch a service which he wants to access. Once the selected service gets the request it builds its own user interface on to the user's device by using the facilities provided by the underlying CAPNET architecture, and thereafter the user can interact with the service directly from his device.

The context-aware file browser offers the user a possibility to organize and query his data based on contextual data recognized by the system. User can store data also in the conventional way without context-aware indexing. Indexing takes place when the user is storing the data with the context-aware file browser by the current context perceived by the system or through the calendar in which case the content can be indexed for future contexts. When retrieving data the user may choose to use context-awareness or traditional query. If he chooses to use the context-aware features, he is presented with the list of content related to current context as perceived by the system at the same excluding the content which is not related to current context.

Calendar acts in the prototype as an event organizer which provides also contextual information for the system. Calendar events in the application have three types of information associated with them - the type, time, and place of the event. The type of the event describes types of events, such as 'meeting'. The meaning of time is the period of time when the event occurs and the place of the event is the symbolic name of the place where the event should occur, such as 'cafeteria Datania'.

Context-aware profile manager adapts the profile of the device based on the user's context. For example, when the user enters a meeting, his device may automatically change the profile of his device to meeting mode, if the user has accepted that the profile manager should automate the changing of profile based on the user's context. In an earlier

prototyping round the system was also able to automatically detect recurring patterns in the user's behavior regarding the device profile. The earlier prototype capable of learning user's routines and routine learning algorithm used in recognition are described in [10] and [11]. However, this functionality was not yet integrated into the second prototype during the time when the user tests were carried out on the system and the rules were pre-specified for the system in the prototype which was tested now.

Projector service was implemented as an example of a dynamic service which can be accessed through the service discovery mechanism of the architecture and the service browser. Through the projector service the user can send presentations to and control them on whichever of the projectors which have been registered in the system.

An overview on how the applications concretely behaved during the real user test scenarios will be illustrated in more detail in the chapter 4.

The technical environment for the prototype consisted of a 3870 Compaq iPaq PDA with a WLAN card and IBM's J9 virtual machine, which were used to host the personal assistant application, a server running on Red Hat Linux 9 in the university premises hosting a MySQL database and some of the services required by the underlying architecture, and a WLAN-enabled laptop with Windows XP were used to host the projector service. Prototype was mainly implemented according to PersonalJava 1.2a specification with few native extensions, which were utilized through the Java Native Interface. The network in the test environment was the university-wide WLAN network in Oulu also known as panOULU (public access network OULU) (<http://www.panoulu.net/index.shtml.en>). Test environment was equipped also with Ekahau positioning engine (<http://www.ekahau.com/products/positioningengine/>) to provide accurate positioning of the devices based on WLAN signal strengths.

3. TEST SETUP

In order to make user test it would be beneficial to have fully operational and reliable tool not a prototype. However, preliminary tests in early phase of product development are necessary to perform in order to achieve information about end user's preferences and needs [3]. Therefore we made user tests and the goal was to find out would this kind of ubiquitous application really supports users in their daily work. Moreover, user experience issues were acquired as well as new ideas how this kind of application could be exploited in users' daily life. According to Preece et al. [12] user experience includes the following issues: satisfying, enjoyable, fun, entertaining, helpful, motivating, aesthetically pleasing, supportive of creativity, rewarding and emotionally fulfilling. Instead, usability concerns aspects like efficient to use, effective to use, safe to use, have good, utility, easy to learn, easy to remember how to use [12].

According to Nielsen [9] different user test methods supplement each others. Moreover, Arhippainen and Tähti [2] presents that especially when studying user experience issues, several methods are required in order to get better understanding of users' experiences and emotions. Therefore, we used five different techniques in this evaluation; interview, observation, walkthrough, capturing screen events and emotion collection method.

The main methods were interviews and observation. Test

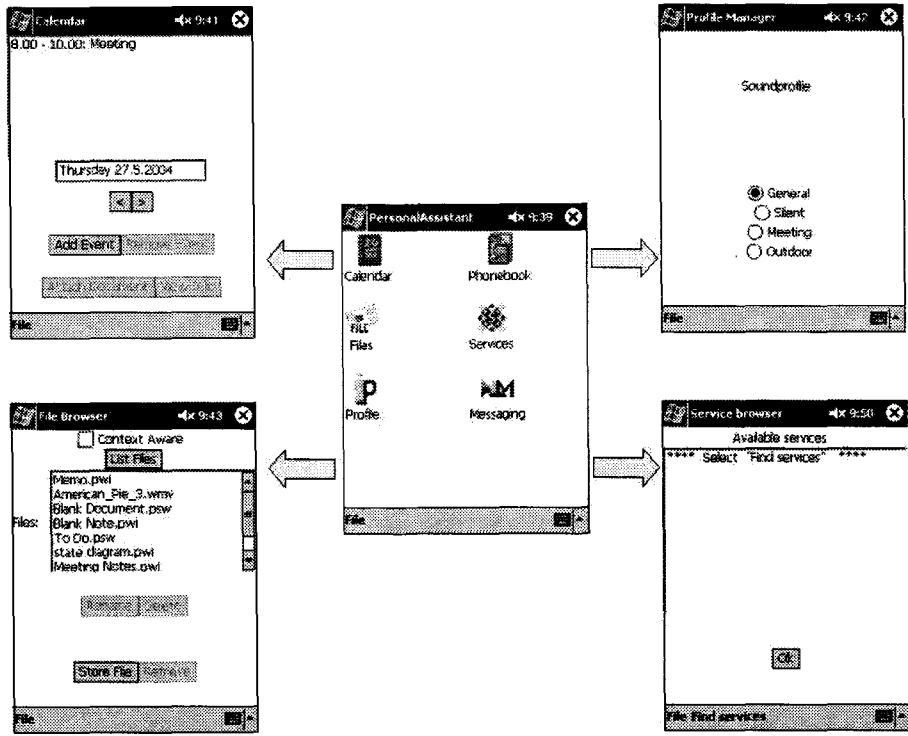


Figure 1: The basic application views. Calendar (upper left), personal assistant (center), profile manager (upper right), context-aware file browser (down left) and service browser (down right).

users were interviewed before and after the use of the application. The questions of the interview considered of user's background, what users think about this kind of application and would it support users in their daily life. The participants were asked to think aloud and observed while using the prototype application. Interviews and observations were recorded by video recorder. Moreover, the PDA screen events were captured on video, to see every action made by the user. In addition, we used SAM (Self Assessment Manikin) method for emotion collection [7]. We used walk-through in order to follow user's actions and maybe advice if necessary. However, we wanted participant to use prototype independently so we offered the user written instructions of the prototype.

The experiment was performed so that one researcher was as a moderator and interviewer, and another was recording the test to the video. One subject performed the experiment at the time and each experiment took approximately 40 minutes.

According to Dumas Redish [5] and Nielsen [9] three to five participants is comfortable in order to notice all problems. Our experiment consisted of one pilot test and five actual test. During the pilot test the experiment settings as well as interview questions were evaluated. After the pilot test there was no need for changes in settings or questions so it was taken into account when analyzing the results.

All test users were male, their age was 33-53 years and they all have university degree. The test users were selected to cover the average users that could need this kind of application. Thus, all test users worked in a position where they have many meetings and that way they could need this

kind of application in their work. Participants have different amount of prior experience with PDA devices. Three of the test users have own PDA device and they use it daily. Two of users had some experiences with such device; they have used it a couple of times but did not have a device of their own. One of the users had not used PDA before but he was very interested in using it.

4. WALKTHROUGH OF THE SCENARIO

This chapter walks the reader through the test scenario which the test users performed during the user tests. The user tests were conducted in an office-type environment (3rd floor of Tietotalo of the University of Oulu).

Figure 2 shows the path in the university premises through which the test users traversed. In the path you can see numbered key points where the application behavior is explained in detail. You can see also the services which were registered to the system in the figure marked with dots with a letter connected to them. They are referred to with the letters during the walk through.

The scenario started in the user's office, which is marked to the figure as key point 1. In the office user added the meeting into the calendar application by defining the time, place and type of the event. He also attached a presentation he had prepared for the meeting by using the file browser application. User enabled also the dynamic service discovery in the service browser.

When meeting was about to start, application notified the user about oncoming meeting. The user then left his office and at key point 2 service discovery dynamically offers

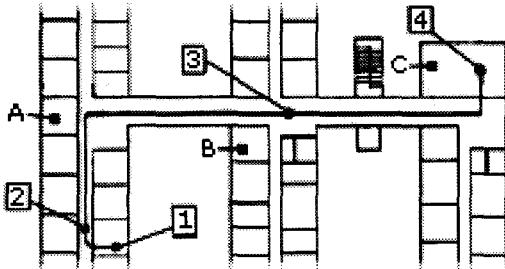


Figure 2: The physical path traversed in the test scenario.

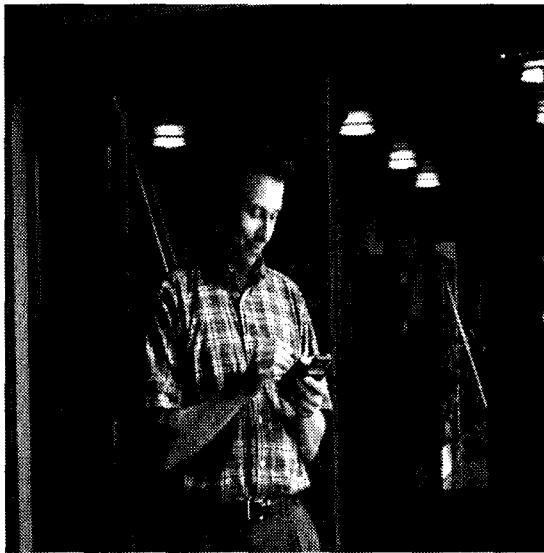


Figure 3: The user is walking to the meeting room and checking the nearest services at the key point 3(.eps format).

services to the user in ordered by distance - that is, first A, then B, and finally C.

As user continued walking towards the meeting room, he arrived at key point 3 (Figure 3), where service discovery offered services to the user in the order B, C, and A, again ordered by distance.

Finally, when the user arrived in the designated meeting room at key point 4 the system recognized that user has entered a place which the system knows by symbolic name, and the service discovery offers only service C (data projector) to the user as it is the only service which is present in the meeting room.

When the user entered the room and the time was appropriate for the meeting, system also recognized that user has entered the meeting context and the profile of his device was changed to meeting mode without any need for interaction.

Then in the meeting the user selected the projector service from the service browser and started it. The user interface of the projector service appeared on the screen of the PDA and then user chose to send a presentation to the projector, which caused the context-aware file browser to be invoked on the device. Now user could choose the context-aware file

browsing option and was presented only the files, which were relevant to the user's current context - the meeting context in this case, which meant that user was presented only with the presentation he had attached to this context back at his office, thus filtering out all the other irrelevant files. Then user chose the presentation he had prepared back in his office and it was transferred to the projector service running on a laptop and finally it appeared on the presentation screen in the meeting room and the user could keep the presentation.

5. USER TEST RESULTS

In this chapter we present the results of the conducted user tests. The point of view is how well this prototype can support users' daily work in office-type environment. In addition, SAM results are presented in order to confirm user experience gathered by interviews and observation. The results of SAM will give some ideas about the emotions elicited by the prototype. Moreover, some new ideas for improving and using the application, presented by the test users, are described.

5.1 Supporting work

All of the test users liked the idea of the prototype. For almost all users the application would be very useful and supportive during normal working hours. Test users mentioned that this kind of application would make working life easier, and reduce stress because they would not need to carry so many devices, disks etc., and worry whether they have all documents along with them.

5.2 New ideas

One of the most positive results from user testing of prototype was that the test users gave lots of new ideas and proposals for improvement. Most of these proposals came during the interview, but some during test period, when users made suggestions what would make the prototype serve them better. Subjects told what application might be useful in meetings and, for example, in school environment.

Additional and useful features for meetings could be: one shared and synchronized calendar, where one would see all available times for keeping a meeting. The slides could be seen on the PDA screen and there could be possibility to draw something on the slides and everybody in the meeting could see the changes. It would also be useful to use PDA for transmitting files to colleague's computer and printer. Also accessibility to documents from anywhere (even abroad) anytime was seen as a useful feature.

This kind of application would make extempore ideas possible, without worrying whether one would have that and that slideshow or not. This kind of application could really profit the school system, too. It could make the teachers work easier, for example, by offering mobile and direct access to class room reservation information and material in the warehouse. Thus, there would be no need to go and pick up some papers and make class room reservations or materials, like slides, from the warehouse. This kind of application will not only make the teachers' work easier but also save time for other activities like teaching or preparing lectures.

5.3 SAM results

SAM method was used to collect emotions elicited during the use of the prototype. The results collected by SAM are presented in Table 1. In the Table 1 the scale of plea-

Table 1: SAM results and users' prior experience with PDA

User	PDA experience	Pleasure	Arousal	Dominance
1	active user	7	5	6
2	active user	6	4	4
3	active user	6	5	7
4	some experience	6	7	8
5	some experience	7	6	8
6	not used	9	8	3
mean		6,8	5,8	6

sure, arousal and dominance are as follows: **Pleasure:** 1-9 (unhappy- happy), **Arousal:** 1-9 (sleepy - aroused) and **Dominance:** 1-9 (low- high).

From the Table 1 we can see that the application elicited quite positive emotions and that all users find the application quite pleasurable. The mean value of every emotional dimension was over the average value (5). Even though there were some test users that rated the arousal and dominance below the average value.

The SAM results seem quite positive, but we must remember that the test users were filling the SAM questionnaire in front of the testing personnel and this may affected to the final results. We can assume that the results are slightly too positive, than what they would be without the presence of testing personnel.

The SAM results support the results that were collected via interviews and observations. For example, those test users that were confident and independent while using the application selected high values for dominance and those having some troubles selected lower values.

6. SUMMARY AND FUTURE WORK

In this chapter we summarize user experience results and also present some future features for context-aware services. The Table 2 presents how users regarded that application can support their work. Also user's worries, needs and wishes are described. We identified usability and user experience issues from users' comments. Those issues are presented in the Table 2. according to Preece et al. definition [12]. In addition, some comments did not clearly fit the definition, but are very important from user point of view.

The Table 2 depicts that users find a lot of different situations where they would need context-aware mobile application. Some of the needs are related to technical requirements of office-type working life e.g. "I can draw something on the slides in PDA and all in the meeting could see the changes in the slides".

Moreover, some of the comments illustrate that the user's work is stressful and he need some support for that: "Do not have to worry about all documents".

During this study we found several improvements ideas for such context-aware mobile application. Some ideas related to components of application, for example, for calendar we identified such improvements: Share the calendar with colleagues, synchronized the calendar automatically, search and propose group meeting times, search an available room automatically.

Automatically changing profile (the profile manager) would be efficient to extend to cover more wide area e.g. meeting room, lunch room, own office, lecture room.

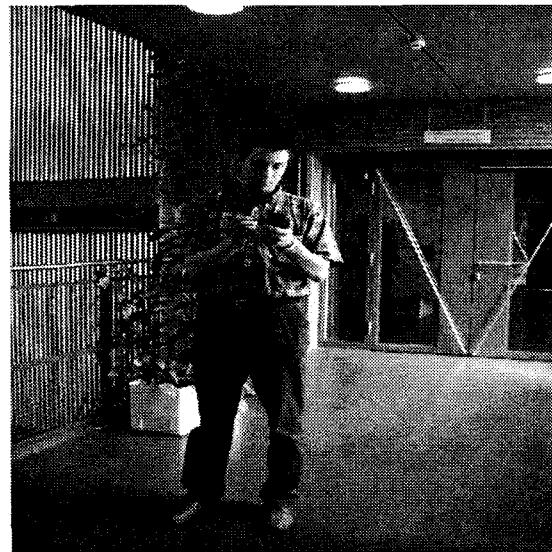


Figure 4: The user had to stop when checking the nearest services.

The context-aware file browser could be more aware of context. This means that it could transmit the file automatically to the right service (e.g. from the PDA to the data projector). Moreover, the component could remind the user the documents that he had to update or create based on context and calendar information. Also, the document can be shared automatically with all people in the meeting.

The service browser could use different graphical interface, for example, it could show the services on the map or other graphical view but not as a list. Sometimes it was difficult for the user follow the available service when walking at the same time (Figure 4). In addition, the component could provide some guidance to the user how to find the service. Guidance could be implemented by voice, text, graphic or vibration (Figure 4). In addition, service browser could include more services but they can be shown on the screen according to user profiles or needs.

The prototyping work will be carried on and in the upcoming versions of the prototype our purpose is to provide the applications with more versatile context information. In addition to the position, time, and user-generated events, which were context types utilized in this version of prototype, we have planned to include acceleration sensors on the device, temperature and other weather sensors available in the local scale in the Oulu region. This is enabled by CAP-NET architecture which allows us to easily introduce new types of sensors in the system.

Also we are looking for introducing more services for common routines which are accessed in the dynamic way by the user as presented earlier in the article. In addition the interpersonal communication, collaboration and sharing in the system will be improved drastically and also context-aware features related to the communication will be introduced.

And finally we are planning to port the system on Symbian environment and utilize the context information and inference provided by the system in widely used 3rd party applications, such as MS Word or OpenOffice, to bring the features closer to the real-life of the users.

Table 2: The summary of the user test results (S=Support, W=Worries, N=Needs and wishes)

Comments	usability issues	user experience issues	misc issues
S: It would be useful in work life	utility		
S: It would easier work life	utility		
S: It would reduce stress		helpful, supportive	
S: Context-aware profile changing is good		helpful	
W: Can I use the application everywhere			availability
W: Does it works always as it should	effectiveness		
N: A quick connection to the data projector	utility		
N: An easy connection to the data projector	efficiency		
N: Shared and synchronized calendar	efficiency	supportive	
N: Prefer a mobile phone than PDA	easy to learn		
N: Do not have to worry about all documents		supportive	
N: I can see slides on the PDA screen	efficiency, easy to use		
N: Can I draw something on the slides in PDA	efficiency	supportive	
N: All in the meeting could see the changes in the slides	efficiency		
N: To use PDA for transmitting files to colleagues	efficiency		
N: Accessibility to documents from anywhere	efficiency	supportive	availability

7. CONCLUSION

In this paper we have presented the context-aware mobile application for supporting work life in the office-type environment. In addition, we have described user experience results that we have obtained via experiments. Six office-type employees used the context-aware mobile application in real environment which includes workrooms, meeting rooms, corridors and services e.g. printers.

Based on our findings we can argue that office-type work can be supported via context-aware mobile application. Especially supportive characteristics are very welcome to user's daily working life. The context-aware application could easier user's daily routines by taking care of user's presentation slides and other documents. It would decrease stress if a person does not need to worry about all necessary documents and devices e.g. laptop. However, context-aware adaptive devices have to enable to the user a dynamic working environment. This means the user has to have availability make changes in to documents whenever and wherever he wants. This requirement is close to Weiser's vision of the availability of technology.

In the future, it would be interesting to increase context-awareness. Application could adapt according to user's place, moment, activity, mood, needs, and pre-defined profile, etc. Moreover, context-aware application could take into account context-switching which means that the user moves one context into other smoothly. Therefore, the adaptive device should be able to behave the same way.

8. ACKNOWLEDGEMENTS

We wish to thank CAPNET team for their work with the prototype, and all the test users who participated to the test. This work was done within the CAPNET research program funded by TEKES. In addition, warmth thanks to Academy of Finland funded ADAMOS project as well for the opportunity to carry out these experiments.

9. REFERENCES

- [1] M. Aittola, T. Ryhanen, and T. Ojala. Smartlibrary - location-aware mobile library service. In *Mobile HCI Proceedings*, pages 411–416. Springer, September 2003.

- [2] L. Arhipainen and M. Tahti. Empirical evaluation of user experience in two adaptive application prototypes. In *MUM Proceedings*, pages 27–34, December 2003.
- [3] F. Bellotti, R. Berta, A. D. Gloria, and M. Margarone. User testing a hypermedia tour guide. *IEEE Pervasive Computing*, 1(2):33–41, April-June 2002.
- [4] A. K. Dey and G. D. Abowd. Cybreminder: A context-aware system for supporting reminders. In *HUC Proceedings*, pages 172–186, September 2000.
- [5] J. S. Dumas and J. C. Redish. *A practical Guide to Usability Testing*. Intellect, Ltd, UK, 1999.
- [6] M. Fleck, M. Frid, T. Kindberg, E. O'Brien-Strain, R. Rajani, and M. Spasojevic. From informing to remembering: Ubiquitous systems in interactive museums. *IEEE Pervasive Computing*, 1(2):13–21, April-June 2002.
- [7] P. J. Lang. Behavioral treatment and bio-behavioral assessment: Computer applications. In *J. B. Sidowski, J. H. Johnson, T A. Williams (Eds.), Technology in mental health care delivery systems*, pages 119–137. Norwood. NJ: Albex, 1980.
- [8] N. Marmasse and C. Schmandt. Location-aware information delivery with commotion. In *HUC Proceedings*, pages 157–171, September 2000.
- [9] J. Nielsen. *Usability Engineering*. AP Professional, USA, 1993.
- [10] S. Pirttiangas, J. Riekki, S. Porspakka, and J. Roning. Know your whereabouts. In *CNDS Proceedings*, January 2004.
- [11] S. Pirttiangas, J. Riekki, and J. Roning. Routine learning: Analysing your whereabouts. In *ITCC Proceedings*, pages 208–212, April 2004.
- [12] J. Preece, Y. Rogers, and H. Sharp. *Interaction Design, Beyond human-computer interaction*. John Wiley Sons, Inc., USA, 2002.
- [13] M. Weiser. The computer for the 21st century. *Scientific American*, 265(3):94–104, December 1991.

Towards Connectivity Management Adaptability: Context Awareness in Policy Representation and End-to-end Evaluation Algorithm

Jun-Zhao Sun, Jukka Riekki, Jaakko Sauvola, and Marko Jurmu

Department of Electrical and Information Engineering
P.O.Box 4500, University of Oulu
FIN-90014, University of Oulu, Finland

{ junzhao.sun, jukka.riecki, jaakko.sauvola, marko.jurmu } @ ee.oulu.fi

ABSTRACT

An infrastructure based on multiple heterogeneous access networks is one of the leading enablers for the emerging pervasive and ubiquitous computing paradigm, in which the optimal management of diverse networking resources is a challenging problem. This paper presents a context-aware policy mechanism and related end-to-end evaluation algorithm for adaptive connectivity management of multi-access wireless networks. A policy is used to express the criteria for adaptive selection of the best local and remote network interfaces. The best connection can then be used for the establishment of a channel as well as the maintenance of on-going data transmission. Rich context information is considered in the policy representation with respect to user profile and preference, application characteristics, device capability, and network QoS condition. The decision of the best access networks to be used is made on the basis of an end-to-end evaluation process. The decision can be made in both Master-Slave and Peer-to-Peer modes. The paper focuses on the methods for policy representation and connection evaluation algorithm. A case study is presented to show the usability of the proposed policy mechanism and decision-making algorithm in the adaptive management of heterogeneous networking resources.

Categories and Subject Descriptors

C.2.1 [Computer-Communication Networks]: Network Architecture and Design – wireless communication

General Terms

Algorithms, Management, Design

Keywords

Heterogeneous Networks, Connectivity Management, Context Awareness, Policy.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MUM 2004, October 27-29, 2004 College Park, Maryland, USA.
Copyright 2004 ACM 1-58113-981-0 /04/10... \$5.00

1. INTRODUCTION

Recent advances in both portable devices and wireless networks make mobile computing a reality. Embedded and invisible computing resources are paving the way to a new paradigm known as *pervasive and ubiquitous computing*. A next generation mobile communication system is emerging with heterogeneous wireless access networks as one of the leading features. Network connectivity with varying quality of service (QoS) must be offered anytime and anywhere for seamless provision of ubiquitous services. The requirement of persistent connectivity in multiple wireless accesses leads to the problem of effectively managing and synthetically using multiple wireless resources.

A wide range of wireless network technologies can be used for mobile internet access including e.g. Bluetooth, WLAN, 2G(GSM), 2.5G(GPRS), 3G(UMTS), etc. Wireless access networks greatly vary by nature, with regard to e.g. data rate, coverage, subscriber volume, supported mobile velocity, anti-interference, and suitable transmitting environment. Moreover, wireless network QoS parameters vary dynamically over time, in terms of reliability and availability, bandwidth, delay, jitter, response time, and packet loss rate. Finally user mobility leads to continuous changing of location and environment, network operator and service provider, and access networks. Handoff is one main source of network variations in terms of packet delay and packet loss. Nowadays, mobile terminals equipped with multiple network interfaces are common. It has been widely recognized that no single access technique can fulfil all the requirements. New methods are needed to manage multiple access systems so that diverse communication resources can form wireless connections in a simultaneous, collaborative, and complementary way.

Current QoS models [1] are mainly oriented towards low-level network parameters such as bandwidth, latency, and jitter, targeting to provide transparent management on a network transport system to upper applications. However, in multi-access environment the first order QoS issue above all is end-to-end selection of access technologies to be used for various data flow among end-to-end applications. This selection should be based on an evaluation process that can be launched at the time of connection establishment as well as when special events occur, like a new and better connection is available when a user enters the overlapping area. The selection of the best current connection for a particular traffic flow triggers the flow level vertical handoff

from the old access network to the new one. This decision is critical and requires taking rich contextual information into consideration, e.g. device capacity, network condition, application characteristics, user preference, etc.

This paper presents a novel policy-based mechanism for context-aware connectivity management for multi-access networks, with the emphasis on the methods for policy representation and evaluation-making algorithm. The policy is represented as a 4-tuple, including the direction and class of traffic, requirement expression, and concrete evaluation items. Both local and end-to-end context information can be utilized in the policy representation. Three steps are involved in the evaluation process, namely policy traverse, cost matrix calculation, and decision-making. In particular two modes for connection decision are presented as Master-Slave and Peer-to-Peer, in which the functionality of end-to-end negotiation between peer hosts is provided. The rest of the paper is organized as follows: Section 2 reviews related handoff algorithm works. Section 3 presents the architecture. Section 4 describes policy representation methods. Section 5 discusses the detailed evaluation-making process. A case study is presented in Section 6. Finally, Section 7 concludes the paper with a remark on future works.

2. RELATED WORKS

Traditional algorithms of homogeneous access evaluation for horizontal handoff criteria focus mainly on link quality conditions, e.g. radio signal strength, SNR (Signal to Noise Ratio), frame error rate, base station workload, etc. These factors are also taken into account in vertical handoff. Moreover, attention is paid more and more to high level context information including e.g. user preference, cost, application feature, device capacity, bandwidth, etc.

In [2] an algorithm for handoff between WLAN and CDMA2000 cellular network is proposed. Traffic is classified into real-time and non-real-time services. Tradeoff between handoff delay and throughput is considered. Parameters include RSS threshold and continuous beacon signal. The beginning of the handoff is decided by the handoff delay time and throughput according to traffic classes.

Two policies for vertical handoff were proposed in [3]. A method based on a data-driven policy is to choose the network interface that best fits the requirement of the data flow. Parameters for describing data flow include destination, throughput, reliability, and fixed IP source address. A link-driven policy considers two classes of parameters. Internal parameters are maximum link speed and reliability; external parameters are billing, cost, and power utilization. User preferences are also considered.

A policy-based mobile IP handoff decision algorithm is presented in [4]. A Generic Link Layer (GLL) is defined above access networks. A general list of parameters is then defined including the information about link, environment, neighbourhood and link layer management. The algorithm can then utilize link layer information from a defined GLL as input values. A first implementation of the algorithm is described in which the SNR is considered in the policy.

In [5] PROTON is presented to allow seamless connection to highly integrated heterogeneous wireless networks. Networking context fragments are grouped into dynamic and static components. Dynamic components include presence, status,

signal strength, congestion, flows, velocity, position, etc. Static context components provide steady data including the profiles of network, application, user, and infrastructure.

A policy-enabled handoff system is presented in [6] for heterogeneous wireless networks. The system allows users to express policies on what is the “best” wireless system at any moment, and make tradeoffs among network characteristics and dynamics such as cost, performance and power consumption. In particular a policy model based on cost function is presented and experimented.

A handover decision making process is presented in [7] which uses context information regarding user devices and their capabilities, user personal context information, application QoS requirements and user perceptibility of application QoS, user location, network coverage and network QoS. The decision-making process evaluates the context information to decide whether handover is necessary, to which network, and whether any additional adaptations should be applied.

In [8] the issue addressed is how to make use of multiple network interfaces simultaneously and to control the interface selection of both outgoing and incoming packets for different traffic flow. The method used is to extend the base Mobile IP protocol to control the choice of interfaces to use for incoming traffic to a mobile host.

A new handover procedure is proposed in [9] by incorporating the Mobile IP principles in combination with fuzzy logic. During the handover initiation, information on the user profile, QoS perceived by the user, and radio link availability are collected. The selection of the most suitable target segment depends mostly on the user profile containing information such as the minimum and maximum cost and the list of segments with the highest and lowest priority.

A novel mobility management system is proposed in [10]. The system integrates a connection manager (CM) and a virtual connectivity manager (VC). A roaming decision maker and a context database are introduced to act as the interconnection between CM and VC. The context consists of user preferences and technical parameters, such as access delay, available bandwidth, and capabilities of the terminal.

A vertical handoff scheme is presented in [11]. An upward handoff is initiated when several beacons from the current overlay network are not received. Downward vertical handoffs are initiated when several beacons are heard from a lower overlay’s network interface. For latency sensitive applications some enhancements can be used by taking some alternative hints into account to predict a handoff.

In [12] a smart decision model is proposed to smartly perform vertical handoff. The proposed model can properly execute handoff to the “best” network interface at the “best” moment according to the properties of available network interfaces, system configurations/information, and user preferences. In particular the score function is broken down to usage expense, link capacity, and power consumption.

Compared with these related works, our research focused on the policy mechanism for a channel-based flow level vertical handoff. Simultaneous usage of multiple access networks is supported for different application categories, data flow classes, channels, and transmission directions. Policies are defined at three levels: user,

application, and channel. A rich set of context information is taken into consideration including local host connectivity context, network QoS and user-perceived QoS parameters, application characteristics, as well as user preferences. Static, prioritised, and weighted policies can be defined to fulfil various demands on adaptation. Moreover, in the handoff decision-making process, the evaluation and decision can be made not only locally, but also in an end-to-end fashion. The handoff decision can be made with the handoff algorithm in a master-slave mode or peer-to-peer mode. To our knowledge this is the first research enabling handoff negotiation and decision jointly between peer hosts.

3. ARCHITECTURE

3.1 Channel and Connection

The architecture of Connectivity Manager (CM) is illustrated in Figure 1, with an end-to-end example. Each end device contains a CM for adaptive management of the local connectivity resource. Cooperation between CMs residing in peer hosts enables end-to-end management adaptation. This adaptation is achieved through adaptively maintaining a channel's life cycle, including channel creation and I/O operation, event processing, and channel closing.

Channel is defined as an abstract concept. A *channel* is a logical end-to-end link between physical application components that are located in separate network devices, as shown in Figure 1. A channel may be unidirectional or bi-directional, depending on the allowed data transmission direction. Moreover a channel can be set with a traffic class to denote the feature of the data transferred through the channel.

Connection is the physical corresponding of channel. A *connection* is an end-to-end link between two network interfaces of the two separate peer hosts, as depicted in Figure 1. In case of a multi-access system there might be a candidate connection set available for a logical channel. At the moment each channel uses a specific connection to transfer a specific data flow between the two devices. The connection is formed between specific network interfaces of local and remote hosts.

With respect to adaptive connectivity management the adaptation functionalities can be divided into three categories for the support of three different situations in channel maintenance, as follows.

- *Network interface selection*. The application requests data transport service by opening a new channel. If multiple connections are available, the best available connection at the moment will be selected from the candidate connection set for the channel creation. A set of rules must exist to determine “the best” connection to be used.
- *Connection switching*. During the period when the application is operating on data transmission with the channel, network status is subjected to change. For example the mobile device may move into or out of the overlapping area of multiple accesses, or to another access subnet. In this case the connection used by the channel can be seamlessly switched without being perceived by the application, to ensure that the best connection is always used at any time.
- *Disconnection treatment*. During the period that no valid network connection is available, input and output requests of the application should be suspended. The operations should be resumed later when one qualified interface becomes available.

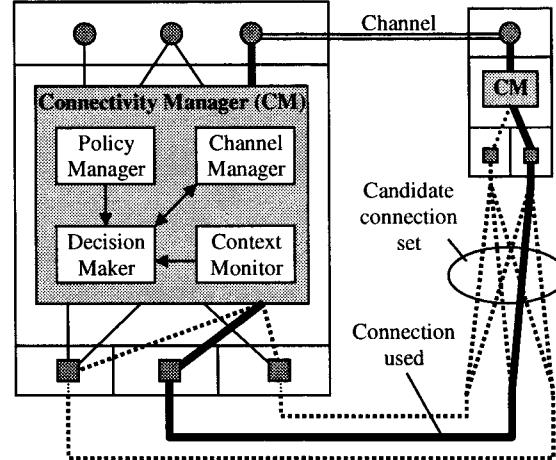


Figure 1. Connectivity manager architecture.

As shown in Figure 1, the adaptive management of multiple network accesses is made in the architecture at channel level instead of device level. Hence at a given moment any application thread in one user terminal can utilize its own channel relying on a particular network access. In this way simultaneous usage of multi-access networks to one user device is possible. For example while another application is downloading content from the web using WLAN connection, a message application can start to use GPRS simultaneously. The key elements as well as their relationship are illustrated in Figure 2. The labels on the arrows indicate the numerical relations between the two elements. There could be multiple channels that are simultaneously active in one user device, belonging to a single application or different ones, each of which depending on one particular connection. Detailed description of the channel mechanism of the architecture is presented in our previous studies [13]. Figure 2 also shows another interesting adaptation, i.e. one user may use multiple devices on which one particular application may be executable. This leads to another level of adaptation, i.e. the adaptive selection of the most suitable device for a particular application. Although the issue of device adaptation is out of the scope of this paper, our approach forms a good framework to address the issue.

3.2 Decision Maker

As shown in Figure 1, the core component of the CM is the Decision Maker. It takes policy and setting, channel information, and related context information as three types of input parameters, and returns concrete decisions as output. According to the analysis in Section 3.1, there are three different decisions in total to be made, including:

- a. During the channel creation, which connection from the candidate connection set will be selected?
- b. When network changes occur, should the channel's connection be switched to another one? If so, to which one?
- c. Should the operation of the channel be suspended? That is, should the disconnection treatment be employed?

The three questions can be combined into one single question: *to a channel under concern, which one is the best connection at the*

moment? In other words the return value of the Decision Maker is “the best connection for the channel at the moment”. So the task of the Decision Maker is essentially a process of connection evaluation. Moreover, the evaluation is made on a channel basis. That is, the returned best connection may be different for a different channel. Detailed discussion about the algorithms used in the evaluation process is presented in Section 5.

The output of the Decision Maker is then provided to the Channel Manager. Thus during the channel creation the returned one will be utilized as the underlying connection. When event occurs the Channel Manager will make a comparison, between the currently used one and the one provided by the Decision Maker, to determine whether switch should be initiated. In case the output is null, which means no valid connection is available for the channel, disconnection treatment has to be employed.

The evaluation process can be automatically made in a periodical fashion, or immediately when one of the following events occurs.

- Device events, including network interface up or down, channel established or terminated, power status changed, and device policy reset.
- Application event, i.e. application policy reset.
- Channel events, including channel creation, QoS out of boundary, traffic class reset, and channel policy reset.

3.3 Channel and Context Information

Policy and setting is one type of the input parameters of the evaluation process, which is introduced in detail in Section 4. Basically a policy is defined by user or application to describe the requirements on the connection and the rules of how to select the optimal one. The other two types of input parameters are channel information and context information.

A channel may be set as unidirectional or bi-directional at the point of channel creation, depending on the allowed data transmission direction. The direction value of a channel can then be *in*, *out*, or *inout*. Moreover a channel can be set with a traffic class to denote the feature of the data transferred through the channel. Both application level and flow level traffic classes can be defined for a channel, as shown in Figure 2. First a channel may inherit the traffic class from the application that owns it. Application level traffic class can be used to specify the QoS category of an application, in which the value can be *bulk transfer*, *interactive*, *responsive*, *real-time*, *bandwidth intensive*, or *network control*. Each of the value can also be treated as a subcategory for which the final values can be defined to enable further application classification. For example for *interactive* subcategory, final values including *VoIP*, *Gaming*, and *Conferencing* can be defined.

Flow level traffic class can be defined by referring to the QoS classes in different domains including:

- IP domain. TC value can be any Type of Service (TOS) defined in [14] including *minimize delay*, *maximize throughput*, *maximize reliability*, *minimize monetary cost*, and *normal service*.
- UMTS domain. The UMTS QoS classes [15] are specified depending on delay sensitivity of the user data traffic used by certain applications, including *conversational*, *streaming*, *playback*, and *background*.

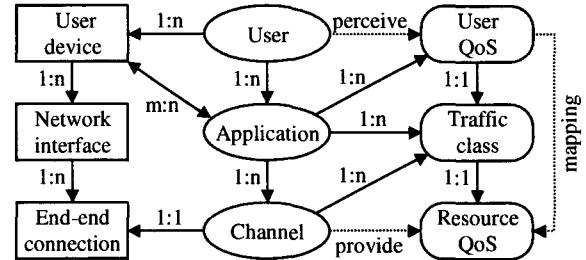


Figure 2. Elements and relationships.

In addition, specific categories and values of traffic class can be defined on both application and flow level, as long as the channel is able to treat with the defined ones.

Context monitor is used to collect timely values for all the related context factors that serve as the input parameters for decision-making. There are both local context factors and end-to-end context factors. Local factors are about local host connectivity context information. The local context factors considered by the policy representation include operator, type, cost, coverage, speed, duration, input error rate, input discard rate, output error rate, output discard rate, power consumption, signal strength, and workload.

End-to-end (E2E) context factors are end-to-end parameters of both network QoS and user QoS. Network QoS parameters include up bandwidth, down bandwidth, RTT (Round Trip Time), and jitter. It is provided by channel. User-perceived QoS parameters allow users to express their perception of QoS for multimedia applications, and are thus closely related to the media types under processed. User QoS parameters include picture detail (pixel resolution), picture colour accuracy (colour bits per pixel), video rate (frame rate), video smoothness (frame rate jitter), audio quality (sampling rate and number of bits) [16]. It is assumed that any user perceived QoS can be mapped into a set of network QoS requirements [17], as illustrated in Figure 2.

4. POLICY REPRESENTATION

Policy is used to express the requirements and criteria according to which the determination of both local and remote network interfaces can be made for the establishment of a new connection as well as the maintenance of on-going data transmission. A policy P is a 4-tuple and defined as:

$$P = (Di, TC, RE, EI).$$

A policy can be specified for the communications of a specific traffic class in a specific direction. The optional element Di is the direction of the data transmission, which can be the value of *in*, *out*, or *inout*. This is to cope with the asymmetry between uplink and downlink. If in a policy the Di is not presented, then it means the policy is applicable to any traffic direction (i.e. any matching). Di is to be used for the matching with the channel's direction attribute.

TC is an optional element of the policy used to specify the traffic class of the transferred data. TC can be defined at two different levels, as channel's traffic class definition in Section 3.3. If in a policy the TC is not presented, then it means the policy is

applicable to any traffic class (i.e. any matching). *TC* is to be used to match with channel's traffic class attribute.

The optional element *RE* is the requirement expression of a policy, in which a set of requirements for the concerned data transmission is specified. If in a policy the *RE* is not presented, then the policy is applicable to any connection, i.e. all the connections are qualified. The *RE* takes individual connection as argument and is then used for filtering all the connections leading to a qualified set of connections to be considered. A *RE* is a logic expression with a set of relation expression connected with *NOT*, *AND*, and *OR*. Each relation expression consists of three items: *context factor*, *relation operation*, and *value*. Relation operations include equal to, not equal to, greater than, less than, greater than or equal to, less than or equal to. Value is constant and specific to different context factors and can be of any type like Boolean, integer, real, float, character, string, etc. Context factor is any local or E2E factor defined in Section 3.3

The policy evaluation items *EI* are used to construct the real policy. There are three types of policy, namely *static policy*, *priority policy*, and *weight policy*, with different evaluation items defined. Static policy can be represented as:

$$(use/default, type/index, value),$$

where concrete network interface type or index is explicitly specified or set as default. These are called *use policy* and *default policy* respectively. Connection index is the unique identifier of each network interface. Network interface types include PPP, Ethernet, GPRS, WLAN, Bluetooth, etc. Priority policy items are a set of *(type/index, value)* pairs, with each of them denoting the integer priority value of a particular interface or interface type.

Weight policy's *EI* are a set of *(factor, weight)* pair as:

$$(f_1, w_1)(f_2, w_2) \dots (f_n, w_n), \sum(w_1, w_2, \dots, w_n) = 1.$$

Each item denotes the decimal weightiness value of the corresponding factor. Basically all the context factors used for *RE* can also be used as weight policy's *EI* factors, as long as there is a corresponding numerical value related. For example the operator or type of the network interface can also be used as a factor if there are related priority values defined for them. Weight policy enables the selection of connection in run-time.

Three different policy scopes can be defined for the representation of connection selection rules at different levels. A *device level policy* is usually set by the user to express personal preference on the usage of the network connections of the whole user device. Application level *TCs* and user-perceived QoS factors are mainly used for device level policy. At the same time each individual application can also set its own connectivity management rules by *application level policy*. Basically the application may provide an interface for user/other applications to set its policy, or maintain its policies internally without change. Finally *channel level policy* can be set by the application when a channel is created for data transmission.

Accompanying with policy definition, there is also a setting of channel end type specified by user or application. It will be used

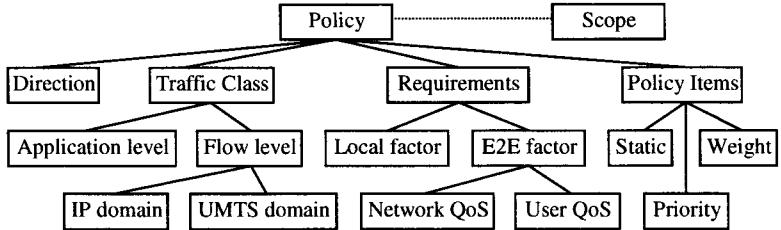


Figure 3. Policy representation outline.

in the decision-making process and is described in Section 5.3 in more detail.

Figure 3 illustrated the outline of the policy representation. A detailed BNF-based formal representation of policy is presented as follows.

```

policy ::=      ( [Di, ]/[TC, ]/[RE, ]/EI )
Di ::=          in | out | inout
TC ::=          IPTOS | UMTSClass | AppTC
IPTOS ::=       minimize delay | maximize throughput |
                maximize reliability | minimize monetary cost | normal service
UMTSClass ::=   conversational | streaming | interactive | background
AppTC ::=       bulk transfer | interactive | responsive | real-time | bandwidth intensive | network control
RE ::=          NOT RE | RE AND RE | RE OR RE | RelExpr
RelExpr ::=     ( ContextFactor RelOp Value )
RelOp ::=       = | < | > | <> | >= | <=
ContextFactor ::= LocalFactor | E2EFactor
LocalFactor ::= operator | type | cost | coverage | speed |
                duration | input error rate | input discard rate | output error rate |
                output discard rate | power consumption | signal strength | workload
E2EFactor ::=  networkQoS | userQoS
networkQoS ::= upBandwidth | downBandwidth | RTT | jitter
userQoS ::=    resolution | accuracy | frame rate | frame jitter | sampling rate
EI ::=          StaticPolicy | PriorityPolicy | WeightPolicy
StaticPolicy ::= ( use | default, Index | Type, Value )
ConnType ::=    ppp | Ethernet | GPRS | WLAN
PriorityPolicy ::= PriorityItem {; PriorityItem }
PriorityItem ::= ( ConnIndex | ConnType, Value )
WeightPolicy ::= WeightItem {; WeightItem }
WeightItem ::=  ( ContextFactor, Value )
  
```

5. EVALUATION ALGORITHM

The evaluation process is a 3-step process, described in detail as follows. To a channel under concern, the evaluation process is initiated at the host in which an event occurs to be cooperatively used at both two sides (hosts) of the channel. Figure 4 illustrates the block diagram of the evaluation algorithm. Besides peer information there are three main input parameters: channel information, policy set and setting, and context information.

5.1 Policy traverse

The purpose of the policy traverse is to go through all the related policies in order to find a most matching policy (MMP). MMP is defined as the policy with the highest matching value (MV). The

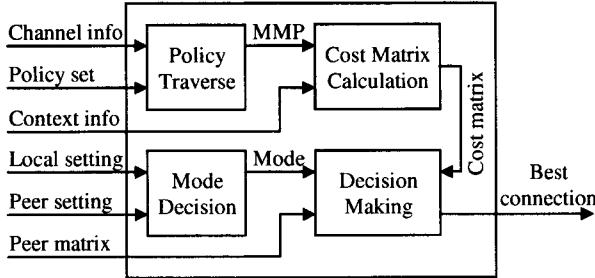


Figure 4. Block diagram of evaluation process algorithm.

traverse is made in the order of policy priority. Two orders of policy priority can be defined. First order priority is defined with policy scopes, as

device>channel>application.

In this order only the *Di* and *TC* elements of a policy have to be checked to investigate whether the policy is a matching one. To policies with the same first order priority, the most matching one will be used. Second order priority is defined with evaluation items, as

use>priorit>weight>default,

to be used when multiple first order policies have the same matching status. Default policy has the lowest priority and will be checked when no any other applicable policy exists. Matching value is assigned to a policy by comparing a policy's *Di* and *TC* with related channel information. The order of different matching cases is defined as:

exact matching > any matching > no matching.

Exact matching means the specified value of the policy equals that of the channel. Any matching means the related policy element is not presented. Detailed algorithm for policy traverse is as follows, which returns the MMP to be used in cost matrix calculation.

- Get channel information including transmission direction, traffic class, and channel policies;
- Calculate MV for each policy of the current priority, started from device policy;
- If one matching policy is found, return it as the MMP;
- If a set of matching policies is found with only one of the highest MV, return the one as the MMP;
- If a set of matching policies is found with more than one of the same highest MV, return the one of the highest second order priority;
- If no matching policy is found, go back to b with next level priority;
- If no matching policy is found, exit and leave the decision to the operating system.

5.2 Cost Matrix Calculation

In Section 5.1 policy traverse an MMP is returned. Then a cost matrix *C* is generated for the channel according to the selected MMP. Suppose a channel exists between host A and host B. The number of network interfaces of A is *m* and of B is *n*. If any E2E

context factor is used in A's MMP (*RE* or *EI*), then host A generates an *m*×*n* matrix *C* as:

$$\begin{bmatrix} c_{11}, c_{12}, \dots, c_{1n} \\ c_{21}, c_{22}, \dots, c_{2n} \\ \vdots \\ c_{m1}, c_{m2}, \dots, c_{mn} \end{bmatrix},$$

where *c_{ij}* is the cost of the case when the channel is built upon A's interface *i* and B's interface *j*. In case only local context factors are used in A's MMP an *m* vector will be generated as

$$C = (c_1, c_2, \dots, c_m).$$

Obviously in the vector each element directly gives the cost of the corresponding network interface without considering a remote interface to be used. Similarly, host B may generate an *n*×*m* matrix or an *n* vector. To ease the processing in the decision-making process, an extension should be made to the matrix if a vector is generated. An *m* vector *C* can be extended to an *m*×*n* matrix by:

$$\text{For } i \in [1..m] \text{ and } x \in [1..n], c_{ix} = c_i.$$

A detailed algorithm for cost matrix calculation at host A is as follows.

- Collect related local and E2E context information;
- If MMP's *RE* and/or *EI* concerns E2E context factors, the for each interface *j* of host B (*j* in *[1..n]*), do *c* to generate the column *j* in the *m*×*n* cost matrix *C*; else do *c* with neglecting the *j* for generating the *m* cost vector *C*;
- For each network interface *i* of host A (*i* in *[1..m]*), IF *i* is unavailable or *RE(i, j)* = false, *c_{ij}* = infinite, continue *c* with the next interface of host A; else set *c_{ij}* = MAX; add *i* into qualified set;
- For each interface *i* in qualified set, do *e* to *h*;
- If MMP is a *use policy* and *i* is the specified one, *c_{ij}* = 0, exit;
- If MMP is a *priorit policy* and *i* has a defined priority value, set *c_{ij}* as *i*'s priority value, continue *d* with next interface;
- If MMP is *weight policy*, calculate *i*'s cost value and assign to *c_{ij}*; continue *d* with next interface;
- If MMP is *default policy* and *i* is the specified one, *c_{ij}* = 0, exit.

In the algorithm above, if MMP is a *weight policy* then cost is the sum of the products of weights and the corresponding values. In particular logarithmic normalization has to be made as in [6].

5.3 Decision Making

After the two sides of the channel have both generated their cost matrixes, the final decision is to be made about which connection is the best and will be used for the channel at the moment. There are two modes for making the decision, namely *Master-Slave* (M-S) and *Peer-to-Peer* (P2P). First the *channel end type* of both the two sides is to be ascertained. The channel end type is a setting specified by user or application, as *master* or *slave*. Then the decision mode can be determined by the XOR (i.e. exclusive or) operation with the channel end types of the two sides. That is, if

the two types are the same, then a P2P mode is chosen. Otherwise a M-S mode is determined. In any mode the ultimate goal is to minimize the overall cost.

Both the channel's two sides are *Policy Decision Points* (PDPs). The decision-making algorithm is a distributed process that is based on the cooperation of the two PDPs, to enable suggestion and negotiation. The algorithm takes the two generated matrixes as the input, and makes the decision of currently the best end-to-end connection for the channel as the output. The detailed algorithm for the decision-making process in M-S mode is as follows. We assume that C_M is the $m \times n$ cost matrix generated at Master PDP (M-PDP) and C_S is the $n \times m$ cost matrix generated at Slave PDP (S-PDP).

- At the M-PDP add all pair (x, y) into candidate set if c_{xy} has the minimum value in C_M ; If the pair is unique, return (x, y) ; else do b.
- At the S-PDP by using the exchanged pair (y, x) of each pair (x, y) in the candidate set as the index, search the corresponding element c_{yx} from C_S ;
- At the S-PDP return (j, i) with the index of which the c_{ji} in C_S has the minimum value;
- At the M-PDP return the exchanged pair (i, j) ;

The detailed algorithm for the decision-making process in P2P mode is as follows. The algorithm describes the process at the PDP that generates an $m \times n$ cost matrix C_1 . The process at the peer PDP that generates an $n \times m$ cost matrix C_2 is the same.

- Get the $n \times m$ cost matrix C_2 from the peer PDP;
- Add C_1 with the transposed matrix of C_2 to sum matrix S , i.e. $S = C_1 + C_2^T$;
- Return (i, j) with the index of which the corresponding cost sum s_{ij} in matrix S is the minimum;
- At the peer PDP the exchanged pair (j, i) is returned.

In any of the two cases above the returned pair (i, j) denotes that the best connection is from interface i at local host to interface j at remote host (S-PDP or peer PDP).

6. CASE STUDY

In this section a case study is presented as the example of how to use the proposed policy mechanism and to show the effectiveness of the proposed policy mechanism. In particular since the focus is on the policy representation and evaluation algorithm we emphasize the decision-making process for various situations. Suppose a concerned channel CH between host A and host B is used by a network application. Related channel information is:

CH.direction = inout;

CH.traffic_class = "normal service";

The number of the network interfaces of host A is 3 and 2 for host B. Tables 1 and 2 list the related local and end-to-end information of the interfaces.

There are five cases in total to be studied in this section, in which the policy and setting at the host A vary while those of the host B persist. Table 3 describes the

Table 1. Local information of interfaces

Interface	Type	Cost	Speed
A.1	X	1c/min	50 Mbps
A.2	Y	10c/min	10 Mbps
A.3	Z	100c/min	1 Mbps
B.1	S	2c/min	2 Mbps
B.2	T	20c/min	10 Mbps

Table 2. End-to-end information of interfaces

E2E	RTT	Bandwidth
(A.1, B.1)	150 ms	70 KBps
(A.1, B.2)	250 ms	150 KBps
(A.2, B.1)	450 ms	120 KBps
(A.2, B.2)	40 ms	330 KBps
(A.3, B.1)	550 ms	30 KBps
(A.3, B.2)	20 ms	80 KBps

Table 3. Policy and setting of host A

Case	Channel end type and policy
1	Slave, device-default, (-, -, -, (default, type, Y))
2	Master, application-priority, (-, -, (cost<2c/KB), (A.1, 1)(A.2, 4)(A.3, 10))
3	Slave, application-weight, (-, -, (RTT<500ms AND speed>300Kbps), (BW, 1.0))
4	Master, channel-weight, (-, -, (RTT<300ms), (cost, 0.4)(RTT, 0.2)(BW, 0.4))
5	Slave, channel-weight, (-, -, (RTT<300ms), (cost, 0.4)(RTT, 0.2)(BW, 0.4))

policy and setting of channel end type applied at host A for each case. In Cases 4 and 5 the policies are the same while the setting on channel end types are different.

Policy at host B is an application level weight policy and the same as of host A in case 3, as follows.

(-, -, (RTT<500ms AND speed>300Kbps), (BW, 1.0))

That is, the application policy is used to any direction of any traffic class, with requirements on RTT and speed, and concerns only bandwidth as factor. The channel end type of host B is set to "client" all the time.

Table 4 lists the results of the four cases, in which C_A and C_B are the cost matrixes generated at host A and B respectively. It is easy to find the effectiveness of the policy mechanism. Note that bandwidth has to be used as its reciprocal value. In Case 2 the RE concerns cost per kilobytes (c/KB). Since, according to Table 1, cost is relative to time (c/min), so bandwidth values in Table 2

Table 4. Results of the four cases

	Case 1	Case 2	Case 3	Case 4	Case 5
C_A	MAX, MAX 0, 0 MAX, MAX	ln1, ln1 ln4, ln4 ln10, ln10	ln(1/70), ln(1/150) ln(1/120), ln(1/330) infinite, ln(1/80)	ln0.50, ln0.41 ln1.26, ln0.52 ln5.72, ln1.99	
Mode	P2P	M-S	P2P	M-S	P2P
Return	(A.2, B.2)	(A.1, B.2)	(A.2, B.2)	(A.1, B.2)	(A.2, B.2)
C_B			ln(1/70), ln(1/120), Infinite ln(1/150), ln(1/330), ln(1/80)		

have to be used to get the cost measurement. While in Case 4 the weighted cost is only used for comparison, so the values in Table 1 can be directly used.

7. CONCLUSIONS

A policy mechanism is proposed for the adaptive decision of connection selection in channel establishment and vertical handoff between heterogeneous access networks. A policy is represented as a 4-tuple, including the direction and class of traffic, requirement expression, and concrete evaluation items. Three steps are involved in the evaluation process, namely policy traverse, cost matrix calculation, and decision-making. A case study shows the usability of the proposed mechanism. The policy mechanism can be easily extended to include adaptive selection of multiple user devices, other than multiple connections. More context information is to be taken into consideration, including e.g. device capability, velocity, geographic location, calendar, etc. Moreover, PCIM specification [18] will be studied to investigate the possibility to define the proposed policy model accordingly.

8. ACKNOWLEDGMENTS

This research was carried out in the CAPNET program. Financial support by Hantro, IBM, Nokia, Serv-It, TEKES, and TeliaSonera Finland is gratefully acknowledged.

9. REFERENCES

- [1] Alam M., Prasad R., and J.R. Farserotu, Quality of service among IP-based heterogeneous networks. *IEEE Personal Communications*, 8(6), Dec. 2001, 18–24.
- [2] Hyosoon Park, et. al, Vertical handoff procedure and algorithm between IEEE802.11 WLAN and CDMA cellular network. In *Proceedings 7th CDMA International Conference*, Seoul, Korea, 2002.11, 217-221.
- [3] Mola G., Interactions of vertical handoffs with 802.11b wireless LANs: handoff policy, Masters theses, Department of Microelectronics and Information Technology, Royal Institute of Technology (KTH), Stockholm, Sweden, March 2004.
- [4] Aust S., Proetel D., Fikouras N.A., Paupu C., and Gorg C., Policy based Mobile IP handoff decision (POLIMAND) using generic link layer information. In *Proceedings 5th IEEE Int. Conf. Mobile and Wireless Communication Networks*, Oct. 2003.
- [5] Pablo Vidales, et. al, PROTON: A Policy-based Solution for Future 4G devices. In *Proceedings 5th IEEE International Workshop on Policies for Distributed Systems and Networks*, Yorktown Heights, New York, June, 2004, 219-222.
- [6] Wang H.J., Katz R.H., and Giese J., Policy-enabled handoffs across heterogeneous wireless networks. In *Proceedings 2nd IEEE Workshop on Mobile Computing Systems and Applications*, New Orleans, Louisiana (1999) 51–60.
- [7] Balasubramaniam S. and Indulska J., Vertical handover supporting pervasive computing in future wireless networks. *Elsevier Computer Communications*, 27 (2004) 708-719.
- [8] Zhao X., Castelluccia C., and Baker M., Flexible network support for mobile hosts. *ACM/Balzer Mobile Networks and Applications (MONET)*, 6(2) (2001) 137–149.
- [9] Chan P.M.L., Sheriff R.E., and Y.F. Hu, Mobility management incorporating fuzzy logic for a heterogeneous IP environment. *IEEE Communications Magazine*, December 2001: 42- 51.
- [10] Zhang Q., C. Guo, Guo Z., Zhu W., Efficient mobility management for vertical handoff between WWAN and WLAN. *IEEE Communications*, 41 (11) (2003) 102–108.
- [11] Stemm M. and Katz R.H., Vertical handoffs in wireless overlay networks. *ACM Mobile Networks and Applications (MONET)*, 3 (4) (1998) 335–350.
- [12] Chen L.-J., Sun T., Chen B., Rajendran V., and Gerla M., A smart decision model for vertical handoff. In *Proceedings 4th ANWIRE International Workshop on Wireless Internet and Reconfigurability (ANWIRE 2004)*, Athens, Greece, 2004.
- [13] Sun J., Tenhunen J., and Sauvola J. , CME: a middleware architecture for network-aware adaptive applications. In *Proceedings 14th IEEE PIMRC2003*, Beijing, China, 2003, 1: 839-843.
- [14] Almquist P., Type of service in the Internet Protocol suite, IETF RFC 1349, July 1992.
- [15] 3GPP, Quality of Service (QoS) concept and architecture, 3GPP TS 23.107v6.1.0, April, 2004.
- [16] Chalmers D. and Sloman M., A survey of quality of service in mobile computing environments. *IEEE Communications Surveys*, 2(2), 1999: 2–10.
- [17] Ghinea G., Thomas J.P. and Fish R.S., Mapping Quality of Perception to Quality of Service: the Case for a Dynamically Reconfigurable Communication System, *Journal of Intelligent Systems*, 10(5/6), pp. 607-632, 2000.
- [18] Moore B., Ellesson E., Strassner J., and Westerinen A., Policy core information model – version 1 specification. IETF RFC 3060, Feb. 2001.

Usage Patterns of FriendZone – Mobile Location-Based Community Services

Asaf Burak

Carnegie Mellon ETC

700 Technology Drive

Pittsburgh, PA 15219 USA

aburak@andrew.cmu.edu

Taly Sharon

MIT Media Laboratory

20 Ames St

Cambridge, MA 02139 USA

taly@media.mit.edu

ABSTRACT

How do users accept, and use, for a long period of time, location based services (LBS) on their mobile handsets? FriendZone, a suite of mobile Location-based Community Services has been launched. The services included Instant Messaging and Locator (IM&L), Location-based Chat, and Anonymous Instant Messaging (AIM), with supporting Privacy Management.

A 21 month usage survey of more than 47,000 users, most of them young adults, followed by user interviews, is reported herein. The results indicate that AIM is the most popular and used service, more than IM&L, with lower use of Chat. The interviews showed that young adults are interested in immediate stimulations and therefore use AIM, which could lead them to face-to-face meetings. In addition, IM&L is limited to one carrier and hence is less attractive. Lastly, young adults using this service are more interested in sharing their location than in their privacy.

Keywords

Location-based Services, LBS, Mobile Communities, Mixed Reality, 3G, WAP, SMS, Ubiquitous Computing

INTRODUCTION

Virtual Communities connect people with common interests by forming virtual worlds on the Internet. These worlds include varied community services, such as Forums, Chat, Dating, Instant Messaging (IM), etc. [20]. The interaction in these worlds is mostly composed of symbolic and anonymous communication. Hence, designing these virtual environments is a non-trivial task [3]. For examples see Babble [2], Chat Circles [5] and RVM [10].

The success of these communities has drawn mobile content developers and users to try and port this type of application into Mobile environments. Wireless handsets distribution has expanded to reach over a 1.5 billion subscribers [6]. The emergence of new technologies, such as Short Message Service (SMS) and Wireless Application Protocol (WAP), has turned mobile phones into enhanced data terminals. SMS, the leading service, mostly among young adults, has reached billions of messages each month [9]. In spite of obvious technological limitations, Mobile Communities have the promise of “access, anytime, anywhere” [17].

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MUM 2004, October 27-29, 2004 College Park, Maryland, USA.
Copyright 2004 ACM 1-58113- 981-0 /04/10... \$5.00

An example of such a mobile community is Freever, a European commercial product, which implemented mobile Information access, Forums, and Chat [7]. Two implementations of IM and Chat that include presence awareness and have been extended to mobile platforms are Hubbub [12] and ConNexus-Awarenex [21].

In mobile devices, location is a key factor. Thus, a new concept of Location-Based Services (LBS) has emerged [23]. LBS were identified as the “Killer Application” for wireless Internet [23]. Most of the developed LBS offered applications such as: personalized information and mobile e-commerce [13, 19], but hardly any social applications.

Furthermore, the integration of LBS and mobile communities has lead to a new branch of applications - Mobile Location-based Communities. Some social applications (rather than a complete suite) have been implemented in this branch. Examples of such applications in the market of mobile services are AT&T “find friends” [15] and InirU [11] that alerted users when their friends or optional anonymous matches entered their physical zone.

Because of the option to locate a mobile user, social LBS raise further privacy concerns. However, extensive privacy management tools have already been integrated in the regular presence awareness applications [10, 12]. When implementing privacy tools, the risk of disturbing the normal use of the application arises. An example for that is RVM - its initial privacy model was too strict and disrupted the introduction of the system to the users [10].

Yet, a major question remains open: considering the low-level user interface in most mobile devices, combined with the requirement to pay the operator for the service and the need for privacy, how will mobile users accept a whole suite of Location-based Community Services?

In this paper we focus on a large-scale, long-term study of FriendZone, a suite of mobile location-based community services [4, 8]. FriendZone consists mainly of Instant Messenger and Locator (IM&L), Anonymous IM (AIM), and Location-based Chat, with Privacy Management tools.

The contribution of this paper is in studying the long-term usage pattern of Location-based Community Services on a large scale and without extensive user guidance. This paper explores the relative usage of the different services offered, pattern changes over time, and the specific interaction style of users within each application. The paper also addresses the issue of how mobile users consider the importance of privacy regarding their location and availability. An additional contribution of the paper is in providing a design concept for mobile community services on various platforms, mostly those with low-level graphics devices.

This paper is structured as follows. The next section describes FriendZone’s services and user interface, with emphasis on location information. The following two sections report the detailed results of both the usage survey and the users’ interviews. In the section before last, we analyze the results and present our conclusions. Lastly, we discuss some future directions.

FRIENDZONE

FriendZone, developed by AxisMobile (formerly Valis) [1], consists of the following Location-based Community Services: IM&L, AIM, and Location-based Chat. It was developed considering the following design principles:

1. Multi-platform - FriendZone should be accessible via multiple platforms, such as different mobile phones, PDAs, PCs, etc., while keeping similar functionality.
2. Integration - The different services should be integrated. Thus, they should share the same "look and feel" and enable the manipulation of data objects between them. For example, a user will be able to invite a buddy from the IM&L application to join a Chat session.
3. Location and Accuracy - Location-based information should be integrated into all the services. Its accuracy is a few hundred meters in urban zones, and several kilometers otherwise.
4. Privacy - Privacy should be inherent in the system to prevent abuse of location information. Users will be provided with an extensive Privacy Management tool.

These design principles were put into effect in FriendZone's applications described below.

1. Instant Messenger & Locator (IM&L)

Mobile communication adds an element of location uncertainty. Perhaps the most common question that mobile users ask each other is "where are you?" [22]. Yet, new technologies enable an operator to locate its network subscribers, whenever their handset is turned on [13].

To answer that need, a service to locate friends and acquaintances is offered. The "L" (Locator) has been added to the IM to form IM&L. The option to view enhanced presence (online/offline, and more) and (real) location of other users stands at the core of FriendZone's services.

The service design is based on popular Internet Instant Messaging (IM). IM&L adds to that a new layer of location information. Users are able to manage buddy lists by adding friends, based on their approval, using phone numbers as identifiers. They can then view their buddies' enhanced virtual presence, send them textual messages which they receive instantly, and view their location.

On non-graphical mobile interfaces, the virtual presence is shown by a set of ASCII emotion icons – emoticons [18]. A double smiley :)) indicates that buddy is excited and thus eager to communicate. A smiley :) stands for happy and available. A sad face :(stands for "do not disturb". In figure 1(a), which presents a typical buddy list, shelly is eager to talk; guy is online; while danny wants to be left alone.

Location information in IM&L may be displayed in two possible resolutions:

1. Relative distance to the users.
2. Absolute cell-ID based location.

The relative distance is attached to the buddy list. In addition to the standard buddy list information, such as nickname and virtual presence, there is a vicinity indicator. The vicinity is depicted in four levels: very near (0-0.6km), moderate distance (0.6-1.3km), far (1.3-2km), and out of zone (more than 2km).

The vicinity indicator is presented using a set of ASCII asterisks. Very near is three asterisks (***) ; moderate distance - two (**); relatively far - one (*); and out of zone - no icons. Figure 1(a) presents an example UI in which danny is very near, guy is within

moderate distance, shelly is relatively far, and adam is out of zone. On graphics-enabled devices, a graphic map is available (see figure 3).

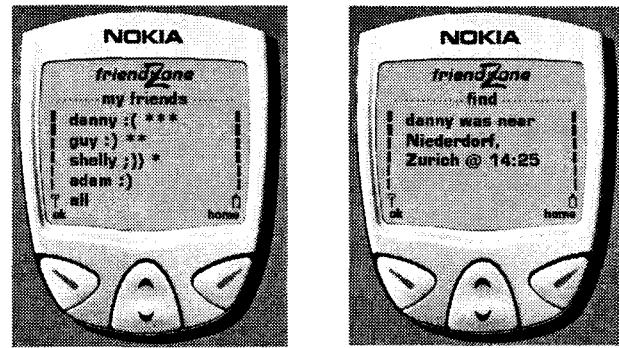


Figure 1: Instant Messenger and Locator interface

In the second resolution, a more accurate location of a specific buddy is given upon an explicit request. The user selects a buddy and clicks "find". The buddy's location is then presented by a textual description of the cell's area, with the time of relevancy (see figure 1(b)).

Hence, IM&L lays the foundation for a location-based mobile community by creating small social circles. Community members can physically map their closest group of friends at all times.

2. Mobile Chat

Similar to Internet Chat, the mobile version allows users to exchange textual messages in virtual chat rooms. Anonymity is preserved and therefore cell-ID based location is hidden. Nevertheless, the relative distance of chat roommates is available.

In contrast to the Internet topic-based chat rooms, a location-based chat room, called a "Local Chat Zone", is also available for mobile users. The Local Chat Zone hosts users who are located in the same area. The room is named after the nearest cell-ID description, e.g., "London Soho Zone". The chatters in that zone might not share common interests, but they share the same location at present time.

By adding location, the Chat (and also the rest of the services in FriendZone) forms a type of a mixed reality environment by contrasting the real and the virtual reality (the latest often includes false identities and information).

3. Anonymous Instant Messenger (AIM)

The AIM service enables an automatic match of two anonymous users, based on their interests. In contrast to IM&L, AIM users are not friends in reality and thus don't have each other's phone numbers. Consequently, they cannot be added to IM&L, nor can their absolute location be revealed, that is, until they exchange phone numbers.

Relative distance plays a role here too. Users have the option to define location as a leading criterion for the matching process. When doing so, only matches within their zone would be presented to them.

Users start by defining their personal profile (see Figure 2(a-c)). The profile includes demographic and personal information (presented in 2(a-b)); and a personal (virtual) picture (presented in 2(c)). This feature is limited to WAP devices that support images. The virtual picture is chosen from a picture gallery that is designed to represent socio-psychological types.

In order to find matches, users define a preferred matching profile. Consequently, they get a list of matching results (see figure 2(d)). The list offers somewhat different information than the buddy list. Instead of the virtual presence, a concise presentation of gender (b-boy, g-girl) and age of the match is given. In addition, an exclamation mark ("!") indicates a Perfect Match (the matched profile was exactly like the preferred matching profile).

For example, see Figure 2(d). The first line is "jo! ***b18", that means that jo is an eighteen (18) years old boy (b), who is very near (***!) and he is also a Perfect Match (!). Similarly, mo is a boy, which is also very near, whose age was not specified; cloe is a 21 years old girl (g) who is relatively far (*); momo has not defined gender, nor age, and is also relatively far (*), etc.

When receiving a list of matches, users can choose to overview the results' complete personal profiles, contact them, or find degrees of separation from them. Finding the degrees of separation is based on the "Six Degrees of Separation" theory, which claims that the distance between any two individuals in terms of direct personal relationships is relatively small [16]. By using it, people can check their levels of social connection with an anonymous match. The algorithm checks the database of buddy lists to find a connection between the two persons. A result between 1-6 degrees is presented. For example, "3 degrees" means that there are three connections via two people in the social network that leads from the user to the anonymous match. For example, here is an example case of three degrees: User <knows> John <knows> Alicia <knows> match.

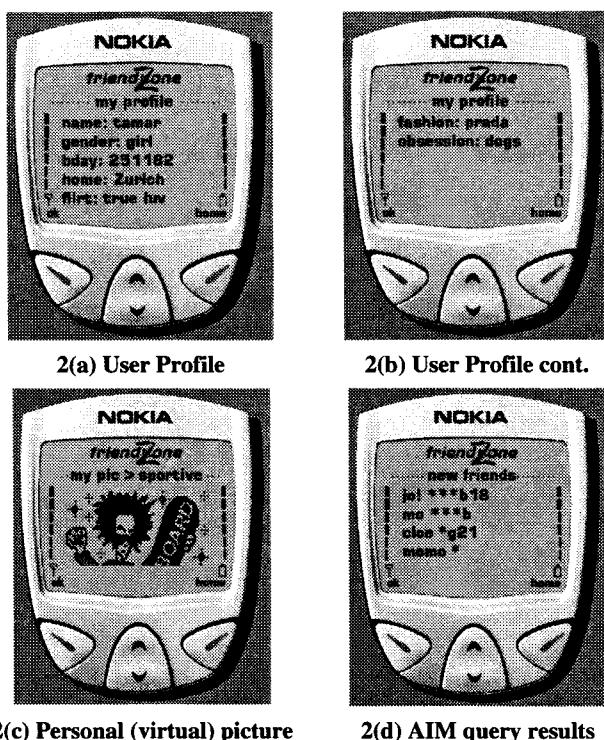


Figure 2: AIM user interface

The information about the degrees of separation adds a teasing clue regarding the real identity of a match, while preserving the basic principle of anonymity.

4. Privacy Management tools

Considering our belief in the importance of user's privacy, the following extensive privacy tools have been integrated in FriendZone:

1. When registering to FriendZone, the users are made aware and required to confirm (Disclaimer) that they understand that, subject to their consent, their location will be made available to other users.
2. Adding users to a buddy list is not automatic; the users should consent to it.
3. A tool, "The Block List", provides the users with a full control over their presence and personal location information at any time. Using the Block List, users can easily see who is capable of receiving their availability and location, and are able to block one, some, or all buddies on the list from seeing them. Blocking means that the buddy will always see the user as unavailable. It can be temporal or permanent. Furthermore, the Block List enables retracting a previously given consent to be added to a contact list, by deleting oneself from it.
4. Inactive users are alerted after a period of a month that their location information will be blocked to protect their privacy. If they do not show any activity for the next three days, the automatic blocking is performed.

5.

5. Other Services

Other services in FriendZone are those not included in the main applications, such as the registration process, customization options, find self-location ("where am I?"), language substitution, online help, etc.

Access platforms and user interface

FriendZone supports different platforms, networks and protocols. Thus, FriendZone provides each platform with a UI that supports its specific limitations and opportunities:

1. SMS only handsets - textual presentation, operated by short menu commands.
2. WAP enabled devices – textual, with some graphics (see figures 1 and 2), operated by textual menus, and based on relatively short online sessions.
3. Third Generation (3G) devices – richer color graphics, operated by menus or pointing device.
4. Internet (PC) – rich color graphics.

3G devices and PC UI enable enhanced presentation. On these platforms, the buddy list is presented in a radar-like graphical map, with distances and compass directions presented around the user (see figure 3). Buddies in the user's zone are plotted inside the circle and those out of zone are plotted outside the circle, with distances attached.

The Internet platform of FriendZone is also interesting. The Web site offers users an option to join the mobile community with all the location-based features on a stationary basis. Users log into the site using their mobile phone number, and their location information is presented to their friends according to their mobile handset location.

USAGE SURVEY

This current evaluation was meant to learn how mobile services are accepted and utilized by a large scale of users on a commercial basis. As opposed to small-scale trials, in this study, users used the system naturally and were charged for each command they activated. Other significant differences are that the users were not guided extensively, and that the community involvement was not controlled or limited, as usually done in a structured trial.

The statistical data was collected over a period of 21 months, between May 2001 and January 2003. The beginning of the survey is marked by the commercial launch of the service in Switzerland. At the end of that period, it had more than 47,000 paying subscribers. The data collected included logs of all the commands issued by all users over the entire period of time, which summed up to millions of commands.

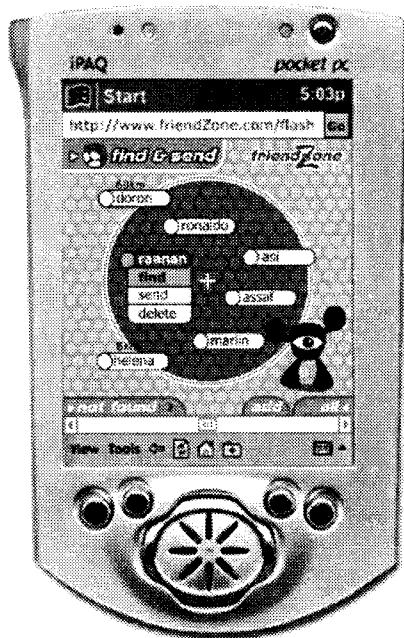


Figure 3: 3G device user interface

Most of the users accessed the service through mobile low-graphics handsets. A relatively small number of users accessed it occasionally through the Web interface. All the users paid for the services through their mobile phone bills (airtime and/or transactions).

IM&L and AIM were included in the very first version launched. Chat was added on June 2002, with data collected about its use for nine months only.

Our expectations were that IM&L would be the most used application of FriendZone. We believed that the ability to reveal the accurate location of other friends and people would be the main attraction for the location-based community. Presenting such a new concept to the public, we also predicted high privacy concerns and awareness among users regarding their own location information.

As for the Internet site, we were curious to discover how the usage will be as the users are billed for their Web actions via their cellular phone bill, a concept new to the "all free" orientation of most services on the Internet.

Services relative usage

The relative usage of the services (IM&L, AIM (including Privacy Management), Chat, and Others (see above)), is measured by the number of monthly commands performed in each service, on all platforms. Figure 4 presents the data on the last month of the survey. Relative use, instead of absolute counts, eliminates the effects of external changes, such as the increase in number of users.

The information reveals a clear dominance of the AIM application (62%). AIM was used twice as much as IM&L (30%). Chat is last with low usage (5%).

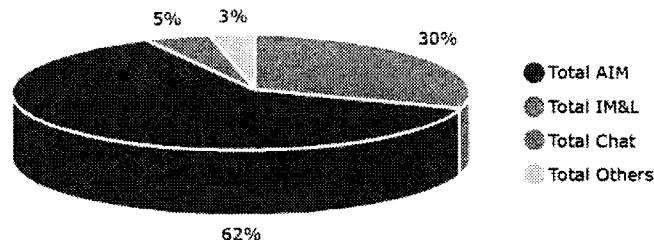


Figure 4: Services usage (February 4th 2003)

Specific commands usage

Table 1 presents the top used commands¹ in FriendZone and their relative usage percentage. The table is structured as follows: the first column is a serial number for each command (that will be referred to in the text in parenthesis), the application it belongs to (column S) using a one character code (A-AIM, C-Chat, M-IM&L, O-Others), the command description, and the right column is its relative usage percentage.

#	S	Command	%
1	A	Check a specific match (profile)	19.5
2	A	Get possible matches (5 at a time)	18.9
3	A	Contact an anonymous match	10.1
4	M	Find all friends on my buddy list	6.6
5	M	Find a friend	5.9
6	M	Join the buddy list of another user	5.2
7	A	Check my AIM preferences	4.2
8	M	Get my buddy list	3.9
9	A	Get degrees of separation	3.8
10	A	Get contacted matches list	2.5
11	C	Refresh Chat messages	2.2
12	A	Change my AIM preferences	2.0
13	A	Set my profile	1.7
14	C	Get Chat topics list	1.7
15	O	Find my location	1.3
16	--	Other commands	10.5

Legend: S-Service, A-AIM, C-Chat, M-IM&L, O-Others

Table 1: Commands distribution

The table shows that the AIM commands are clearly leading, with the command 'Check a specific match' at the top of the list. Three of the AIM commands, including 'Get possible matches', 'Check a specific match' and 'Contact an anonymous match', cover

¹ The commands appear as in their description and not by their "interface-name" in the application.

almost 50% of the overall FriendZone usage. However, only 10.1% out of it is actually for direct contact with the potential matches by composing and sending a textual message. It seems that AIM users spend most of their time on "window-shopping", checking and aiming at the right target.

Even more interestingly, AIM users are much more interested in getting new matches (command 2, with 18.9%) than checking again on their previous contacted matches (command 10, with 2.5%). However, we should consider that some AIM "couples" simply moved to other channels of communications by publishing their phone numbers. Thus, they might have continued their relations in voice conversations, generic SMS (direct one-on-one, without the mediation of FriendZone), by adding each other to IM&L's buddy list, or even meeting face to face.

IM&L correlation to the overall system usage

We found a strong correlation between the number of users in a user's buddy list to the total number of commands performed in the system. The correlation is significant ($r=.64$).

We divided the users into three groups (group sizes in parenthesis):

1. Users with 0 or 1 entries in their list (74%).
2. Users with 2 to 5 entries in their list (17.6%).
3. Users with more than 6 entries in their list (8.4%).

Figure 5 demonstrates the strong correlation between the number of buddies in the list and the total number of commands the user performed in the system (95% of the group is between the lines). The data also indicates that, unfortunately, the majority of users have a relatively low number of buddies in their list.

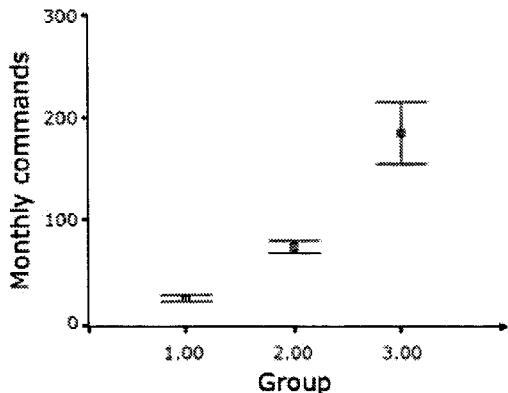


Figure 5: Internal Correlation

Pattern changes over time

The over-time distribution of the services usage is presented in figure 6. Note that each line does not represent an absolute number but a relative amount of use. The results show that in the early days of FriendZone, IM&L was the most dominant service with over 60% of the use per month. Since then, AIM use was rapidly increasing, crossing the height of more than 60%.

When we reviewed the detailed data, we found that the "Block list" (Privacy Management) use has decreased over time. Being once one of the 10 top commands (3%), it decreased to less than 1%.

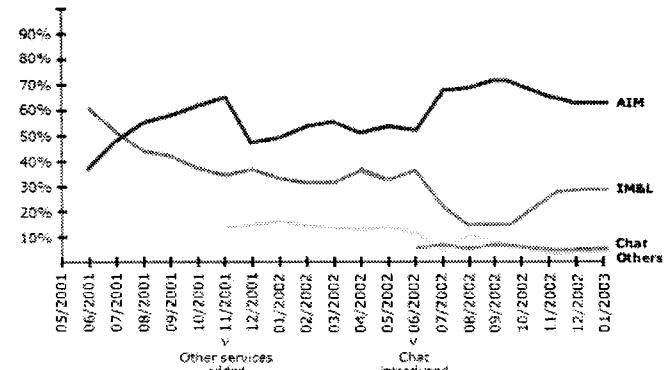


Figure 6: Usage percentage over time

We should note that when studying FriendZone usage over time, we have to take into account the following factors:

1. The community had evolved constantly, with new users joining it on a daily basis.
2. Minor improvements and changes were made in the software and user interface.
3. Chat was added as a new service only on June 2002.

Nevertheless, considering the size of the population and the length of the survey, and the clear trends over time that were found, these effects become marginal.

Web interface

FriendZone Web site usage itself increased over time, but is still relatively low compared to mobile usage. However, the Web site's usage statistics reveals the same dominance of AIM as the most used service. The customer team reported that users testified that it is much easier to manage personal profiles and preferences using the Web than on the limited mobile handset. It basically means that the matching process is much more efficient and faster that way. It is also interesting to note that even though the users were paying for FriendZone, unlike many other Internet IM, Chat, and Dating services, they still found a reason to use it.

USER SURVEY – INTERVIEWS

In addition to the survey of an actual field setting, live interviews with FriendZone users were conducted to confirm that the command logs represent the actual system usage, as well as to try and explain the findings. The interviews were carried out during January–February 2003 and included questions about the usage habits of the system and reasons for preferring one service to others.

29 random users were approached anonymously through the system and interviewed; 18 males and 11 females, age 12–31 ($M=19$, $SD=3.5$). This is consistent with the overall demographics of Friendzone. All users, except one, were using the applications at least once a week, while 12 of them used it on a daily basis.

Popularity of services

The users were asked to order the three core services according to their preference. The results are presented in Table 2. The first column is the order of the service, and the other columns show how many users rated each application as first, second or third.

According to the usage logs, the leading service was AIM, with 13, 10 and 6 participants rating it as the first, second, and third most attractive service, respectively. Surprisingly enough, Chat

was rated second with 12, 9 and 8 participants, higher than IM&L, and relatively close to AIM. IM&L was rated last.

Order	AIM	Chat	IM&L
First	13	12	4
Second	10	9	10
Third	6	8	15

Table 2: Interviews – favorite Services

Reasons behind the usage of services

In order to check why the users prefer certain services, the users were provided with a set of positive and negative claims regarding each service. The claims were based on previous feedback from the support team, as well as from our assumptions about possible causes. The users were asked to rate each claim on a Likert scale of 1-5, where 1 is “disagree with the claim” and 5 is “agree”.

Table 3 presents the main claims given to the users and the average agreement of the users with it. The results show that the users like AIM because it is anonymous (rating 3.69); it provides high stimulus and action (3.45) and leads to face-to-face encounters (3.07). Even more interestingly, 15 of the 29 users interviewed (52%) reported that the AIM interaction actually led to reality meetings with anonymous matches.

#	Claim	Rating
1	Like AIM because it is anonymous	3.69
2	Like AIM because it provides “action”	3.45
3	Like AIM because it leads to face-to-face encounters	3.07
4	Use IM&L less, because it is difficult	1.93
5	Use IM&L less, because it is limited to one mobile network	3.24
6	Use Chat less, because it is difficult to use	2.31
7	Prefer Internet chat over Mobile Chat	3.24

Table 3: Average rating of claims

When checking the relatively low usage and rating of IM&L and Chat, the main question is if the services’ UI is not a cause. In addition, the fact that IM&L is limited to one mobile network might affect this too. This limitation means that users cannot add friends who operate on a different mobile network to the buddy list. Considering the correlation found between the number of listed friends (buddies) and the overall usage, this could have contributed to low use of IM&L.

The results show that the users mostly disagreed with the claim that IM&L and Chat are used less because they are difficult (rating 1.93 and 2.31 respectively). However, the users did agree that they use IM&L less because it is limited to one mobile network (rating 3.24). In addition, users mostly agreed that they prefer Internet Chat over Mobile Chat (rating 3.24).

Preferences/likes of users

In addition to the above claims, the users were given open-ended questions regarding what they like or dislike in FriendZone. When asked what they liked in the system (note it can be more than one thing), 10 users mentioned location (itself or a feature of it), 8 mentioned AIM, 7 mentioned Chat, and 4 mentioned access everywhere/ anywhere/anytime. The results are shown in Table 4 below.

One user described AIM as “exciting”; another liked the fact that he can add anonymous matches to his buddy list, once he is more acquainted with them. Other said: “I enjoy meeting new people, while keeping my close friends”.

Likes	Location/IM&L	AIM	Chat	Anywhere /Anytime
# of users	10	8	7	4

Table 4: Preferences/likes

As for disliking, 6 (21%) users have mentioned Chat. They complained mostly about the inherent mobile medium limitations: slow connection, small number of rooms and chatters, and that the length of messages is limited.

Privacy

The statistics showed that privacy tools were hardly used. In the user survey, users were asked if they use Privacy Tools, and to explain their reasons. 7 (24%) reported using it, 7 (24%) a little, and 15 (52%) not using it at all. The results are summed up in Table 5 below.

	Not using	Little	Using	Total
# of users	15	7	7	29
%	52%	24%	24%	100%

Table 5: Use of Privacy Tools

As for the reasons, of the 15, two had no buddies in their lists and therefore did not need it, one did not know about it (and it obviously did not matter to him), and the rest just said they did not need or want it. The majority simply did not see a reason to block their location information, saying, “I don’t feel any need to hide myself”, “I am not ashamed in my whereabouts” or “Hiding? On the contrary – find me, communicate with me; meet me!”

RESULT AND CONCLUSION

Our goal in this section is to obtain a deeper understanding of the results gleaned out of the services’ utilization and the users’ direct feedbacks.

AIM Is a Killer Application

The results regarding the services popularity (i.e., AIM) are highly coherent. Based on the user interviews, we suggest the following explanations:

1. Anonymity

FriendZone AIM suggests a unique extension to popular SMS. It allows users to send anonymous messages to other users, without knowing their phone numbers. The anonymous environment is appealing to users - it invites them to meet unknown people, design their virtual identity and publish it and in some cases, find a “perfect match” for their interests.

2. AIM promises “action” and leads to real meetings

Mobile data sessions are much shorter than Internet sessions. The content exchanged between mobile users in the form of short textual messages is limited, and costs money. While Internet users can be involved in hours of casual surfing, most FriendZone users log on the system for a reason. AIM might be a natural solution: a promise for high action and excitement in a relatively short time. Furthermore, the mobility of AIM, coupled with the availability of location information, seems to lead faster to face-to-face encounters and thus is more tempting to use.

IM&L requires encouragement & guiding

The relatively low usage and popularity of IM&L stand in contrast to previous expectations. A significant number of users have had a surprisingly low number of friends in their buddy lists, and consequently, did not use IM&L often. The high correlation between the low number of buddies in IM&L list and the low usage of FriendZone in general suggests that users need more encouragement and guidance for using this service.

Based on the user survey, difficulty of use is not the main drawback. However, other possible explanations came up:

1. No IM&L with other Operators

The Swiss version of FriendZone is offered only to Swisscom subscribers. As a result, Swisscom users are not able to add some of their close friends to their lists, if unfortunately those are subscribed to a different operator.

2. IM&L demands extra effort

IM&L requires an initial effort to build the buddy list, which users might not have taken the time to do, since quickly drawn by the AIM attraction of immediate results.

Chat needs more evaluation & improvement

Chat, an Internet favorite, shows low mobile usage compared to AIM and IM&L. Based on users' feedbacks, it seems that current mobile handsets are not a natural to chat applications. Conducting a chat with phone keys is tough and is relatively slow [14], not to mention the limited display. Checking the Chat logs reveals that chat rooms usually host a very small number of users (5-7 average).

However, the interest our users have expressed in Chat, despite its infancy problems, combined with its late introduction, leads us to consider further evaluation before we "bury" this interesting service. In addition, extra effort should be invested in finding more innovative design solutions for improving the mobile chat operation.

High acceptance of location features

The survey suggests that location capabilities have been truly well accepted. However, they are clearly not amongst the most popular features. It seems that the location properties are helpful and interesting so as to promote use of other services that materialize the interaction.

Low privacy concerns

Privacy was a corner stone in FriendZone design and a condition for its legal distribution. The Privacy Management tools were all approved by the Swiss Justice Department. However, it seems that privacy is certainly not the main concern of the users themselves.

This is consistent with the observation that Privacy Management is critical for giving people the peace of mind that they *can* control access, though they rarely do [12].

However, we should point out the low age of FriendZone users. It seems there is some connection between their young age and their low awareness for their privacy. We suspect that when exploring a larger distribution of ages, the results might be different.

DISCUSSION AND FUTURE DIRECTIONS

The main goal of our extensive study was to determine the acceptance and use of Location-Based Community Services on mobile devices. Whilst Internet virtual communities are common, we feel that implementing these services on mobile handsets is

more than just porting them – it calls for different design concepts: a focused and simplified UI, with innovative features to exploit the unique qualities of mobility. These services should not be a mere extension of Internet services but a complementary application. Hence, the future of virtual communities would offer full access for their users - anytime, anywhere.

In addition to the current update of the FriendZone's applications as a result of the study, future directions should include further studies. An interesting direction is to focus on the connection between the demography of users and their usage patterns. Such surveys should take into consideration gender, age and education, and their effects on general use of the various services. Another interesting aspect is analyzing the usage patterns of users, according to their acquaintance with the system - regular users vs. casual ones. This can also lead to further improvements in the user interface. In addition, as the services are introduced in other countries, it might be possible to learn about the differences in usage patterns amongst different cultures.

REFERENCES

1. AxisMobile, <http://www.axismobile.com>
2. Bradner E., Wendy A., Kellogg, Erickson T., The Adoption and Use of 'Bubble': A Field Study of Chat in the Workplace, *ECSCW99*, pp.139-158, 1999.
3. Bruckman A., Resnick M., The MediaMoo Project: Constructionism and Professional Community, *Convergence*, 1,1, Spring 1995.
4. Burak A., Sharon T., Analyzing Usage of Location Based Services, *CHI03*, pp. 970-971, 2003
5. Viegas, F., Donath, J., Chat Circles, *CHI99*, pp. 9-16, 1999.
6. EMC World Cellular Database, <http://wcis.emc-database.com>
7. Freever, <http://www.freever.com>
8. FriendZone, <http://www.friendzone.ch>
9. Grinter R. & Eldridg, M., y do tngrs luv 2 txt msg?, *ECSCW01*, p.219, 2001.
10. Herbsleb J., Boyer D., Handel M. and Finholt T., Introducing Instant Messaging and Chat in the Workplace, *CHI02*, pp.171-178, 2002.
11. InirU, <http://www.iniru.co.il>
12. Isaacs E., Walendowski A. and Ranganthan D., Hubbub: A sound-enhanced mobile instant messenger that supports awareness and opportunistic interactions, *CHI02*, pp.179-186, 2002.
13. Jana R., Johnson T., Muthukrishnan S. and Vitaletti A., Location based services in a wireless WAN using cellular digital packet data (CDPD), *The Second ACM International Workshop on Data Engineering for Wireless and Mobile Access*, pp.74-80, 2001.
14. James C. and Reischel K., Text Input for Mobile Devices: Comparing Model Prediction to Actual Performance, *CHI01*, pp.365-371, 2001.
15. Lawson S., AT&T Wireless Help Callers Find Friends, *PCWorld.Com*, June 26 2002, <http://www.pcworld.com/news/article/0.aid.102269.00.asp>
16. Milgram S., The small world problem, *Psychology Today* 1, 61, 1967.

17. Perry M., O'hara K., Sellen A., Brown B., Harper R., Dealing With Mobility: Understanding Access Anytime, Anywhere, *ACM TOCHI*, 8(4), 2001.
18. Rivera K., Cooke N. and Bauhs J. A., The Effects of Emotional Icons on Remote Communication, *CHI96*, pp.99-100, 1996.
19. Rastimor O., Korolev V., Joshi A. and Finin T., Agents2Go: an infrastructure for location-dependent service discovery in the mobile electronic commerce environment, *The First International Workshop on Mobile Commerce*, pp.31-37, 2001.
20. Rheingold R., The Virtual Community, Homesteading on the Electronic Frontier, *MIT Press*, 2000.
21. Tang J., Yankelovich N., Begole J., Van Kleek M., Li F. and Bhalodia J., ConNexus to Awareness: Extending awareness to mobile users, *CHI01*, pp:121-128, 2001.
22. Townsend A. M., Life in the Real - Mobile Telephones and Urban Metabolism, *Journal of Urban Technology*, 7(2), pp.85-104, 2000
23. Townsend A. M., The Science of Location: Why the Wireless Development Community Needs Geography, Urban Planning, and Architecture, *Position Paper for CHI01 Wireless Workshop*, 2001

Fast Watermark Detection Scheme for Camera-equipped Cellular Phone

Takao Nakamura* Atsushi Katayama*

*NTT Cyber Space Laboratories
1-1 Hikarinooka, Yokosuka-Shi
Kanagawa 239-0847 Japan

Masashi Yamamuro* Noboru Sonehara†

†National Institute of Informatics
2-1-2 Hitotsubashi, Chiyoda-ku
Tokyo 101-8430, Japan

*{nakamura.takao, katayama.atusi, yamamuro.masashi}@lab.ntt.co.jp †sonehara@nii.ac.jp

ABSTRACT

Digital watermarking technology would be very useful as part of a related service introduction system (RSIS); this system provides related information to content, and the function of watermark in RSIS is analogous to that of barcode, i.e., watermark binds content ID to analog content such as an image on printed material.

In this paper, we focus on a camera-equipped cellular phone used as a terminal for RSIS, and propose a fast watermark detection scheme from a captured image. The proposed scheme consists of two processes, one is to correct geometric distortion of the captured image, and the other is to detect watermark information from the rectified image. We also propose a new watermarking algorithm which is robust against small geometric distortion and suitable for the proposed scheme. Moreover, we introduce a quantitative evaluation method for indicating detection reliability, which is indispensable for RSIS service.

Finally, we show that the proposed scheme enables users to detect embedded information in approximately one second, even when implemented as a Java application on a cell phone with limited resources, and report experiments that confirm the proposed scheme's efficiency.

Categories and Subject Descriptors

I.4 [Image processing and computer vision]: Miscellaneous

General Terms

Algorithms, Performance, Experimentation

Keywords

Digital watermark, Cellular phone, Camera

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MUM2004 October 27–29, 2004, College Park, Maryland, USA.
Copyright 2004 ACM 1-58113-981-0/04/10 ...\$5.00.

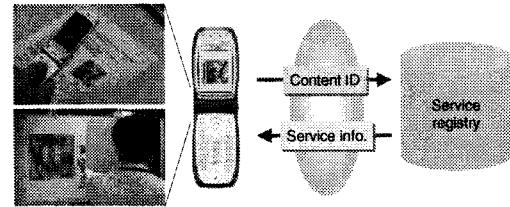


Figure 1: Related Service Introduction System [4] using digital watermark

1. INTRODUCTION

Digital watermarking technology is used to embed additional information into content with slight modification to render it imperceptible, and to extract that information from embedded content [1-3,8,10]. Watermarking technology is also characteristically robust against degradation and transformation of content, such as distortion of lossy compression or scaling, rotation, and cropping of images. For images, several interesting approaches enable extracting watermark information even from a image re-digitalized after an analog output with severe degradation and transformation, e.g., D/A → A/D conversion[2,3].

Sakamoto proposed the related service introduction system (RSIS) in which an end-user can obtain service information by sending a content ID that has been previously bound to the content itself [4]. RSIS with watermarking technology as a binding mechanism makes possible linking an object in the real world to those in the cyber world. For example, printed materials can lead to a site on the World Wide Web. Although people can see a barcode, a technology that allows such linkage [5], it requires some space on a printable area, and they must be conscious of the link between the barcode and content. In contrast, watermarking technology, offers invisible integration with content, does not give rise to the issues of space efficiency, or user awareness, rather it enables providing an intuitive operating interface.

For these reasons, watermarking technology has great potential for use in RSIS; however, extracting information embedded in content requires a great deal of processing power. For this reason a personal computer (PC) has been expected to be the user terminal for RSIS [4]. From the viewpoint of service provision, as convenience is a must, a mobile terminal with an analog input device is much more preferable

than a PC. For example, anyone could easily retrieve service information anywhere and anytime from an advertising display on the street, a screen image on TV or a picture in a magazine whenever a mobile terminal is available.

In point of fact, cell phone users are growing in number every year, and the newest cell phones are camera equipped. Hence, a camera-equipped cell phone is a worthy target as our terminal device. However, as the processing capacity of a cell phone is not large, a new detection scheme is definitely necessary for this purpose.

In this paper, we propose a fast watermark detection scheme that corrects geometric distortion with a calibration pattern and a new watermarking algorithm. We also introduce a quantitative evaluation method that can indicate detection reliability and that is indispensable for providing of RSIS service. Finally we conclude by showing experimental results prove that the proposed scheme sufficiently meets requirements.

2. REQUIREMENTS

2.1 High speed

There are two different approaches to implement watermark detection onto a cell phone. One approach is to provide a detection process as a built-in application. This is the most suitable in terms of processing speed, but requires a certain period of time before commercializing the product. The other is to implement a detection process as loadable software on an existing terminal so that RSIS would be available straightaway. From the viewpoint of business strategy, uncertainty of marketability causes businesses to prefer implementing via a loadable module at first, since loadable software is a faster way to provide a service.

Implementing Java applications on cell phones has become commonplace in Japan, and is called "i-appli" [6]. To implement a watermark detection process as i-appli suffers in comparison to original equipment implementation because the processing power of cell phones is very low. Furthermore, implementing this service on the Java VM interpreter is much harder. Adoption of an existing terminal, however, has a significant advantage as described above. For this reason, we selected i-appli implementation as prerequisite.

Since former studies show that a time interval within 2 seconds is the only acceptable one with respect to human interface [7], we applied the same rule for the time interval from taking a picture of an image embedded watermark to obtaining a detection result.

2.2 Robustness

The process of capturing an image on analog media with a cell phone camera causes distortion, which consists of 1) D/A conversion such as degradation due to printing, 2) A/D conversion such as degradation due to camera device performance, and 3) geometric transformation such as projection distortion due to shooting angle. In consequence, the watermarking scheme requires robustness against these causes of distortion.

3. PROPOSED SCHEME

3.1 Design Policy

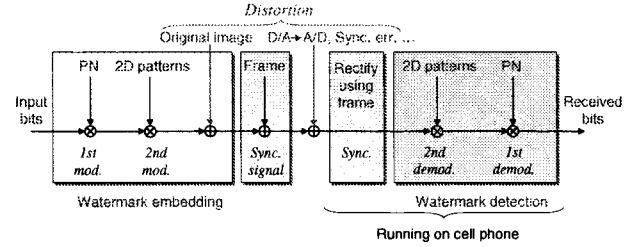


Figure 2: Model of proposed watermarking scheme

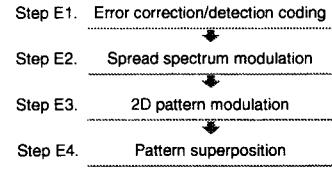


Figure 3: Operation flow for watermark embedding

A model of the proposed watermarking scheme is shown in Fig. 2. Considering the digital watermark as a communication model, we must transmit modulated watermark information through a channel that has severe distortion – Examples are the original image, $D/A \rightarrow A/D$ conversion, and synchronization error caused by shooting angle– and then perform demodulation.

We achieve robustness against serious synchronization error such as projection distortion due to the shooting angle by adding a calibration pattern to the image and thereby recovering synchronization [5]. Next, by applying 2D pattern modulation with stable characteristics at each small block, we achieve robustness against synchronization error caused by small distortion such as that from error resulting from correction of a geometric distortion , a gentle curvature of the printed materials, and lens distortion. Finally, robustness against additive noise is achieved by employing spread spectrum as the first order modulation [8].

The proposed embedding and detection methods are described in detail below.

3.2 Watermark embedding

Figure 3 shows the operation flow for embedding a watermark. The original image $I = \{I_{x,y}\}$ ($Width \times Height$ pixels), k -bit watermark information ID , and embedding strength a , which controls the balance between the image quality and robustness, are input into the embedding process. In the following discussion, pixel value $I_{x,y}$ indicates a gray-scale. For a color image, the brightness component is processed as $I_{x,y}$.

3.2.1 [Step E1.] Error correction/detection coding

Using error correction/detection coding for watermark information ID , the n -bit codeword $Code$ is generated. The capabilities of error correction/detection code mentioned here are important to the discussion pertaining to the evaluation of the reliability of the watermark detection as described later.

3.2.2 [Step E2.] Spread spectrum modulation

With predetermined integer value N , each bit of $Code$, c_j ,

is repeated $l = N^2/n$ times to produce a redundant sequence with length N^2 :

$$\{b_i\} = \underbrace{c_0 \cdots c_0}_{l \text{ times}} \underbrace{c_1 \cdots c_1 \cdots}_{N^2} \cdots c_{n-1} \cdots c_{n-1}. \quad (1)$$

A pseudo random sequence with length l , $\{r_i^{(j)}\}$ ($r_i^{(j)} = \pm 1$, $\sum_{i=0}^{l-1} r_i^{(j)} = 0$), is prepared for each $j = 0 \dots n-1$, and sequence with length N^2 , $\{r_i\}$ is obtained by

$$\{r_i\} = \underbrace{r_0^{(0)} \cdots r_{l-1}^{(0)}}_{N^2} \underbrace{r_0^{(1)} \cdots r_{l-1}^{(1)}}_{N^2} \cdots r_0^{(n-1)} \cdots r_{l-1}^{(n-1)}. \quad (2)$$

Then, the repeated message $\{b_i\}$ is modulated by a carrier signal $\{r_i\}$. This modulation is known as Direct Sequence Spread Spectrum (DS-SS) modulation, and is performed by

$$s_i = b_i r_i \quad (i = 0 \dots N^2 - 1), \quad (3)$$

where the value of b_i is reread as -1 when the bit value is "0", and the value of b_i is reread as $+1$ when the bit value is "1" [8].

Furthermore, a pseudo randomly determined permutation is prepared as

$$\begin{pmatrix} 0 & 1 & 2 & 3 & \cdots & N^2 - 1 \\ o_0 & o_1 & o_2 & o_3 & \cdots & o_{N^2-1} \end{pmatrix}. \quad (4)$$

With this permutation, the order of $\{s_i\}$ element is scrambled and the embedded sequence $\{t_i\}$ is obtained as below.

$$t_i = s_{o_i} \quad (i = 0 \dots N^2 - 1). \quad (5)$$

Through the process described above, robustness based on DS-SS modulation is obtained. Furthermore, scramble by permutation is brought into effect as interleave coding, i.e. reducing the imbalance of robustness among bits.

3.2.3 [Step E3.] 2D pattern modulation

The two 2D sine curves with 90 degree rotational symmetry, $P^- = \{P_{x,y}^-\}$, $P^+ = \{P_{x,y}^+\}$, in which frequencies relative to image size are ($Freq, Freq$) and ($Freq, -Freq$), are set as

$$P_{x,y}^- = \sin 2\pi(f_x \cdot x + f_y \cdot y) \quad (6)$$

, and

$$P_{x,y}^+ = \sin 2\pi(f_x \cdot x - f_y \cdot y) \quad (7)$$

$$(x = 0 \dots Width - 1, y = 0 \dots Height - 1),$$

where

$$f_x = \frac{Freq}{Width}, \quad f_y = \frac{Freq}{Height}. \quad (8)$$

P^- and P^+ are divided into $N \times N$ blocks respectively, and block image groups $\{P_{x,y}^{-(h,v)}\}$ and $\{P_{x,y}^{+(h,v)}\}$ are obtained. Next, for each block position (h, v) in the block fragmentation above, following raster scanning of the blocks one-by-one, the element of the embedded sequence t_i ($i = h + vN$) is selected. According to the value of t_i , watermark pattern block image $P_{x,y}^{(h,v)}$ is selected as

$$P_{x,y}^{(h,v)} = \begin{cases} P_{x,y}^{-(h,v)}, & \text{if } t_i = -1 \\ P_{x,y}^{+(h,v)}, & \text{if } t_i = +1. \end{cases} \quad (9)$$

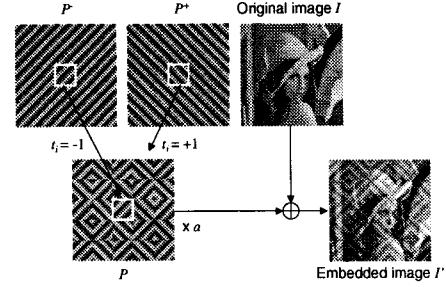


Figure 4: 2D pattern modulation

By determining $P_{x,y}^{(h,v)}$ for all (h, v) , a watermark pattern $P = \{P_{x,y}\}$ with $Width \times Height$ pixels is obtained, as illustrated in Fig. 4.

3.2.4 [Step E4.] Pattern superposition

The watermark pattern P is amplified by embedding strength a , and watermark embedded image $I' = \{I'_{x,y}\}$ is obtained by superposing the original image I on P (Fig. 4) as

$$I'_{x,y} = I_{x,y} + aP_{x,y} \quad (10)$$

$$(x = 0 \dots Width - 1, y = 0 \dots Height - 1).$$

Basically, the process of watermark embedding is performed as described above, and we apply adaptive pattern superposition to improve the balance between the image quality and the robustness of the watermark.

3.2.5 [Step E4'.] Adaptive pattern superposition

In the watermark embedding process described above, the strength of the watermark pattern is uniform over the entire image. In some parts of the image, however, there may be perceptible degradation caused by added noise, while in other parts the noise may be difficult to detect. In such case, robustness can be improved while maintaining perceived image quality by increasing the relative strength of the embedded watermark pattern in the parts of the image where the resulting degradation is difficult to perceive [3].

First, considering the Human Visual System for noise, the weight matrix $W = \{W_{x,y}\}$ (with the same size as original image) is generated based on the original image. When the pixel value of each coordinate I changes, the modifiable range of the target pixel value that makes the visual stimulus constant is suitable for W . Finally, we obtain the embedded image $I' = \{I'_{x,y}\}$ as

$$I'_{x,y} = I_{x,y} + aW_{x,y}P_{x,y} \quad (11)$$

$$(x = 0 \dots Width - 1, y = 0 \dots Height - 1),$$

thereby amplifying the value of each point P by using the element value corresponding to W [3].

3.3 D/A → A/D Conversion and Synchronization

After generating the embedded image through the process described in Section 3.2, a frame is placed around the image to rectify geometric distortion that occurs when photographing [9]. Then, the framed image is output as analog media, for example, by printing it.

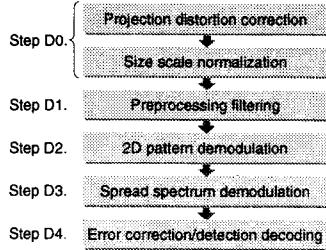


Figure 5: Operation flow for watermark detection

The addition of a frame to the image improves its usability because the frame works not only as a calibration pattern, but also as an indicator showing that the watermark has been embedded. This is similar to the function of the underline used in hypertext for WWW documents.

The camera-equipped cell phone re-digitalizes the framed image. The synchronization process, correction of geometric distortion, must be performed before watermark detection. When the frame is recognized, the coordinates of the four corner points are located. Given a four-point template that corresponds to these four points, the parameters of projection transformation can be estimated. With this estimation, projection distortion of the captured image can be corrected by inverse transformation. We proposed the method for this function, even through i-appli, corner recognition and projection distortion correction can be implemented at high-speed processing of 650 msec [9]. Furthermore, the estimated value of the width of the frame can be acquired at the same time. The width value enables accurately extracting the target image for watermark detection, and this accuracy is distinctly effective.

3.4 Watermark Detection

The target detection image can be acquired as different primarily in the size scale and each rotation factor of 90 degrees, i.e. 0, 90, 180 and 270, as compared with the image right after embedding the watermark. The target image also contains the small geometric distortion such as a gentle curvature of the printed materials and an error resulting from correction of projection distortion.

Normalizing size enables eliminating the difference in scale since watermarks are embedded with relative scale. However, although fractional rotation can be corrected by correcting projection distortion [9], uncertainty of 90-degree rotation that depends on shooting angle still remains unsolved.

Even so, the robustness against geometric distortion required by watermark detection is sufficient to deal only with the small distortion and uncertainty of 90 degrees rotation factor. Figure 5 illustrates the operational flow of watermark detection.

3.4.1 [Step D0.] Size scale normalization

The size of the detection target image is normalized to a predetermined size. The normalization not only absorbs the difference in size scale, but also enables the efficient processing of Step D1, preprocessing filtering, as described in the next section. Actually, generation of the target image completes the normalization process, with the template in normal size. The normalization generates a squared image

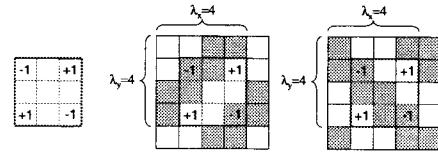


Figure 6: Convolution operator of preprocessing filter and its effects

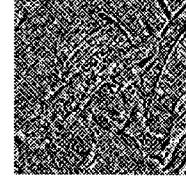


Figure 7: Example of preprocessing filtered image

and its size $\text{Normalsize} \times \text{Normalsize}$ is set as

$$\text{Normalsize} = 4\text{Freq}. \quad (12)$$

3.4.2 [Step D1.] Preprocessing filter

Using the preprocessing filter to normalize the target image decreases the effects of original image on the watermark signal and increases robustness of watermark.

As illustrated in Fig. 6, the preprocessing filter is a four-pixel reference convolution operator. From Eqs. 6–8, and 12, the two 2D sine curves in the normalized image have wavelength of $\lambda_x = \lambda_y = 4$ in both the x, y directions. By using this operator, both components of the two 2D sine curves can be efficiently picked out by only one scanning pass as shown in Fig. 6. This contributes to the further increase in the processing speed.

Furthermore, Each filtered pixel value is clipped into positive, negative, or 0 value. The original image signal is further decreased as noise by the clipping process. Figure 7 shows an example of a preprocessing filtered image.

3.4.3 [Step D2.] 2D pattern demodulation

The filtered image is divided into $N \times N$ individual blocks, and on each block ($M \times M$ pixels, $M = \text{Normalsize}/N$), energy of the frequencies corresponding to the two 2D sine curves is calculated. Specifically, at first the two convolution operators that correspond to the 2D sine curves illustrated in Fig. 8 are applied to the block pixels. For the pixel of the block at position (h, v) , the pixel values obtained after applying convolution operator C^- and C^+ are set to $e_{x,y}^{-(h,v)}$ and $e_{x,y}^{+(h,v)}$ ($x, y = 0 \dots M - 1$), respectively. On $\{e_{x,y}^{-(h,v)}\}$ and $\{e_{x,y}^{+(h,v)}\}$, the sum of the absolute values of all the ele-

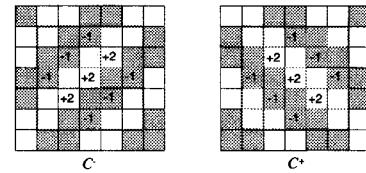


Figure 8: Two convolution operators for 2D pattern demodulation

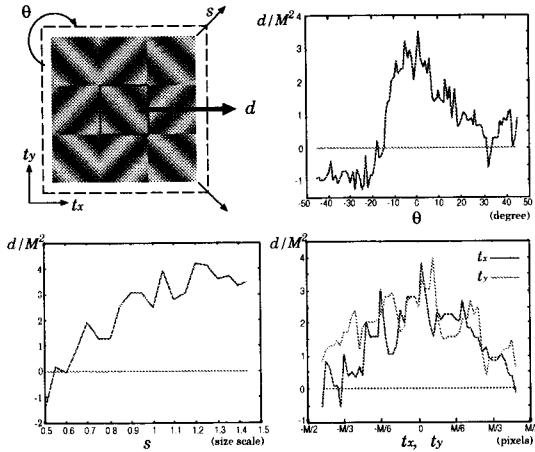


Figure 9: Response of detection value d to geometric distortions

ments in a block is obtained as

$$E_{h,v}^- = \sum_{x=0}^{M-1} \sum_{y=0}^{M-1} |e_{x,y}^{-(h,v)}|, \quad E_{h,v}^+ = \sum_{x=0}^{M-1} \sum_{y=0}^{M-1} |e_{x,y}^{+(h,v)}|, \quad (13)$$

respectively. $E_{h,v}^-$ and $E_{h,v}^+$ are regarded as the energy of frequencies, and detection value $d_{h,v}$ derived from the block at the position (h,v) is obtained as the difference between the two frequency energy levels by

$$d_{h,v} = E_{h,v}^+ - E_{h,v}^-. \quad (14)$$

By acquiring $d_{h,v}$ for all blocks, detection value matrix $D = \{d_{h,v}\}$, an $N \times N$ square matrix, is obtained.

The basic idea of this process is to find which sign is embedded superiorly in the block, and measure the degree of its superiority. If there is absolutely no noise, then the sign of detection value $d_{h,v}$ and the sign for the embedded sequence element $t_i (i = h + vN)$ exactly match, and absolute value $|d_{h,v}|$ generally becomes large. Conversely, when there are various noise factors such as those from the original image or the error in rectifying geometric distortion, the degree to which the sign of the $d_{h,v}$ and that for t_i , match is lower, and $|d_{h,v}|$ often becomes a comparatively low value.

In consideration of this, the negative influence of the despreading process described in the next section is considered to be small. Figure 9 shows how $d_{h,v}$ responds to rotation, scaling, and translation of image. We find that there is almost no reversal in the sign of d , within the range of ± 10 degrees for rotation, that of 0.6 to 1.5 for scaling factor, and that of $\pm M/3$ pixels for translation, consequently 2D pattern modulation is robust against small geometric distortions.

3.4.4 [Step D3.] Spread spectrum demodulation

A sequence $\{g_i\}$ ($i = 0 \dots N^2 - 1$) is obtained by raster scanning of the each element of detection value matrix D . Using pseudo random permutation in Eq. 4, we de-scramble the order of $\{g_i\}$ element by inverse permutation as

$$h_{o_i} = g_i \quad (i = 0 \dots N^2 - 1). \quad (15)$$

Next, the subsequence of $\{h_i\}$ with $l = N^2/t$ length is ob-

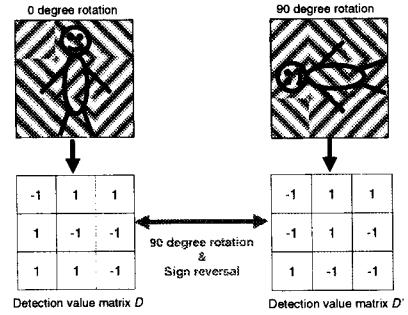


Figure 10: Influence of 90-degree rotation of image on the detection value matrix

tained as

$$x_i^{(j)} = h_{i+jl} \quad (i = 0 \dots l - 1) \quad (16)$$

for each $j = 0 \dots n - 1$. $\{x_i^{(j)}\}$ is a extracted sequence from $\{h_i\}$ according to the interval of c_j in Eq. 1. And $\{x_i^{(j)}\}$ is normalized to the mean value of 0 and the variance of 1. Given the same pseudo random sequence $\{r_i^{(j)}\}$ as in Section 3.2.3, DS-SS demodulation, namely despreading operation, is performed [8]. Correlation value ρ_j corresponding to bit position j is obtained as

$$\rho_j = \sum_{i=0}^{l-1} x_i^{(j)} r_i^{(j)}. \quad (17)$$

Then, detected bit value c'_j is determined as below.

$$c'_j = \begin{cases} 0 & , \rho_j < 0 \\ 1 & , \rho_j \geq 0. \end{cases} \quad (18)$$

3.4.5 [Step D4.] Error correction/detection decoding

Constructing the detected bit string $\{c'_j\}$ as the detected codeword $Code'$, error correction/detection decoding processing are performed. In this process, correctable bit errors are properly corrected, and the watermark information ID' of k -bits is decoded. Furthermore, if uncorrectable bit errors are detected, the detection process is terminated in a proper manner. For a case beyond the ability of error detection, we might have a result that a detected watermark information is correct in spite of being detected incorrectly. The solution to this problem is described later in detail.

3.4.6 Robustness against 90-degree rotation

The detection process described so far does not account for images with a 90-degree rotation factor.

For the case of watermark pattern with 90-degree rotation, the block layout may well be rotated 90 degrees. In addition to the layout rotation of the blocks, the two 2D sine curves are simultaneously interchanged since they are symmetric to 90-degree rotation.

Given the detection target image with 90-degree rotation as input, illustrated in Fig. 10, the detection value matrix D' is obtained after Step D2. Compared with the detection value matrix D for the image with 0 degree rotation, the layout of elements of D' is rotated 90 degrees, and the signs of all elements are reversed. The 270-degree case is the same as the 90-degree case. In the case of a 180-degree rotation, the sign is not reversed.

Hence, given an image with any 90-degree rotation, the detection value matrix D can be obtained. Next, matrices D_{90° , D_{180° , and D_{270° are generated from D by 90, 180, and 270 degrees rotation, respectively. Under these conditions, the signs of all the elements for D_{90° and D_{270° are reversed. The processing for Step D3 is performed for each of these four matrices. The detection reliability index ρ described in Section 3.5 is calculated for each matrix and the one with ρ at maximum is selected. The corresponding detection codeword $Code'$ is adopted, error correction/detection decoding is performed, finally the detection watermark information ID' is obtained.

Based on the processing described above, the processing load increases because of rearranging the detection value matrix and doing Step D3 three extra times. However, the experimental results described in Section 4 make clear that the processing as described in this section is efficient.

3.5 Reliability of Watermark Detection

In general, one of the important requirements for watermarking technology is to suppress the probability of a detection error [10]. This error is called the false positive and includes

- A. To commit a successful detection even though the image does not contain watermark information, and
- B. To commit a successful detection even though the detected information differs from the correct.

With regard to these requirements, we introduce a detection reliability index that represents the accuracy of the detected watermark information.

3.5.1 Reliability evaluation for Type A error

Consider the following equation,

$$y_j = \frac{1}{\sqrt{l}} \sum_{i=0}^{l-1} x_i^{(j)} r_i^{(j)} = \frac{1}{\sqrt{l}} \rho_j. \quad (19)$$

Now set up a null hypothesis

$$H_0 : \text{A pseudo random sequence } \{r_i^{(j)}\} \text{ is independent of a detection target sequence } \{x_i^{(j)}\}.$$

If l is sufficiently large, we can approximate that y_j is a random variable that follows a standard normal distribution $N(0,1)$ by exploiting the central limit theorem. Next, let $\{z_j\}$ denote a sequence which is obtained from $z_j = |y_j|$, then define the detection reliability index ρ as

$$\rho = \frac{1}{\sqrt{n}\sigma_z} \sum_{j=0}^{n-1} (z_j - \mu_z) \quad (20)$$

where the mean value of z_j is μ_z and the variance is σ_z^2 . If n is sufficiently large, again by the central limit theorem, we can approximate that ρ is a random variable that follows $N(0,1)$. As $\mu_z = \sqrt{2}/\pi$ and $\sigma_z^2 = 1 - 2/\pi$ under the hypothesis H_0 , we can evidently calculate ρ . Figure 11 shows the simulation results of the ρ distribution with no watermark ($l = n = 32$) as a good estimation.

When ρ deviates on a large scale from the $N(0,1)$, H_0 is rejected, i.e., $\{r_i^{(j)}\}$ is dependent on $\{x_i^{(j)}\}$. By hypothesis

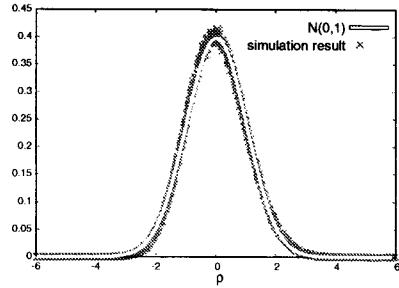


Figure 11: Simulation result of ρ distribution with no watermark (White line is $N(0,1)$)

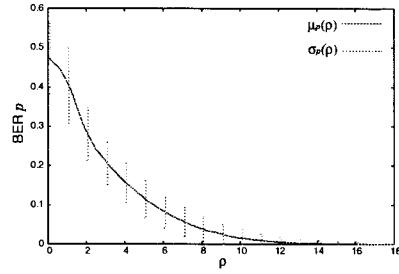


Figure 12: Mean value of BER $\mu_p(\rho)$ and standard deviation $\sigma_p(\rho)$ for ρ by simulation

testing, the existence of watermark can be quantitatively established. The false positive rate of Type A for ρ is

$$P_A = \frac{1}{\sqrt{2\pi}} \int_{\rho}^{\infty} e^{-\frac{x^2}{2}} dx. \quad (21)$$

In actual implementation, the allowable false positive rate $P_A = \varepsilon$ should be predetermined from the service quality requirements, and the corresponding threshold $\rho = T_\rho^{(\varepsilon)}$ should be calculated in advance with Eq. 21. We only have to judge whether ρ is greater than the threshold.

3.5.2 Reliability evaluation for Type B error

Discussion regarding Type B error is more difficult than Type A, because we must measure the degree of deterioration of the detected watermark information. First, we measure the degree of degradation based on a simulation. Figure 12 shows the simulation result, the mean value of bit error rate μ_p and standard deviation σ_p for ρ , in the case of $n = 32$ bits. As ρ is an indicator of the existence of the watermark, $\mu_p(\rho)$ decreases monotonously as ρ increases.

Next, we can have the probabilities P_C that a codeword is decoded correctly, P_D that errors are detected, and P_E that a codeword is decoded incorrectly and no errors are detected, for error correction/detection code used in Step E1 and Step D4 ($(P_C + P_D + P_E = 1)$). Generally, these probabilities are determined based on BER p . Let $f_p(p)$ denote the probability density function which gives distribution of BER p for ρ . Then with the probability $P_E(p)$ at BER p and $f_p(p)$, the false positive rate P_B of Type B error for given ρ is calculated as,

$$P_B = \int_{-\infty}^{\infty} P_E(p) f_p(p) dp. \quad (22)$$

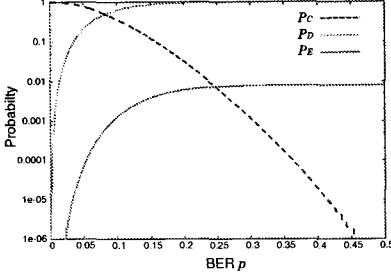


Figure 13: Probabilities of (31+1, 16) extended BCH code (2-tuple bit error correction)

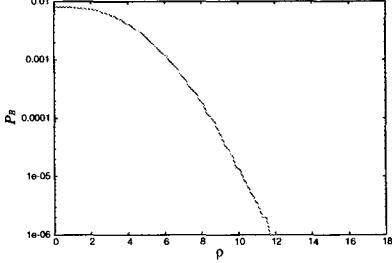


Figure 14: Type B false positive rate P_B for ρ

Figure 13 shows $P_C(p)$, $P_D(p)$, and $P_E(p)$ of (31+1, 16) extended BCH code (2-tuple bit error correction) [11]. Finally, P_B for ρ was obtained from $f_\rho(p)$ of the simulation results and $P_E(p)$ of this code, as plotted in Fig. 14.

We can have the threshold $T_\rho^{(\delta)}$ for the allowable false positive rate $P_B = \delta$ in the same way as Type A. For example, if we choose $\delta = 10^{-3}$, we obtain the threshold $T_\rho^{(\delta)} \simeq 6.2$ from figure 14.

As shown in the discussion above, we can quantitatively evaluate the false positive rate P_B with the detection reliability index ρ . When applying $T_\rho^{(\delta)} = 6.2$ as the threshold for Type A error, we obtain $\epsilon \simeq 2.82 \times 10^{-10}$, which seems fairly severe for ϵ . If we use threshold $T_\rho^{(\epsilon)}$ ($T_\rho^{(\epsilon)} < T_\rho^{(\delta)}$) corresponding to ϵ as a milder criterion, then we can obtain three statuses of detection reliability level:

- (a) Watermark does not exist ($\rho < T_\rho^{(\epsilon)}$)
- (b) Watermark does exist, but the detected information may be incorrect ($T_\rho^{(\epsilon)} \leq \rho < T_\rho^{(\delta)}$), and
- (c) Watermark is correctly detected ($\rho \geq T_\rho^{(\delta)}$).

For (b), the application software is supposed to prompt the user to retry shooting.

4. EXPERIMENTAL RESULTS

In order to validate the effectiveness of the proposed scheme, we implemented it as i-appli and conducted various experiments. The i-appli only supports integer arithmetic, which is a restriction on implementation [6]. Almost all of the detection processes described above, however, are constructed with integer addition, subtraction and multiplication. Furthermore, for a few processing such as the detection reliabil-

Table 1: Processing Speed

Step	i-appli	PC
D0. Projection dist. correction	650 msec.	9.00 msec.
D1. Preprocessing filtering	55 msec.	0.22 msec.
D2. 2D pattern demodulation	323 msec.	0.93 msec.
D3. SS demod. (4 times)	16 msec.	0.14 msec.
D4. Error c./d. decoding	1 msec.	0.001 msec.
Total	1045 msec.	10.29 msec.

Table 2: Embedding strength and PSNR

	α	β	γ
Lena	37.762	35.245	31.936
Baboon	32.004	29.530	26.320

ity index calculation, a floating-point operation is required. These parts can be implemented by fixed-point coding.

In the experiments, (31+1, 16) extended BCH code (2-tuple bit error correction) was used as the error correction/detection code. Therefore the length of the watermark information k was 16 bits, and the code length n was 32 bits. Furthermore, we adopted the configuration in reference [3] for the adaptive pattern superposition in Section 3.2.5.

4.1 Processing Speed

Table 1 gives the processing time of i-appli (non-optimized coding) and that of a PC (C language implementation on a 1.6 GHz Pentium 4) for reference. Here, the projection distortion correction processing time is an averaged value since it varies according to a captured image [9]. The watermark detection processing time after correction, however, was always constant. From these results, we found that the processing time for detection from rectified image is 395 msec., and the total processing time of watermark detection is 1045 msec. These times satisfy the requirements in Section 2.1. Furthermore, we confirmed that the processing time for Step D3 was 16 msec, while SS demodulation was performed four times as described in Section 3.4.6. Accordingly the proposed method for handling 90-degree rotation is efficient.

4.2 Robustness

In order to evaluate the robustness of the proposed scheme, we performed watermark detection experiments using two evaluation images (Lena and Baboon) with three levels of embedding strength a ($a = \alpha, \beta, \gamma, \alpha < \beta < \gamma$).

Table 2 shows PSNR between the original and embedded images. Figure 15 shows the embredded image with a frame under $a = \beta$. The reason for the difference in PSNR between Lena and Baboon, even using the same embedding strength, is due to the adaptive embedding process as described in Section 3.2.5.

We conducted an experiment on robustness by shooting the printed images with a camera-equipped cell phone. We used a color laser printer and printed out two different sizes of embedded images with a frame, 5 cm square and 25 cm square. If ρ is greater than $T_\rho^{(\delta)} = 6.2$ and no errors are detected by error detection decoding, the detection is deemed successful.

In our field experiments, we suggested the test subjects only capture the entire frame, since this instruction is required for frame recognition. The detection results are given in Table 3. Figure 16 shows example images of detection failure. We found that the detection fails easily when there was



Figure 15: Embedded images (embedding strength $a = \beta$) with frame (Left: Lena, Right: Baboon)

Table 3: Detection results (when no specific instructions are given for a way of shooting)

	a	5cm×5cm		25cm×25cm	
		success	$\bar{\rho}$	success	$\bar{\rho}$
Lena	α	43%	7.05	88%	13.09
	β	87%	11.70	92%	18.49
	γ	93%	17.91	95%	22.36
Baboon	α	43%	6.66	67%	8.08
	β	93%	12.72	92%	14.32
	γ	95%	19.59	95%	21.09

motion blur due to camera movement and when the degree of projection distortion was large.

Next, the test subjects were instructed to capture the image from a position right in front of it to decrease the projection distortion. The detection results are given in Table 4. Detection accuracy was greatly improved by using the instruction above.

Furthermore, compared with the 5 cm squared images, the detection success rate for the 25 cm square image was higher on the whole. The reason of this result might be caused by the difficulty in capturing the image in focus. The capture of 5 cm squared images is an inevitable requirement for close-up shooting.

From the results above, for a 5 cm squared image with $a = \beta$, captured without intention by the test subjects, the probability for detection failure was 13% for Lena and 7% for Baboon. However, if instructed to reshoot from right in front of the image, the test subjects successfully detected the watermark information in almost all cases.

Of 1440 trials, no case of Type B false positive occurred throughout the entire set of experiments.

5. CONCLUSION

We proposed a fast digital watermark detection scheme for camera-equipped cell phones, that utilizes projection dis-

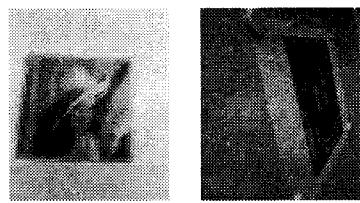


Figure 16: Example images of detection failure (288×352 pixels)

Table 4: Detection results (when instructed to shoot from right in front of the image)

	a	5cm×5cm		25cm×25cm	
		success	$\bar{\rho}$	success	$\bar{\rho}$
Lena	α	93%	10.50	100%	18.54
	β	100%	17.40	100%	22.53
	γ	100%	24.30	100%	28.48
Baboon	α	87%	9.65	98%	11.22
	β	100%	17.57	100%	19.23
	γ	100%	25.07	100%	27.06

tortion correction employing frames and a digital watermark algorithm which is robust against small geometric distortion. Furthermore, we quantitatively evaluated the reliability of watermark detection, and showed that the proposed evaluation method can guarantee the desired quality of service. We implemented our scheme as a Java application on cell phones. In our experiments, the scheme achieved high-speed detection of approximately one second. We also confirmed that the proposed scheme is sufficiently robust.

As future work, we intend to analyze the image dependency of the proposed scheme's robustness and the differences between analog output methods such as commercial printing, CRTs, and LCD displays. With the results of this analysis, we hope to improve the trade off between visual quality and robustness.

6. REFERENCES

- [1] A.Z.Tirkel, G.A.Rankin, R.M.van Schyndel, N.R.A.Mee, C.F.Osborne, Electronic Water Mark, Digital Image Computing, Technology and Applications (DICTA-93), pp. 666-672, 1993.
- [2] Cox,I.J. Kilian,J., Leighton,T., and Shamoon,T., Secure spread spectrum watermarking for images, audio, and video, Proceedings of the 1996 IEEE International Conference on Image Processing, pp. 243-256, 1996.
- [3] T.Nakamura, H.Ogawa, A.Tomioka, and Y.Takashima, Improved Digital Watermark Robustness against Translation and/or Cropping of an Image Area, IEICE Trans. Fundamentals, Vol. E83-A, No. 1, Jan. 2000.
- [4] H.Sakamoto, H.Fujii, S.Irie, and H.Yamashita, Service Gateway to Enable the Introduction of Content Related Services, In Proceeding of the IEEE International Conference on Multimedia and Expo (ICME) 2001, pp. 637-640, 2001.
- [5] QR Code.com, <http://www.qrcode.com/>
- [6] Specifications of Java for i-mode, http://www.nttdocomo.co.jp/english/p_s/i/index.html
- [7] Ben Shneiderman, Designing the user interface 2nd edition, ISBN:0-201-57286-9, Addison-Wesley Longman Publishing Co., Inc., 1992.
- [8] Jonathan K. Su, Frank Hartung, and Bernd Girod, Digital Watermarking of Text, Image, and Video Documents, Computers & Graphics 22(6), 1998.
- [9] A.Katayama, T.Nakamura, M.Yamamoto, and N.Sonehara, New High-speed Frame Detection Method: Side Trace Algorithm (STA) for i-appli on Cellular Phones to Detect Watermarks, In Proceedings of the Mobile And Ubiquitous Multimedia 2004 (MUM2004), Oct. 2004 (to be submitted).
- [10] W.Bender, D.Gruhl, and N.Morimoto, Techniques for Data Hiding, In Proceedings of the SPIE Conference on Storage and Retrieval for Image and Video Databases III, Vol. 2420, pp. 164-173, Feb. 1995.
- [11] F.J.MacWilliams and N.J.A.Sloane, The Theory of Error Correcting Codes, North-Holland Publishing, 1977.

New High-speed Frame Detection Method: Side Trace Algorithm (STA) for i-appli on Cellular Phones to Detect Watermarks

Atsushi Katayama* Takao Nakamura*

Masashi Yamamuro* Noboru Sonehara†

*NTT Cyber Space Laboratories
1-1 Hikarinooka, Yokosuka-Shi
Kanagawa 239-0847 Japan

†National Institute of Informatics
2-1-2 Hitotsubashi, Chiyoda-ku
Tokyo 101-8430, Japan

*{katayama.atusi, nakamura.takao, yamamuro.masashi}@lab.ntt.co.jp †sonehara@nii.ac.jp

ABSTRACT

We developed a system that enables a camera-equipped cellular phone to read digital watermarks embedded in various media in real time, and that presents to the user a link to a Web page, video, or music associated with that watermark information. A picture captured by a camera is the result of applying a projective transformation combining rotation, scaling, and tilting to the original picture. The picture must be subjected to an inverse projective transformation prior to reading the watermark in order to return it to the same geometric form as the original picture. This inverse transformation requires transformation parameters, and the corners of the picture outline can be used as feature points for determining these parameters. In this paper, we propose a Side Trace Algorithm (STA) that reduces the processing time required to find corners of the picture less than 1/100 that when using the Hough transform and the conventional pattern matching, and present results of its implementation.

Categories and Subject Descriptors

I.5.5 [Pattern Recognition Implementation]:

General Terms

Algorithms, Performance, Experimentation

Keywords

Corner detection, Line detection, Projective transformation, Watermark, Cellular phone, Camera, i-appli, Java

1. INTRODUCTION

The use of “resolution applications” to provide services that combine Internet functions and camera-equipped cellu-

lar phones is spreading[1]. These applications extract information from camera pictures and index that information to introduce users to Web sites and other areas of the digital world. Fig.1 shows the concept behind such a service. We are working to implement a service of this kind using the high-speed digital-watermarking scheme proposed by Nakamura[5] to extract information from a camera picture. In this regard, in the past, a system would send the camera picture to a remote server that would then read the digital watermark[2]. In the construction and evaluation of this system, though, it was found that the response delay caused by the transmission time presented a problem for the user interface. Therefore, there is a need for a system that can complete all tasks from capturing the image to reading the watermark in the terminal.

One method for accomplishing this in the cellular phone itself is to use native software and a dedicated LSI (Large Scale Integrated circuit) incorporated in the terminal. The implementation cost in this case is a problem however, and to reduce this cost it was decided to employ “i-appli”, a specialized system for cellular phones now finding widespread use in Japan[3]. It is a Java-based programming standard established by NTT DoCoMo to provide an environment for executing user programs on a cellular phone. It makes Java-

programs can be downloaded to the terminal freely, thereby reducing the implementation cost to nearly zero. The draw-

back to this method is that user programs run on a JAVA Virtual Machine resulting in slow execution. When attempting to read a digital watermark from a camera picture, we

must keep in mind that the camera picture represents a transformation of the original picture by scaling, rotation, and tilting. This means that the camera picture must be

subjected to an inverse transformation as preprocessing to reading the watermark in order to return it to the same geo-

metric form as the original picture. For scaling and rotation, there are techniques for performing an inverse transforma-

tion by inserting frequency domain templates[11]. An orthogonality transformation (a transformation to the frequency

domain) does not, however, preserve the tilting transformation component, which prevents these techniques from

being used. To achieve a tilting transformation, projective transformation parameters between the original picture and camera picture can be estimated, and an inverse projective transfor-

mation can be performed based on those parameters.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MUM2004 October 27–29, 2004, College Park, Maryland, USA.

Copyright 2004 ACM 1-58113-981-0/04/10 ...\$5.00.

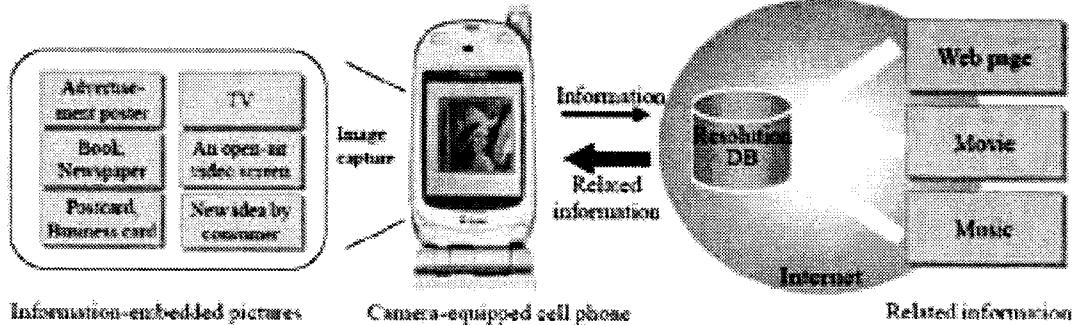


Figure 1: Image of services

Here, scaling and rotation transformations are automatically included in a projective transformation, which means that there is no need to perform them separately. Once a geometric form that is the same as the original picture is obtained by the inverse projective transformation, the digital watermark is read from that picture. Fig.2 shows this process flow.

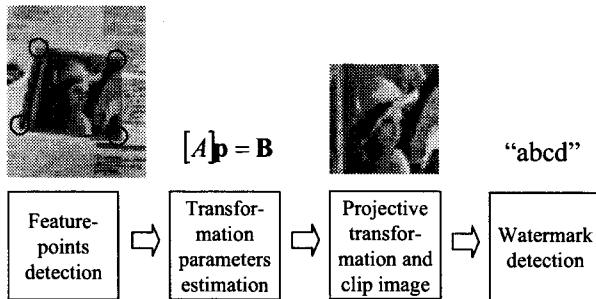


Figure 2: Operation flow

To estimate projective transformation parameters, multiple feature-point coordinates between the original picture and the camera picture must be obtained[4], and to do so using i-appli where computational power is limited, an algorithm that can perform an efficient search for feature points in a picture is indispensable. In this report, we use the corners of the target picture as feature points and describe the Side Trace Algorithm (STA), a corner-search algorithm that can perform real-time processing using i-appli, and describe the results of its implementation.

2. REQUIREMENTS AND CHALLENGE

2.1 Hardware Conditions

In the following study, the cellular phone incorporates a 32-bit processor with a maximum operating frequency of approximately 100 MHz, and i-appli runs on a Java Virtual Machine on that processor. The i-appli operating environment has the following features:

- Supports only integer arithmetic.
- Processing speed and memory access are slow (Table1)

- Memory capacity is small storing no more than two 512x512 integer arrays (depend on a terminal).
- Application program interface (API) for transferring data from the camera frame buffer area to memory is limited in size requiring divided processing for pictures in excess of 288x352(depend on a terminal) pixels.

Table 1: Cellular Phone Benchmark

Terminal	A	B	C
Access one element of 2D int array	227	223	783
int ADD	31	34	85
int MUL	55	57	135
int DIV	805	753	937
int SHIFT	43	44	63
Comparison conditional branch	50	49	87
(nsec.)			

From Table1, a considerable character of cellular phone, the memory access is slower than calculation. That character is more remarkable than the ordinary computer system. The ratio of computational cost among *MA* the memory access, *ADD* the add/subtract operation, *MUL* the multiplication operation, *DIV* the division operation, and *CB* the comparison conditional branch is following:

$$MA : ADD : MUL : DIV : CB = 7 : 1 : 2 : 25 : 2 \quad (1)$$

While the number of pixels comprising a camera frame depends on the type of terminal, 288x352 and 480x640 are typical pixel configurations. In this study, the former configuration becomes the main target considering the above API and memory size limitations.

2.2 Response Time

Past studies derived several indexes for the maximum time that users will tolerate from taking a picture to obtaining digital watermark data, that is, from performing an input operation to obtaining a result [6, 7]. All of these studies, however, agree that a time interval within two seconds is acceptable to users, and two seconds therefore becomes the upper limit here. Because the maximum time required for reading a digital watermark is 0.4 seconds using i-appli [5],

the processing time allowed for feature-point searching and inverse projective transformation is $2 - 0.4 = 1.6$ seconds.

2.3 Noise

In printed matter, noise in the form of characters and other pictures often exists along the periphery of a picture having an embedded digital watermark. A feature-point search must be able to withstand the effects of this noise, which may even include other pictures that themselves embed digital watermarks. Here, the possibility of capturing two or more complete pictures with digital watermarks within a single frame may be excluded considering that users would tend to position the target picture at the center of the camera frame obtaining a sizeable amount of that picture. What should not be excluded, however, is the possibility of capturing a part of another picture that has an embedded digital watermark.

2.4 Challenge

Based on the above requirements, the problem that needs to be solved here is to find feature points within a camera picture within a limited number of computations in integer-arithmetic processing. There has been considerable research[8, 9] to date on the problem of finding lines or patterns in a picture and search accuracy has improved. The reported techniques do not, however, envision processing on the i-appli operating environment characterized by slow processing speed and support of only integer arithmetic. Consequently, they cannot be applied directly without creating problems. Table 2 shows processing times using i-appli when searching for the periphery of a picture using the high-speed Hough transform for line searching[8], and using the Sequential Similarity Detection Algorithm (SSDA) [9], a high-speed pattern-matching process. In either case, processing time is more than 10 times that of the required 1.6 seconds. Olmo's Integer Hough transform [10] will gain 3 times speed of the ordinary Hough transform, but it is still shorter.

Table 2: Traditional Algorithm Processing Time

Terminal	A	B	C	P4 1.5GHz PC
Line search by the Hough transform	27	24	28	0.55
Pattern Matching by SSDA	740	1215	1602	0.01 (sec.)

Let us examine the reasons for this. It is that round-robin processing within a whole image. It is disadvantageous, if the memory access is slow. This is proved by Table 2 that shows the Hough transform is faster than SSDA on i-appli environment, reversely the Hough transform is slower than SSDA on the PC. This is because a speed of the Hough transform which treats only black pixels after binarization of a whole image depends on the arithmetic operation cost higher than the memory access cost. For our purposes in which finding a finite number of feature points is considered sufficient, all pixels of the image do not need to be processed. This leaves open the possibility of high-speed processing and is the focus of this paper.

3. BASIS OF CORNER SEARCH ALGORITHM

3.1 Basic Algorithm

We focus on the fact that the user will try to place a quadrilateral at the center of the frame when taking a picture with his camera-equipped cellular phone. Now, if we consider a crosswise line that bisects the camera frame dividing it into upper and lower sections, that line will cross the left and right sides of that quadrilateral (Fig.3) meaning that those sides can be found by searching only along that line. The STA makes use of this property and detects the corners using the following search procedure, which is also shown in Fig.4.



Figure 3: Bisector line

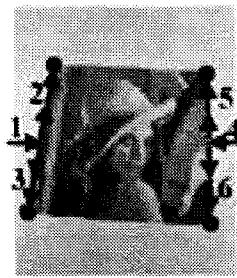


Figure 4: Side Trace

Step1 Starting at the midpoint of the left edge of the camera frame, advance to the right searching for the left side of the quadrilateral.

Step2 On finding the left side, trace the side upward from that position and consider the point where the left side breaks off as the upper left-hand corner of the quadrilateral. When tracing the left side, the possibility of a slanted side must be considered, and for this reason, one pixel to the left and right of the current side position will also be checked as opposed to checking only the pixel directly above the current position.

Step3 Starting again from the position where the left side of the quadrilateral was originally found, trace the side downward using the same method as that in Step 2 and consider the point where the side breaks off as the lower left-hand corner of the quadrilateral.

Steps4,5,6 Starting at the midpoint of the right end of the camera frame and advancing this time to the left, perform the same process as that in Steps 1, 2, and 3 to obtain the upper right-hand and lower right-hand corners.

Step7 Exit procedure.

We note here that this algorithm does not take into account the case where the quadrilateral is not crossed by the line bisecting the camera frame, that is, where the quadrilateral is small and situated in either the upper or lower section of the camera frame. There is really no need to deal with this case, however, since reading the digital watermark from a picture in this condition is bound to fail. In the digital watermark system by Nakamura[5], watermark information is embedded in the picture by making fine adjustments to the brightness of each pixels of the original picture within a

range unnoticeable to the human eye. If a picture taken by a camera is too small, however, multiple pixels each modulated differently in terms of brightness will degenerate into a single pixel. This makes it difficult to restore the brightness-modulation components and increases the possibility of failure in digital watermark reading. Accordingly, if the bisecting line does not cross the quadrilateral, it would not be inappropriate to declare a reading error at that time and to simply terminate processing. This termination, moreover, corresponds to a prediction as to the possibility of a reading failure and does not constitute a trial reading of the digital watermark. This, in essence, improves response time at the time of an error, which we take to be a convenient outcome.

3.2 Picture Frame and Frame Search

The above technique of following the sides of a quadrilateral to find corners does not work for pictures having no background or picture divider coinciding with the sides. It can be made to work on any picture, however, by adding a picture frame along the outside of the picture and following that frame instead of the sides. The frame can be also a sign that indicates a watermark is in the picture. Fig.5 shows the frame design. Frame conditions are given below.

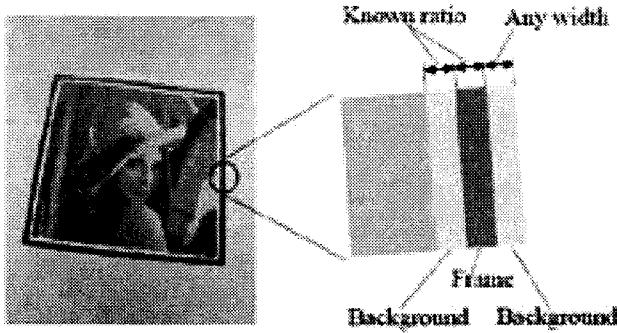


Figure 5: Frame details

- a A belt-shaped background section must be placed on both sides of the frame. This is necessary to enable frame recognition if the color of the frame is the same as that of the target picture or of the paper that the picture is printed on.
- b The frame width and the width of the outer background section are arbitrary.
- c The ratio between the width of the inner background section and the frame width must be set to enable the distance from the frame to the picture to be determined on the basis of frame width when extracting a picture.
- d The background-section color is arbitrary. It may even be the same as the color of the printed paper.
- e The frame color is arbitrary as long as the frame intensity is made sufficiently low compared to the background intensity so that a difference in intensity between the two can be obtained with the camera in question.

The following function is used to determine whether a certain point in the camera frame is a part of the picture frame. Given intensity matrix I consisting of $3 \times n$ pixels centered about the examined point (where n is the value two pixels greater than the pixel-width of the frame), perform a convolution operation using frame-detection filter operator matrix F also consisting of $3 \times n$ pixels. One example of how a filter operator matrix F can be configured is to give matrix elements along both sides of the matrix a value of $n - 2$ and all other elements a value of -2 . If the result is larger than the frame threshold value th , the point is judged to be part of the picture frame (eqn 2).

$$FrameValue = \sum_{i=1}^3 \sum_{j=1}^n I_{ij} F_{ij} \quad (2)$$

Frame : if $FrameValue \geq th$
Not Frame : if $FrameValue < th$

3.3 Computational Cost

Let us now estimate the computational cost of this basic algorithm. Let MAs denote the number of times the image-memory is accessed, $ADDs$ the number of add/subtract operations, $MULs$ the number of multiplication operations, and CBs the number of comparisons of the conditional branches. Let the distance from the left end of the camera frame to the picture frame be m pixels, the length of the picture be l pixels, and for the sake of simplicity, we assume the same for the right half. In eqn(3), p is a number of pixels to search. In expression of p , the factor 3 of l comes from adjacent pixels search for a slanted side (section 3.1 Step 2).

$$\begin{aligned} MAs &= 3np \\ ADDs &= 6(n-1)p \\ MULs &= 6p \\ CBs &= 3np \\ p &= 2(m+3l) \end{aligned} \quad (3)$$

In i-appli, multiplication, addition, and shift operations all have the same order cost as shown in Table 1. Thus, while multiplication is not expanded into addition and shift operations in eqn (2), and since it is easy to convert multiplication to addition and shift operations in the convolution-operation section, all multiplication may also be processed by arithmetic and shift operations in a processing system having a high multiplication cost. Now, if we substitute in the MAs expression of eqns (3) the values obtained for m , l , and n actually obtained from the Fig.6 to be performed a frame search using this basic algorithm, we obtain the following.

$$MAs = 3 \cdot 6 \cdot 2(13 + 250 \cdot 3) = 27468 \quad (4)$$

This number is approximately 30% of the total number of pixels in the camera frame, which means that the picture frame could be detected without having to access all pixels in accordance with our original objectives. Additional advantage of this algorithm is that it can process within the local image area. When searching an upper left corner, there is no need to consider other corners area. In actuality our system divides the image into four pieces, and search a corner separately. This saves the memory, therefore it is suitable for cellular phones.

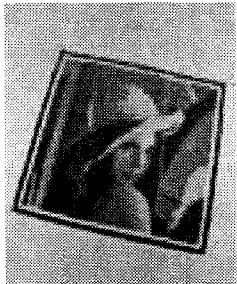


Figure 6: Original

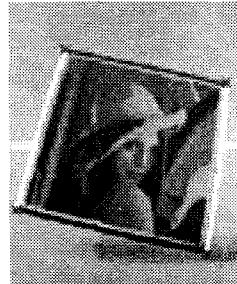


Figure 7: Image After locating sides(white dot) and corners(cross)

3.4 Frame Width and Threshold Estimation

The proposed algorithm uses a $3 \times n$ filter operator for picture frame detection. Here, the value of n depends on the frame width, and since it cannot be known beforehand, it must be estimated in order to configure the filter operator. This estimation is performed by differentiating pixel intensity in the crosswise direction with respect to the frame, and treating the difference between the maximum-differential and minimum-differential positions as the frame width. The frame width of the camera picture increases or decreases according to the frame width printed on the original picture, the number of camera pixels, and the shooting distance, and may even increase or decrease within the same camera picture depending on the camera tilt. The frame width estimation is therefore performed periodically when tracing a picture frame.

The frame intensity and background intensity of the camera picture varies according to the photographing environment and lens performance. The threshold value th used in frame searching should therefore be adjusted dynamically. The value th can be expressed as a first-order equation as in eqn (5) in terms of background intensity I_b and frame intensity I_f . In this equation α , β , and γ are coefficients that are dependent on the configuration of the filter operator.

$$th = \alpha I_b + \beta I_f + \gamma \quad (5)$$

If the frame width is known, it becomes relatively easy to determine the positions for sampling the background intensity and frame intensity, which means that intensity sampling and calculation of the threshold value are carried out following the frame width estimation discussed above using the value so obtained. The additional amount of processing performed due to the frame-width and threshold estimation constitutes less than 1% of all frame-search processing and can therefore be ignored in the following discussions regarding processing time.

4. ACHIEVING EVEN FASTER SPEED

4.1 Hopping Search (Coarse Search)

There is room for making the basic algorithm presented in Section 3 even faster. For example, when tracing the picture frame, memory access can be decreased by hopping h pixels at a time (coarse search) instead of tracing one pixel at a time. With this approach, however, the point where the frame breaks off may be in error by a maximum of h pixels.

To recover from this error, tracing retreats a full h pixels from where the frame was first found to break off and then restarts advancing one pixel at a time.

4.2 Direction Limitation

Direction limitation may be introduced into the tracing process along the picture frame. Specifically, the frame direction determined when beginning frame tracing can be recorded and tracing in any other direction can be excluded. Until this direction is determined, however, searching to the left and right is performed within a range of $\pm t$ pixels while hopping vertically h pixels at a time. The value of t can be given by $\tan \theta = t/h$ where θ is the allowable tilt of the frame. When the direction to take is determined, the left/right search range is reduced to ± 1 pixel. The procedure for direction limitation is as follows. The variable f denotes the number of hops before determining the direction.

- Step1** Record the x coordinate of the frame-tracing start point.
- Step2** Follow the frame advancing in the vertical direction h pixels at a time until f hops are made. For every move of h pixels, the horizontal range for frame searching is $\pm t$ pixels.
- Step3** Divide the difference between the current x coordinate and the x coordinate of the frame-tracing start point by f to obtain the *grade*, and record that value as the determined direction.
- Step4** Follow the frame while advancing vertically h pixels at a time until loosing track of the frame. For every move of h pixels, the horizontal range for frame searching is now $+grade \pm 1$ pixel.
- Step5** Move backwards vertically h pixels from the position where the frame was lost.
- Step6** Start following the frame again in the original direction while advancing vertically 1 pixel at a time until the frame is again lost. The horizontal range for frame searching at this time is ± 1 pixel.
- Step7** Take the position where the frame is lost to be a corner and exit the procedure.

Found corners using this improved algorithm is the same as the image in Fig.7.

Computational costs MAs' , $Adds'$, $MULs'$, and CBS' of this algorithm can be estimated as follows where symbols m , l , and n have the same meanings as given in section 3.3.

$$\begin{aligned} MAs' &= 3np' \\ Adds' &= 6(n-1)p' \\ MULs' &= 6p' \\ CBS' &= 3np' \\ p' &= 2(m+f(2t+1)+3(l/h-f)+3h) \end{aligned} \quad (6)$$

For the purpose of comparison, Table3 gives the results of substituting $t = 6$, $h = 10$, and $f = 5$ and values for m , l , and n obtained by actual measurements from Fig.7 in eqns (3) and (6) with normalizing using ADD cost = 1. The results reveal that improved algorithm reduces the computational cost to less than 1/3 that of the basic algorithm.

Table 3: Computational costs

	Basic	Improved
Computational cost normalized by ADD	1	0.30

The actual number of memory access of the improved algorithm, MAs' is following:

$$MAs' = 3 \cdot 12(13 + 5(12 + 1) + 3(25 - 5) + 30) = 6048 \quad (7)$$

MAs' is less than 10% of the total number of pixels of the camera picture.

5. ANTINOISE

Noise in the form of characters or other lines in the vicinity of the frame can be erroneously identified as part of the frame. One way to prevent this from happening is to perform labeling on the discovered line. In labeling processing, the longest set of consecutive pixels is usually taken to be the real frame. However, the computational cost of labeling processing is high; therefore, it cannot be adopted for our purposes. We therefore introduce the following two heuristic algorithms those make use of noise features to eliminate peripheral noise efficiently.

- a Making use of the fact that a line segment in a character cannot be very long, a line for which frame tracing fails under a certain threshold length is not considered to be a true frame. This algorithm is not, however, compatible with the coarse search described in section 4.1. To rectify this, the initial stage of the coarse search is modified to advance one pixel at a time with the result that a coarse search and character-segment elimination are both achieved.
- b After reaching a corner and attempting to continue frame tracing at a right angle to the line in question toward the inside of the picture, the line is not considered to be part of the correct frame if frame tracing cannot continue. This algorithm is a countermeasure to the case in which an adjacent picture with a picture frame attached is captured together with the intended picture as in the image in Fig. 8. An adjacent picture frame can be distinguished since it turns at the corner in a direction opposite that of the correct frame.

Fig.9 shows the results of frame detection using anti-noise heuristic algorithms. Applying noise countermeasures increases the computational cost and memory access in a manner proportional to the number of pixels in the noise elements crossing the line bisecting the camera frame, i.e., the search path. Denoting the total number of pixels in a character crossing the search path as n_c , the total number of pixels in another line crossing the search path as n_l , and the total number of pixels in the correct frame as sl , the rate of increase in both computational cost and memory access relative to the case with no noise can be given as below.

$$\frac{n_c + n_l}{sl} \quad (8)$$

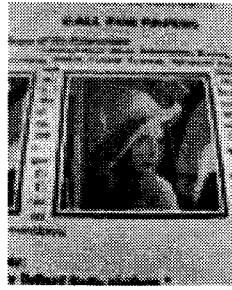


Figure 8: Original



Figure 9: Image After locating sides (white dot) and corners (cross)

For the case with no noise, the numerator in eqn (8) is zero and anti-noise overhead is zero as well. Furthermore, as noise not on the search path does not affect the amount of required processing, characters that can affect the processing speed are extremely few even in a case such as that in Fig.8 where a picture is set among characters. In Fig.9, there is only one fake line and one character on the search path. In lower center image in Fig.10, there are three characters. We found that those are not special cases empirically.

6. PROJECTIVE TRANSFORMATION

6.1 Parameters Estimation

We estimate projective transformation parameters from four pairs of corner coordinates (x, y) obtained by the process presented above and four pairs of corner coordinates (x', y') from the original picture. These eight parameters constitute constant terms in eqns (9), the projective transformation equation.

$$\begin{aligned} x' &= \frac{a_1x + b_1y + c_1}{a_0x + b_0y + 1} \\ y' &= \frac{a_2x + b_2y + c_2}{a_0x + b_0y + 1} \\ (x', y') &: \text{original picture position} \\ (x, y) &: \text{camera picture position} \end{aligned} \quad (9)$$

Now, if we substitute the four sets of corresponding coordinate pairs $(x, y), (x', y')$ into eqns (9) rewritten as eqns (10), we obtain eight simultaneous first-order equations, and solving them will give us the eight parameters.

$$\begin{cases} xx'a_0 - xa_1 + 0a_2 \\ +yx'b_0 - yb_1 + 0b_2 - c_1 + 0c_2 = -x' \\ xy'a_0 - 0a_1 - xa_2 \\ +yy'b_0 - 0b_1 - yb_2 - 0c_1 + c_2 = -y' \end{cases} \quad (10)$$

Because i-appli is incapable of decimal-point arithmetic, the approach adopted to solve these equations is to multiply them by an appropriate constant K and apply the Gauss-Jordan method as follows. First, assuming 160x160 pixels for the original picture and 288x352 pixels for camera resolution, the equation-matrix coefficients range from a maximum absolute value of $yy' = 352 \cdot 160 = 56320 \cong o(10^4)$ to a minimum absolute value of one. Treating this maximum absolute value as a pivot and dividing by the other coefficients

and multiplying by constant K must satisfy $K \geq o(10^4)$ to ensure that the minimum absolute value of one does not underflow. In addition, to prevent 32-bit integer overflow when multiplying large-valued coefficients by each other during forward elimination, it must satisfy $o(10^4) \cdot K < 2^{31} \cong o(10^9)$. The final condition can therefore be expressed as:

$$o(10^4) \leq K < o(10^5) \quad (11)$$

From this, we consider the following value for K to be appropriate.

$$K = 2^{16} \simeq o(10^4) \quad (12)$$

The time required for solving the eight simultaneous first-order equations after multiplying by K at terminal A is under 1 msecound, a sufficiently small value.

6.2 Transformation

The inverse projective transformation is performed using eqns (9). The coordinate values resulting from the transformation are not integers, however, and an appropriate integer-approximation method is therefore needed. Three candidates for this role are the nearest neighbor method, the bilinear method, and the bicubic method. From left to right, processing time and approximation accuracy of these methods increase for each method. Table4 shows processing-time results after implementing each of these methods on Terminal A. Since the problem of whether the digital watermark can be correctly read is of more concern than how natural the resulting picture appears, approximation accuracy is evaluated here on the basis of watermark-reading performance. It was found that the bilinear method was optimal taking a balance between processing time and approximation accuracy into account.

Table 4: Processing time and success rate of watermark detection for each interpolation method

Method	Nearest	Bilinear	Bicubic
Processing time on Terminal A (sec.)	0.20	0.55	2.16
Success rate of watermark detection	85%	90%	90%

7. EXPERIMENTS AND RESULTS

An experiment is performed from the two perspectives of corner-search accuracy and processing time. A total of over 1000 pictures are used for evaluation purposes. These pictures are taken by more than one person using a cellular phone (Terminal A) under fluorescent lamps and sunlight to provide two types of lighting conditions. Size of all pictures are 288x352 pixels. Captured frame widths are 3~5 pixels. Fig.10 shows some of the pictures used in the experiment and results. The inner black and white blocks are only used to obtain a baseline to determine the accuracy of the detected corners. In evaluating the corner-search accuracy, corner coordinates detected within an error of 2.3% of picture width are treated as correct. A number of 2.3% comes from Fig.11 of the pre-experiment of a watermark detection reliability. The Fig.11 shows a relation between artificial added corner position errors normalized by a picture width and watermark

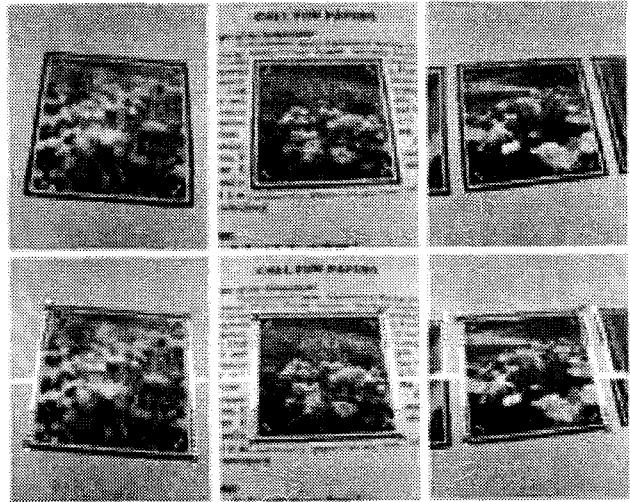


Figure 10: Sample images and resulting images

detection reliability index[5] values. Nakamura[5] says the minimum value of the reliability index to detect watermark correctly is 6.2. The corresponding corner position error value to reliability index= 6.2 is 2.3%.

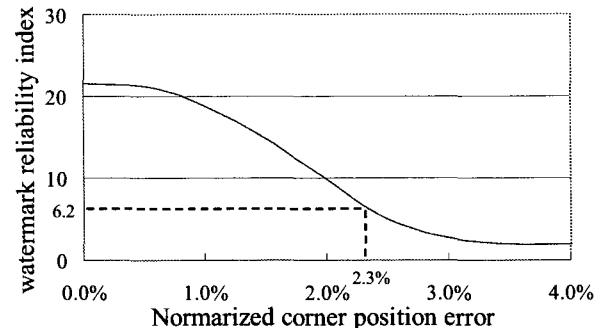


Figure 11: relation between normalized corner position errors and watermark detection reliability indexes

Processing time is determined by measuring the time taken to find all four corners. Table 5 and Table 6 give the experiment results.

From Table5, we see that sufficient frame-search accuracy is obtained even for pictures including noise. From Table6, we see that a processing time of only 0.1 sec can be achieved even when including anti-noise processing on Terminal A. Add to this the 0.55 sec (Bilinear time of Table4) for inverse projective transformation and approximation, the resulting processing time of 0.65 sec is still sufficiently below our upper limit of 1.6 sec.

From examination of pictures which corner searches had failed, reasons of fail are poor lighting and too small size of pictures. It is expected that it is difficult to detect watermarks from those pictures. To verify this, we performed corner searches with the naked eye on pictures for which corner searches had failed, and then examined the success rate of reading digital watermarks from those pictures after subjecting them to inverse projective transformations based on those manually determined corners. We find that digital-

Table 5: Accuracy of corner detection

Target image	Alone without noise	With character noise	With another picture	All
Total	320	376	364	1060
Good corner detection	271	291	264	826
Success rate	85%	77%	73%	78%

Table 6: Processing time for each method

Terminal	A	B	C	Pentium4 1.5GHz PC
Improved algorithm with anti-noise (sec.)	0.07	0.06	0.16	0.002
Line search by Hough transform (sec.)	27	24	28	0.55
Speed up ratio	386	400	175	275

watermark detection will certainly fail for 70% pictures in which corner search fails, and that the corner-search performance is enough to read digital watermarks.

8. CONCLUSION

We presented the performance features of the i-appli processing system for cellular phones and proposed a frame-corner search algorithm designed specifically for i-appli. To achieve high-speed processing here, we made use of the fact that a line bisecting the camera frame into upper and lower sections crosses the left and right sides of the quadrilateral outlining the target picture thereby limiting the search area in a heuristic manner. This scheme successfully reduced the number of accessed pixels for searching to less than 10% of the total number of pixels in the camera frame and achieved significantly higher processing speeds. To deal with noise in the form of peripheral characters or other quadrilaterally shaped pictures that hinder searching, we performed high-speed pruning by assessing the length and direction of consecutive pixels. This achieved a satisfactory level of noise resistance requiring just a small increase in processing time proportional to the amount of noise present. We also performed an evaluation experiment using actual handsets incorporating the proposed algorithm and found that processing speeds more than 100 times faster than general image-processing algorithms were obtained (Table6). The same experiment revealed that a corner-search success rate of at least 73% could be obtained and that a picture that failed a corner search also tended to fail digital-watermark reading. This finding confirms that sufficient search accuracy can be obtained for reading digital watermarks.

9. ACKNOWLEDGEMENTS

The authors extend their deep appreciation to Dr. Susumu Ichinose, former Director of NTT Cyber Solutions Laboratories, for the opportunity to perform this study; to Ms. Yoko Takahashi of NTT Cyber Space Laboratories for very

helpful information on the usefulness of a frame-search operator; and to Mr. Yoshinori Kusachi of NTT Cyber Space Laboratories for his valuable ideas on frame tracking.

10. REFERENCES

- [1] Toshiki Iso, Shoji Kurakake, Toshiaki Sugimura, "Visual-Tag Reader: Image Capture by Cell Phone Camera," , 2003 International Conference on Image Processing(ICIP2003), ProceedingsIII pp.557-560, Sep. 2003.
- [2] A resolution service for cellular phones, <http://p-warp.jp/index.html>
- [3] Specifications of Java for i-mode, <http://www.nttdocomo.co.jp/english/p.s/i/index.html>
- [4] K.Kanatani,"Geometric Computation for Machine Vision," The Oxford Engineering Science, No 37, 1993.
- [5] Takao Nakamura, Atsushi Katayama, Masashi Yamamoto, Noboru Sonehara,"Fast Watermark Detection Scheme for Camera-equipped Cellular Phone, " Mobile and Ubiquitous Multimedia (MUM2004), Oct, 2004.
- [6] Miller, R. B. , "Response time in man-computer conversational transactions," Proc. AFIPS Fall Joint Computer Conference Vol. 33, 267-277, 1968
- [7] Ben Shneiderman, "Designing the user interface 3rd edition," ISBN:0-201-69497-2, Addison-Wesley Longman Publishing Co. Inc., 1998.
- [8] L.Xu, E.Oja, P.Kultanen, "A new curve detection method: randomized Hough transform(RHT)," Pattern Recognition Letters 11(5) pp.331-338, 1990.
- [9] D.I. Barnea and H.F. Silverman, "A Class of Algorithms for Fast Digital Image Registration," IEEE Trans. On Computers, Vol C-21, No. 2. Feb. 1972
- [10] Gabriella Olmo and Enrico Magli, "All-integer hough transform: Performance evaluation," ICIP01, III:pp.338-341, 2001
- [11] Takao Nakamura, Hiroshi Ogawa, Atsushi Tomioka, Youichi Takashima, "Improved Digital Watermark Robustness against Translation and/or Cropping of an Image Area," IEICE TRANS. FUNDAMENTALS, Vol.E83-A, No.1, Jan, 2000.

Automatic Video Production of Lectures Using an Intelligent and Aware Environment

Michael Bianchi
FOVEAL SYSTEMS, LLC
190 Loantaka Way
Madison, New Jersey 07940-1910
MBianchi@Foveal.com

ABSTRACT

This paper makes the case that much of the promise of ubiquitous multimedia depends on the availability of valued material. In business and academic environments the “presentation in a lecture room with laptop graphics” is a common way of communicating, but making a presentation readily available outside of the room is still a challenge because of the complexities of capturing and distributing the material. The AutoAuditorium™¹ System creates a multi-camera video program of a lecture in real time, without any human control beyond turning the system on and off. The system reduces the opportunity costs of making such a program to the point that it gets used for events previously not seen as candidates for video. Thus an event does not need nearly as many viewers to be considered worth capturing and many more events are seen by many more people.

This paper presents a quick overview of the AutoAuditorium System technology and operational characteristics, a history of its ancestry, development and use, and some usage experiences that demonstrate its current utility and future potential.

The AutoAuditorium System is an example of an “intelligent and aware environment.” In particular, it is:

- | | |
|-------------|--|
| intelligent | – about creating multi-camera television programs of lectures, in real time, with one or more people on a stage using projected visuals. |
| aware | – of the motion and gesturing of the people on stage.
– of changes in the projected visuals. |

Originally created in the early- and mid-1990s as a research project [1] at Bellcore (Bell Communications Research, now Telcordia Technologies), it has been available as a commercial product from Foveal Systems since 1999.

¹ AutoAuditorium and AutoAud are trademarks of Telcordia Technologies Inc., used under license.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MUM2004 October 27-29, 2004 College Park, Maryland, USA
Copyright 2004 ACM 1-58113- 981-0 /04/10 ...\$5.00.

Categories and Subject Descriptors

H.4.2 [Information Systems Applications]: Communications Applications —Videoconferencing; I.4.9 [Image Processing and Computer Vision]: Applications; K.3.1 [Computer Uses in Education]: Distance Learning

Keywords

automatic video production, multimedia content creation, multimedia production

1. OVERVIEW

The AutoAuditorium System [2] adds to an ordinary auditorium, lecture hall or classroom the ability to automatically make video broadcasts and recordings of lectures and talks. Permanently installed in the room, it uses cameras and microphones to be “aware” of what is happening on stage. It televises, in real time, the most common auditorium, lecture hall or conference room talk: people speaking, showing projected visuals to a local audience. The intent is to allow other audiences to see and hear the event as a television program in another place or at another time. Three subsystems create the TV program:

- An automatic Tracking Camera tracks the person or people on stage.
- The automatic Director selects camera shots based on what is happening in the program.
- The automatic Audio Mixing combines stage and audience sound to create a complete sound track, including questions and answers.

Because the system is built into the room the people on stage and in the local audience are not distracted.

Because it is produced using multiple cameras and appropriate video effects (e.g. picture-in-picture), an AutoAuditorium program is often indistinguishable from one produced by a crew.

Because it is completely automatic, it is easy and economical to use, and therefore it is used often. Thus many more people can avoid traveling, or missing talks because of a schedule conflicts. Classes, talks and seminars occurring in one location can be telecast as video programs to other locations and/or recorded for later use. Now the everyday events of business and education can become the feed stock for “ubiquitous multimedia”.

2. THE HISTORY OF THE AUTOAUDITORIUM SYSTEM

Reducing Manual Production to One Person. Before the mid-1970s, television technology was only usable by organizations willing to finance a professional operation. The most common of those were television broadcast stations and companies that produced programs for sale to TV stations.

The advent of video tape technology arose from the economies of scale that the transistor and integrated circuit revolutions brought to electronics. Complex functionality was now encapsulated to the point that in-depth engineering knowledge was no longer necessary to use the equipment and create programs. The video cassette meant that virtually anyone could play TV programs when needed and corporations started using video for programs previously presented on 16 millimeter film. The prices were then affordable in many industrial settings.

The home video cassette accelerated the economies of scale and the attendant reduction in prices. By the late 1980s, it became clear that video technology was moving from the “industrial age” into the “consumer age” and technology created for the home market made industrial equipment much more reliable and affordable.

In this environment, the demand for video recordings of corporate events and functions grew rapidly and the VHS home video format became ubiquitous. The assumption was that everyone had access to a VHS video cassette player, and in technology companies many types of educational and corporate communications were distributed as “video”. The “VHS” was understood.

But many of the programs created in the corporate world made poor use of the training and skill of professionals. When the material being presented was little more than a lecture with projected visual aids in the form of 35mm slides, letter-size foils, or (eventually) computer-generated stills, the programs demanded very little of the production crew:

- Follow the person speaking on stage.
- Switch to the projection screen when appropriate.
- Switch back to the person.

Many organizations tried to “roll their own” video programs using just a camcorder, but the results were usually disappointing, tedious to watch, and/or unintelligible. So while many families made home videos using a camcorder, creating the material in corporations remained a “professional” activity, requiring several trained people and professional equipment unavailable to the majority of home videographers.

In the late 1980s, a project called the Distributed Auditorium System at Bellcore reduced the crew requirement for the basic lecture program to one person. The key insights were that:

- TV cameras and related equipment were now inexpensive enough to install permanently in a room.
- TV equipment was acquiring useful automatic features, such as autofocus and automatic white balancing.
- a single, trained person could usually keep up with the requirements of a simple programs.

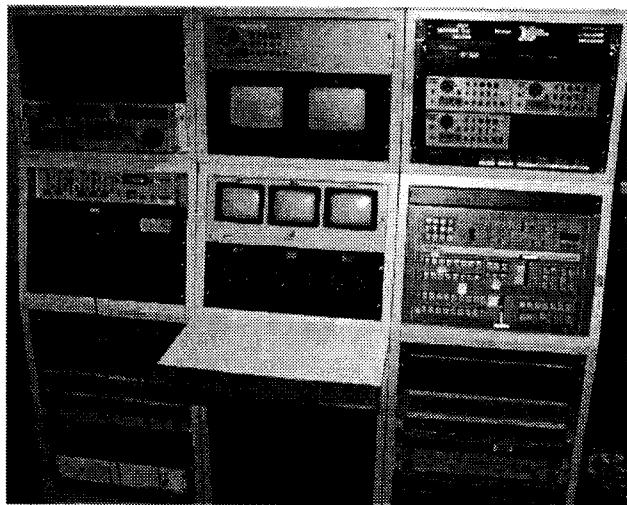


Figure 1: The Distributed Auditorium Console. Notice the three camera monitors, each above a cluster of controls for that camera.

Although the standard equipment was organized around a one-person-for-each-job model, it was possible to redesign the workspace, using human factor design principles, so most of the controls were usable without looking at them. The operator knew which control they were handling by its shape and direction of action.²

A goal of the project was to create a means of recording and telecasting talks that had minimal impact on the people giving the talks and the people watching them. We did not want an environment that felt like or operated like a television studio. Instead we wanted to create a space that looked and felt like an ordinary meeting room and where the technology was unobtrusive (if not invisible) to all involved. So the cameras were hung from the ceiling, added lighting was made as inconspicuous as possible, microphones were also hung from the ceiling, and all evidence of television production was put in a back-room.

Figure 1 shows the operator’s console of an original Distributed Auditorium System. The central control panel (two color monitors, three camera monitors and consolidated camera and video mixer controls) enable one person to position, adjust and select the three remotely controlled cameras. Only the most useful controls from the surrounding equipment were duplicated in each camera’s control cluster.³

Two cameras are hung on the ceiling in the center of the room, pointed at the stage. The first is used primarily for following the person speaking on stage. The second is used primarily to capture the images on the projection screen. The third camera was hung above the corner of the stage opposite the lectern. It was primarily used to look across the stage, taking a shot of the person speaking from the side. It could also look at people in the audience, especially when

²For example, the Focus control moved left-and-right while the Zoom control moved up-and-down.

³For example, each camera cluster has one button to put that camera’s image into the program. Those three buttons are duplicates of ones found among the 75 on the video mixer immediately to the right.

someone was asking a question.

In the Distributed Auditorium environments the operators were seldom people with television production experience. But a well written training manual, some time to practice, and constructive criticism quickly resulted in programs that were worth watching if you were interested in the topic. These were not award winning productions, but to someone faced with either the time and expense of traveling to the event or missing the presentation altogether they were far better than nothing.

Four Distributed Auditorium Systems provided video tapes and live inter-location telecasts for over a decade at Bellcore. The majority of talks important enough to attract a sizable audience were scheduled in the DistAud auditoriums and people's expectations became that they would be televised and taped. Many video cassette recordings were cataloged into the company library.

Genesis of the AutoAuditorium System. The existence and successful use of the Distributed Auditorium Systems as corporate resources motivated the projects which ultimately became the AutoAuditorium System. Although they grew incrementally, the goals of the new projects grew out of the goals successfully implemented in the older one. We wanted the system to be unobtrusive, reliable, valuable, easy to use and frequently used.

Because the operators of the DistAud Systems had other responsibilities, it was not possible to schedule them for every talk. Those missed opportunities motivated experiments in partially automating, and then fully automating, the video production of lectures.

Automatic Audio Mixing. The first practical automation of the Distributed Auditorium System involved automatically mixing the audio from the stage with the audio from the audience, so the remote viewers could hear both. Experience quickly taught us that no matter what we did to encourage people in the audience to use a specific "question microphone" they would simply shout out their questions or comments during talks. The size of the rooms and the seating plans simply made informal exchanges too easy. We could not enforce any other discipline. The Distributed Auditorium's control console was given an easy-to-use audio mixer so the operator could quickly turn up the microphones hung from the ceiling above the audience, but that often proved unsatisfactory.

After much experimentation and a few false starts, an automatic audio mixer with input priorities was created. The wireless microphones, if used, were given priority over microphones above the stage, which were given priority over the mics above the audience. The result worked well. Polite give-and-take between the primary speaker (wearing the wireless mic), other people who came on stage and the audience resulted in good coverage. If more than one person spoke at once, the priorities decided which person would actually be heard clearly by the remote viewers. The automatic audio mixing tended to work without attention. It also behaved well when the wireless microphone was forgotten, accidentally turned off or suffered a dead battery. It would revert to the microphones mounted over the stage.

The features that make priority audio mixing practical are now common in modern audio mixers, but were not found in the early 1990s.

The automatic audio mixing was often acceptable without any adjustment. In fact, when we could not schedule an operator, we would sometimes set the video mixer on a single picture-in-picture image, combining the projection screen with an inserted image of the lectern, and record the talk with just that image and the automatic audio mix. Since the programs were not intended as entertainment, the audience was often tolerant of less-than-ideal "production values" if they could see most of what they wanted to see and hear most of what they wanted to hear. "*Anything* is better than nothing!"

This success made adding further automation to the system seem practical.

Automatic Camera Tracking of the People on Stage. A common problem in Distributed Auditorium System programs was that the operator could be distracted by motion. A person on stage who paced back and forth would prompt the operator to track them with one of the remote control cameras. This required constant attention, and so the operator sometimes would miss the fact that the *interesting* image was not the person but rather what was on the projection screen.

This is exactly the reason that real television production is done with a person behind every distinct job. When things get busy, each camera operator manages a camera, the person running the video mixer has both hands full, the audio mixing keeps another person busy, while the director does nothing but watch all the monitors and call out instructions to the crew.

Even at a well designed console, having one person trying to serve all those functions inevitably resulted in occasional cognitive overload or running out of hands.

This need motivated a Bellcore research project in motion tracking. The result was an automatic Tracking Camera called the ICU, Intelligent Camera Unit,⁴ that was eventually added to one of the Distributed Auditorium Systems. The goal was to give the DistAud System operator an "assistant" which would always follow the person on stage.

In 1994, when this work was being done, there were commercial products that were also intended to track a person automatically, but they required that the person wear a target device that ran on batteries. The ICU avoided that need by using two cameras:

- a Spotting Camera that watched the entire area of interest, and
- a robotic Tracking Camera that followed the moving person.

Software analyzes the Spotting Camera image, looking for moving objects on the presentation stage. It then points the Tracking Camera at those moving objects by robotically moving the camera and adjusting the lens. See **Figure 2**.

While simple to describe, creating a refined ICU took some doing. But eventually the ICU Tracking Camera became useful enough that it proved a great boon to the operator by considerably lightening the work load. The result was fewer cases of not showing the most interesting image.

Reducing the operator's load didn't fix the scheduling problems. But it did offer an alternative.

By using the ICU Tracking Camera for a picture-in-picture shot of the person and the projection screen, we improved

⁴Yes, it is a pun: I-See-You.

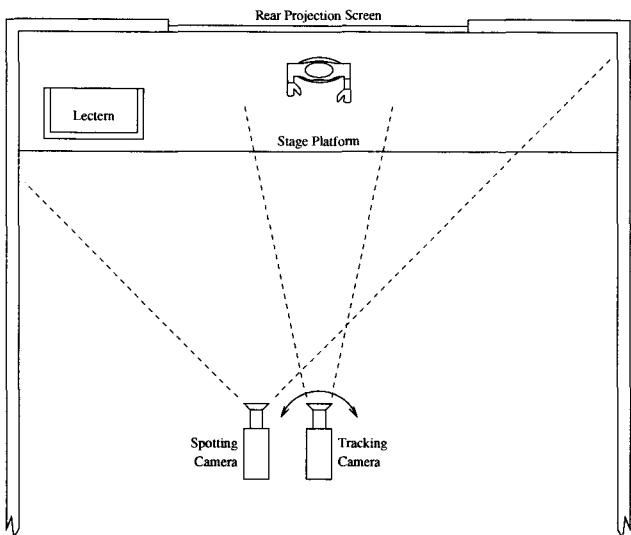


Diagram of the overhead view of the Spotting and Tracking Cameras.

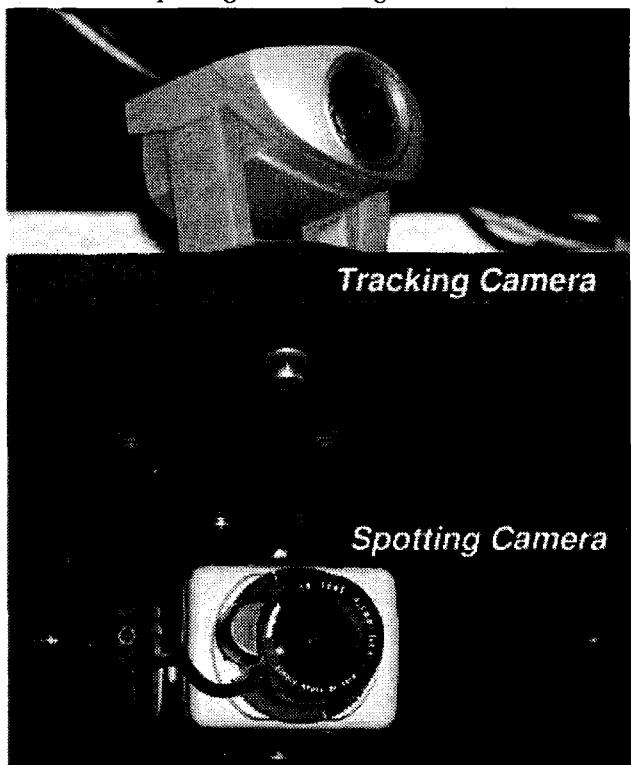


Figure 2: The Tracking and Spotting Cameras are mounted close together to minimize parallax between the Spotting Camera image and the pan, tilt and zoom axes of the Tracking Camera.

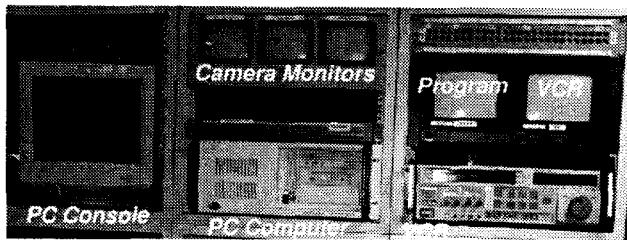


Figure 3: Some of the main components of the AutoAuditorium System. The Video and Audio Mixers are not shown.

the operatorless program quite a bit. It was *still* just one picture-in-picture image for the entire program, but there were fewer times when the person on stage was not visible. Again, these programs were not “good television” but the content telecast was so important to some viewers that the effort was praised.

Automatic Camera Selection. Another insight lead to the full automation of the program production. The same technology that made the ICU Tracking Camera work could also determine:

- if the projection screen was blank or not, and
- if the projection screen had changed, and by how much.

From these observations, and a few heuristics and timers, the system acquired the ability to react appropriately to most of the events of a presentation.

By making use of the serial control interface on the Distributed Auditorium System’s video mixer, the entire program could be produced in real time without any human control beyond:

- turning the system on,
- connecting the video signal to the corporate television network (if telecasting), and
- starting the VCR (if recording).

The result became known as the AutoAuditorium System (AutoAud for short) and as it became more refined and more accepted, it gradually took work away from the human-controlled Distributed Auditorium System.

The original prototype AutoAuditorium System, considerably updated, still runs at Telcordia Technologies. It still shares the automatic audio mixing function with an original Distributed Auditorium System. See Figure 3.

The keyboard, mouse and monitors are only used for maintenance functions. Normal operation is controlled by an ON/OFF switch and an Operating Mode switch.

Commercial Applications of the AutoAuditorium System. During the late 1990s the audio/video industry started to commercialize the technologies that made the AutoAuditorium System possible. Originally it required several hardware “hacks” to put the right combinations of remotely controlled automatic focus, remote control zoom with position feedback, camera and video mixer synchronization (aka “gen-lock”), and video mixer remote control into the same application. Television equipment of that era often assumed that a person was ultimately in control of the

equipment. Remote control interfaces often did not provide feedback of the equipment state. The assumption was that the human operator could always see the image or which buttons were lit. Getting around those limitations took either considerable imagination or a willingness to void the warranties by opening up the equipment and soldering wires where needed.

Slowly, oh so slowly, the industry began to understand that more and more of their equipment would be controlled by computers instead of by people, and the interfaces became more complete.⁵

IBM Watson Research became the first commercial customer of the AutoAuditorium System in December 1999. By the end of 2001 they had three AutoAud Systems. Originally purchased in support of their e-Seminar [9] research project, it has since become a corporate service managed by the Audio/Video group.

The first IBM installation was in a newly renovated 110 seat lecture hall, designed to support studio-quality videos with permanently present cameras and microphones. Several fortuitous coincidences made it relatively easy to install the AutoAuditorium System as an alternative way of making videos in that room. They had selected camera equipment and pan/tilt heads that were compatible with AutoAuditorium control software, and their equipment placement was also suitable. When the renovations were completed, they quickly started using the AutoAuditorium System to make videos. The AutoAud System did such a good job (even with some of its first-installation learning curve issues) that the need for manual productions was greatly reduced.

The second IBM system was installed in their 300 seat formal auditorium as a retrofit. One installation challenge was an existing movable wall that could split the room in half. That required that there be two AutoAuditorium configurations, each quite different. A single rotary switch controlled the entire system, selecting between:

- System Off
- Full Room
- Half Room
- Maintenance Mode

The fact that the system could be controlled with single switch made user training minimal. AutoAuditorium videos made in that room also became popular.

IBM's third system uses one controller and one video mixer but two sets of cameras and microphones, one set in each of two rooms. A video+audio routing switcher under AutoAuditorium System control determines which set is used at a particular time. Again, a single rotary switch selected between the two operating modes.

Boeing Phantom Works also has an AutoAuditorium System in a room which can seat up to 80 people. It is very similar to the first IBM installation.

The original AutoAuditorium prototype system is still at Telcordia Technologies. It is now over 8 years old and was recently moved to a larger auditorium and updated.

⁵Today it is still common to find "computer-controlled" equipment where action commands do not have corresponding status queries, and vice versa, or where status cannot be requested or is not reported correctly while an action is in progress. "Can't you see what is happening?" No, I cannot. The good news is that some of the manufacturers do a good job of providing computer interfaces as capable as the human controls.

3. RECENT EXPERIENCES

The best measurements of AutoAuditorium System use come from IBM Watson Research. They have three systems in two locations in down-state New York; the first is at their laboratory in Hawthorne and the other two are in their Yorktown Heights research headquarters, about 10 miles away. These are connected by high quality video links so talks given in one location's AutoAuditorium room can be seen easily in the other locations.

Prior to the e-Seminar project and the AutoAuditorium System installation, the Audio/Video group made about 50 crew-based video productions a year.

When the AutoAud Systems were installed at IBM Watson Research, they were used to create source material for their e-Seminar project, which was already in progress. (In 2002 the project was renamed "Research Media Portal" [6].) Over time they moved away from manually recording programs, which resulted in more programs being captured. The room reservation system was given an additional check box to request an AutoAuditorium recording. At first the recordings were made on video tape, which were later encoded for delivery over IP networks. IBM has eight laboratories around the world, and these encoded files were sent via FTP to IBM VideoCharger™ servers at each lab.⁶

As their experience grew and the e-Seminar system improved, the roles reversed: the encoding was performed in real time and video tape was used as a safety backup. If the encoding was deemed good, the tape was used again for the next lecture.

By 2003 the interconnection between the Research Media Portal and the AutoAuditorium Systems had matured to the point where everything worked smoothly. The expectations of the users of the systems and the sponsors of talks were being met and most talks of consequence were recorded. That year the Audio/Video group made 65 talks produced using crews, only a few of which were recordings of talks in the AutoAuditorium rooms. During the same year, 233 AutoAuditorium videos were created, which is just shy of one every business day. The Research Media Portal library of recordings then contained over 600 presentations, the majority of which created in the AutoAuditorium rooms.

4. LONG TERM EXPERIENCE

Between the Distributed Auditorium and AutoAuditorium Systems, we have over 15 years of experience with business videos in research lab and business settings. While the following observations are mostly anecdotal, they point to some useful insights.

First and foremost, that not all recorded lectures are created equal, because not all lectures are created equal. The ones that were regarded as uninteresting by the live audiences generally were not viewed as recordings. Even a well-made video cannot rescue a badly given presentation.

On the other hand, less-than-perfect videos are still worth watching if the material is interesting. Some of our "locked-down" Distributed Auditorium videos were very tedious as television programs, but nonetheless valued by interested viewers because the material was presented well, the projection screen was always visible and the automatic audio mixing covered both the stage and the audience reasonably

⁶Streaming video is rarely sent to distant labs, because of the many timezones that separate them.

well. "I just could not get there that day. Thank goodness that recording was made."

The AutoAuditorium System does not require any attention from the people making presentations. It works best when people forget it is there and simply give their presentations as they normally would to their local audience.

The AutoAuditorium System design recognizes the capabilities and limitations of the technologies and each installation is customized to emphasize their strengths and accommodate their weaknesses. Similarly each installation is adjusted to the particular environment it operates in and to the most common uses we expect to see there. In short, a great deal of attention to detail is required before the system performs as a hands-off capability.

The AutoAuditorium Systems' customers use them mostly as a means to capture and share more information with more people, rather than as ways to reduce the costs of video productions. Sometimes their use is unscheduled, such as when a "little event" attracts more people than expected. More than once, the AutoAuditorium System has been turned on and the program routed to rooms down the hall where the overflow crowd is then able to watch in relative comfort.

5. AUTOAUDITORIUM PRODUCTION HEURISTICS

Modern automatically produced videos still fall short when compared to the best manually produced videos, but they often have advantages. Because they are automated, they follow some simple heuristics when deciding what to do next. But a tired, bored, uninterested or distracted operator will make mistakes, such as failing to show the screen because they did not notice when the projection changed. The automatic production heuristics are very unlikely to make that mistake.

And the people in the production crew may have other jobs. More than once we've heard of a person recording a lecture and being called away for another task. "What could I do? I just locked down a shot and walked away."

Automatically produced video can be superior to those that are hand-made. We've seen "professionally" produced videos with some glaring flaws:

- Some camera operators don't know when the screen is important and when the person is. We've seen programs with long shots of the back of a person who is clearly talking about what is on the screen, but we never see it. Because the speaker is moving a lot, pointing at the screen, the operator perceives the person as being "where the action is".

The automation doesn't know when the screen is more important than the person either, but the AutoAuditorium heuristics know enough to show the screen every time it changes and to hold the screen shot much longer if the change affects most of the projected image.

- Shots of the screen are sometimes held too little time to read. The lack of motion on the screen is seen as a reason not to show it.

Because of comments from our audiences, the AutoAuditorium heuristics tend to hold shots of the projection screen longer than we've seen in live-crew productions. Also, if the screen is unchanged for a long

period of time, it is reshowed periodically so the remote audiences can refresh their memories.

- Television production "wisdom" dictates that showing people watching a presentation is interesting, and changing camera angles and moving the camera constantly is "good television." But constant switching of camera angles and shots of the audience do not necessarily help the remote audiences understand the material being presented. The AutoAuditorium heuristics favor the projection screen and the people on stage far above other shots and camera angles.

The AutoAuditorium heuristics only show the audience and other covering shots when the projection screen has not shown anything new in a while. Then other shots are added to the program to avoid just cycling between the person and the unchanging screen.⁷

Reasonably well made videos of talks, lectures and seminars fill a very real need. The promise of ubiquitous multimedia is that "place shouldn't matter. You can get what you want, when you want it, wherever you are." But that is only true if what you want exists inside the network. If it doesn't, the rest is irrelevant. The ability to capture lectures, talks and seminars automatically with reasonable fidelity is crucial to meeting the needs of people who must attend from another place or at another time.

6. RELATED DEVELOPMENTS

The idea of automatically capturing talks, lectures and seminars as multimedia has been researched for years. For instance:

- the Declarative Camera Control Language[4]
- the eClass project[3],
- the E-Seminar project[9],
- design issues of capturing collaboration[10]
- intelligent camera management[8]
- the Smart Classroom[11],
- the Virtual Director project[5]
- the Virtual Videography project[7]

7. CONCLUSIONS

The promise of ubiquitous multimedia is the possibility of having access to all *forms* of information almost anywhere. But it isn't the *form* of the information that really interests us; it is the *content* that demands our attention.

The AutoAuditorium System addresses the problem of making a common form of information sharing, namely a presentation to a group of people with projected visual aids, readily available for distribution over the multimedia networks of today and tomorrow. It does so by using existing technology (cameras, video mixers, microphones, audio mixers, computers and image processing software) in existing environments (classrooms, lecture halls and auditoriums) to capture the events that are frequently held there (presentations, talks and seminars).

Users of the AutoAuditorium Systems are already moving towards the day when most presentations are available at a

⁷It is possible to adjust the heuristics to put more variety and "production values" into the programs if the customer feels it makes them look more professional.

distant place or a distant time. As we have seen, when a customer creates and keeps an AutoAuditorium program every business day, it suggests that they have achieved the ease-of-use necessary to make these every-day-events into multimedia communication assets.

8. REFERENCES

- [1] BIANCHI, M. H. AutoAuditorium: a fully automatic, multi-camera system to televise auditorium presentations. In *Joint DARPA/NIST Smart Spaces Technology Workshop* (1998). www.AutoAuditorium.com/nist/autoaud.html.
- [2] BIANCHI, M. H. Autoauditorium system home page, 2004. www.AutoAuditorium.com.
- [3] BROTHERTON, J. A. *Enriching Everyday Activities through the Automated Capture and Access of Live Experiences; eClass: Building, Observing and Understanding the Impact of Capture and Access in an Educational Domain*. PhD thesis, Georgia Institute of Technology, July 2001.
- [4] CHRISTIANSON, E. A. Declarative camera control for automatic cinematography. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence* (1996), pp. 148–155.
- [5] E. MACHNICKI, L. R. Virtual director: Automating a webcast. In *Proceedings of SPIE Vol. 4673: Multimedia Computing and Networking* (2002), pp. 208–225.
- [6] KERNMANI, P. The IBM research media portal, December 2002. www.poly.edu/Podium/eef2002.cfm.
- [7] MICHAEL GLEICHER, J. M. Towards virtual videography. In *ACM Multimedia 2000, Los Angeles, CA* (November 2000). www.cs.wisc.edu/graphics/Papers/Gleicher/Video/vv.pdf.
- [8] RUI, Y., HE, L., GUPTA, A., AND LIU, Q. Building an intelligent camera management system. In *Proceedings of ACM Multimedia 2001* (2001), pp. 2–11.
- [9] STEINMETZ, A. A. E-seminar lecture recording and distribution system: a bottom up approach from video to knowledge streaming. In *Proceedings of SPIE Volume 4312* (2001). www.spie.org/web/meetings/programs/pw01/confs/4312.html.
- [10] TRUONG, K., ABOWD, G., AND BROTHERTON, J. Who, what, when, where, how: Design issues of capture and access applications. In *Ubicomp 2001 Conference Proceedings* (2001), ACM Press, pp. 209–224.
- [11] YUANCHUN SHI, E. A. The smart classroom: Merging technologies for seamless tele-education. In *IEEE Pervasive Computing Magazine* (April-June 2003).

A Novel Video Coding Scheme for Mobile Devices

Yi Wang
Information Processing Center
Dept. of EE and IS
Univ. of Science and Tech. of China
Hefei, China 230007
wy1979@mail.ustc.edu.cn

Houqiang Li
Information Processing Center
Dept. of EE and IS
Univ. of Science and Tech. of China
Hefei, China 230007
lihq@ustc.edu.cn

Chang Wen Chen
Wireless Center of Excellence
Dept. of ECE
Florida Institute of Technology
Melbourne, FL 32901 USA
cchen@fit.edu

ABSTRACT

In this paper, we propose a novel video coding profile for the multimedia applications oriented for mobile wireless communication. Because mobile devices generally have limited computational capability and constrained power consumption, we have considered jointly both video coding efficiency and implementation feasibility when developing the proposed video coding scheme. We achieve a good trade-off between coding efficiency and complexity by minimizing the video reconstruction distortion caused by adopting reduced complexity algorithms. A number of experiments have been conducted and the results have shown the efficiency of the proposed profile. A demo implemented on the Nokia 6600 cell phone demonstrates the feasibility of video coding scheme for mobile devices in wireless communication applications.

Categories and Subject Descriptors

E.4.3 [Coding and Information Theory]: Data compaction and compression.

General Terms

Algorithms

Keywords

Mobile devices, video coding, mobile multimedia, H.264, computational complexity, wireless video

1. INTRODUCTION

The burgeoning of wireless communication has brought a new era of mobile wireless network based multimedia applications and services. With the benefit of the increase in bandwidth in the emerging 3G wireless networks, new applications and services such as mobile visual phones and video streaming over mobile devices are pervading people's everyday life. The multimedia services will become one of the most important components for the 3G wireless network.

Although numerous video coding schemes have already been developed for the past two decades, none of these schemes were truly designed specifically for applications in video communication over mobile wireless links. There are two fundamental constraints in video coding for mobile wireless transmission: low bit rate transmission and limited computational capacity. These two constraints are often contradictory because the desired increase in coding efficiency in order to satisfy the constraint of low bandwidth transmission usually results in an increase in the computational complexity of the coding algorithm. While reducing the computational complexity of the video coding scheme usually results in degradation in video coding efficiency that may become unacceptable to users.

For current video coding schemes, those with high coding efficiency typically are computationally complicated in implementation and therefore cannot be adopted for mobile devices. Those video coding schemes with reduced computational complexity that can be implemented on mobile devices are usually unable to provide adequate video coding efficiency for acceptable visual quality at the receiver.

In addition to the two fundamental constraints we indicated early, a video coding scheme developed for mobile device implementation also needs to meet various other resource constraints. In general, portable, battery operated mobile devices would pose the following particular constraints [3] that need to be carefully taken care of when implementing a video coding scheme:

- 1) Power (battery life)
- 2) Memory size
- 3) Cache size/memory bandwidth
- 4) CPU speed
- 5) Display Resolution
- 6) Bandwidth

Based on these constraints as well as the applications intended for widespread use of video over mobile devices, we believe the desirable attributes of a high performance video coding scheme developed for mobile device applications should include:

- 1) High coding efficiency to meet the limited bandwidth constraints.
- 2) Balance of coding efficiency and complexity of implementation geared towards inexpensive consumer-level appliances.
- 3) Low cost implementation of decoder, considering the limited CPU power and the limited cache/memory.
- 4) Massive deployment in vast numbers of consumer-level mobile receivers
- 5) Tools included should provide significant coding gain relative to the cost for their decoder implementation.

In this research, we propose a novel video coding profile for the multimedia applications oriented for mobile wireless communication aiming at a balanced trade-off between computational complexity and video coding quality. The scenario in which our scheme will be applied is called local playback; that is, video content is first downloaded to mobile device over wireless network, and then the content is played back on the device. Focusing on such playback application, we adopt several video coding strategies, respectively, at the encoder end and the decoder end.

At the encoder end, we shall adopt several techniques that can achieve high coding efficiency in order to meet the constraints of limited bandwidth for the transmission of compressed video. Since the encoding is not implemented on the mobile device, we can afford to adopt high performance video coding algorithms that may be computationally complex. Whereas at the decoder end, we shall develop some video decoding techniques with low computation complexity so as to facilitate the mobile device implementation with limited resource constraints, including CPU speed, memory size, cache size, memory bandwidth, and power supply. However, we employ several balancing strategies to ensure that a reduction in computational complexity would result in minimum coding efficiency degradation at the receiving end. That is, in the proposed video coding profile, both coding efficiency and implementation feasibility have been simultaneously taken into account. We aim at achieving a balanced trade-off between coding efficiency and complexity.

The rest of this paper is organized as follow. Section 2 introduces the proposed video coding scheme. We have described in detail each component of the scheme. We address implementation issues of each component and proposed corresponding solutions to reduce the complexity of the algorithm in order to meet the resource constraints. Section 3 presents the experimental results that demonstrate the performance of the proposed video coding scheme. Section 4 illustrates the implementation of the proposed video coding on Nokia 6600, one of the current mobile devices that can handle multimedia contents. Finally, Section 5 concludes this paper with some discussion.

2. THE PROPOSED VIDEO CODING SCHEME

We choose to implement the proposed video coding scheme to be compliance with the current H.264/AVC standard. H.264/AVC is the powerful and state-of-the-art video compression standard that has recently been developed by the ITU-T/ISO/IEC Joint Video Team consisting of experts from ITU-T's Video Coding Expert Group (VCEG) and ISO/IEC's Moving Picture Experts Group (MPEG). The H.264/AVC standard represents a delicate balance between coding gain, implementation complexity, and costs based on state of VLSI design technology.

2.1 The Overall Video Coding System

Like H.264, our proposed scheme is also based on the conventional block-based motion-compensated hybrid video coding [1]. The block diagram of the proposed video coding scheme is shown in Figure 1. The video input consists of 4:2:0 YUV sequence. Each frame is processed in units of macroblock which corresponds to 16x16 pixels in the original image. Each macroblock is encoded in either intra or inter mode. In either case, a prediction macroblock is formed based on a reconstructed frame. In Intra mode, prediction macroblock is formed from samples in the current frame that have previously been encoded, decoded and reconstructed. In inter mode, prediction macroblock is formed by motion-compensated prediction from one or more reference frames. The prediction macroblock is then subtracted from the current macroblock to produce a residual macroblock. Then, the residual macroblock is transformed and quantized to produce quantized transform coefficients. These coefficients are re-ordered and entropy encoded.

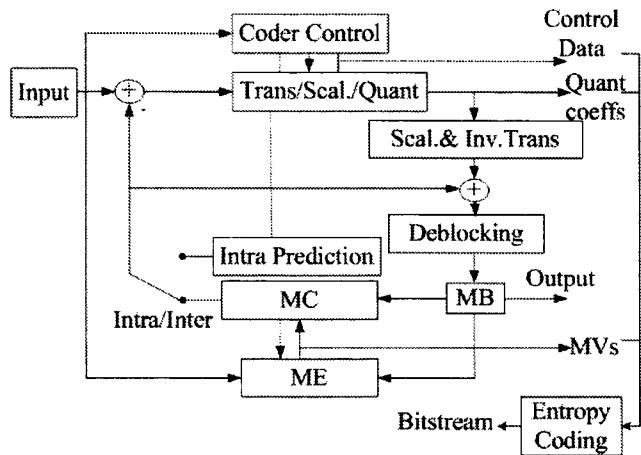


Figure 1: The block diagram of video coding

2.2 Implementation Details

To implement the proposed video coding scheme as shown in Figure 1 on mobile devices, we need to investigate each component of the video encoding and decoding in terms of its contribution to coding efficiency against its underlining implementation complexity.

Since the proposed video coding scheme is designed for local playback on mobile device, we focus on those coding modules that are adopted by both encoder and decoder. This is because these modules have greatest impact on both coding efficiency at the encoder and the implementation complexity at the decoder. However, when a module is included only at the encoder, we shall adopt a high coding efficiency coding technique even though the module is high in implementation complexity. Such coding module will not impact the mobile devices since these modules will not be implemented at the decoder.

In the following, we will elaborate in detail the proposed implementations on several key modules in video coding. One common feature of these implementations is that these key components will be compliance with the state-of-the-art H.264/AVC standard. Such implementation enables the potential widespread adoption of proposed scheme on various mobile devices from different manufacturers and for diverse applications.

2.2.1 Inter Mode

We employ an inter frame prediction approach that is the same as the approach adopted in H.264/AVC standard. In H.264/AVC, each inter mode corresponds to a specific partition of the macroblock into the block shapes used for motion estimation and motion compensation [1]. Each macroblock (16x16) can be partitioned into blocks with sizes 16x16, 16x8, 8x16 and 8x8. An 8x8 block can further be sub-partitioned into sub-blocks with sizes 8x8, 8x4, 4x8 and 4x4. In this case, each block has its own motion vector. Therefore, there are seven inter mode: INTER16x16, INTER16x8, INTER 8x16, INTER8x8, INTER8x4, INTER4x8, INTER4x4.

In addition to the motion-compensated macroblock modes described above, a macroblock can also be coded in the so-called skip mode. Skip mode is a special case of INTER_16x16 mode. For this special coding type, neither quantized prediction error signal, nor motion vector or reference index parameter is transmitted. The reconstructed signal is obtained by referring the picture which is located at index 0 in the multi-reference buffer. The motion vector used for reconstructing the Skip macroblock is similar to the motion vector predictor.

Although these seven inter modes are noticeably different from previous video coding standards, the adoption of such

inter modes facilitates more precise motion compensation and therefore results in higher coding efficiency. However, the adoption of multiple inter modes does not lead to an increase in implementation complexity at the decoder. We adopt these seven inter modes in the proposed video coding scheme.

2.2.2 Intra Mode

In H.264, two types of intra coding together with chroma prediction are supported. The first type of intra coding can be denoted as Intra_4x4. The Intra_4x4 mode is based on predicting each 4x4 luma block separately and is well suited for coding parts of a picture with significant details. The second type can be denoted as Intra_16x16. The Intra_16x16 mode, on the other hand, performs prediction of the whole 16x16 luma block and is more suitable for coding those relatively smooth areas of a picture. In addition to these two types of luma prediction, a separate chroma prediction scheme has been developed. For Intra_4x4 mode, nine prediction methods are available. These are vertical, horizontal, DC, diagonal down-left, diagonal down-right, vertical-right, horizontal-down, vertical-left, horizontal-up. Four prediction methods are available for Intra16x16 mode. These are vertical, horizontal, DC, and Plane. Chroma prediction has the similar methods to Intra_16x16 mode. More details on these intra modes can be found in [2] [5].

To reduce complexity, we only adopt Intra_4x4 mode and use five prediction methods in the proposed scheme. A list of modes is shown in Table 1.

Table 1: Luma Intra Modes

Mode Number	Mode Name
0	Vertical
1	Horizontal
2	DC
3	Diagonal left
4	Diagonal right

We use the same prediction methods for intra chroma block as H.264. A list of these methods is shown in Table 2.

Table 2. Chroma Intra Modes

Mode number	Mode name
0	Vertical
1	Horizontal
2	DC
3	Plane

By extensive experiments, we found that the intra coding without the Intra_16x16 mode and the more complicated prediction methods for Intra4x4 mode only leads to negligible loss of coding efficiency. So it is a good choice to take out the Intra_16x16 mode as well as the four intra_4x4 prediction methods in order to reduce the implementation complexity.

2.2.3 Quarter-sample-accurate motion compensation

In H.264, the accuracy of motion compensation is in units of one quarter of the distance between luma samples. It has been shown that such sub-pixel accuracy in motion compensation would lead to significant gain in coding efficiency. We have also conducted extensive experiments and found that this technique can achieve about 0.8 dB gain comparing with the computationally simpler half-sample-accurate motion compensation. Therefore, we enable this technique in the proposed scheme.

2.2.4 Rate Distortion Optimization

Rate distortion optimization (RDO) technique can be used to successfully improve video coding efficiency. RDO technique is usually used by encoder only. Through computing the actual distortion and the rate for every mode, RDO can more precisely decide the best mode than the general method that is based only on the sum of absolute difference (SAD). In our experiments, the RDO technique can improve the coding efficiency by about 0.4dB. Because it is implemented in encoder, the adoption of the RDO will not affect the complexity of the decoder. We believe the adoption of RDO is necessary and have implemented this technique in our scheme.

2.2.5 Transform & Quantization

One main feature of the H.264 standard is that the transform adopted in H.264 encoding is different from all previous video coding standards. An integer transform has been adopted in H.264. With integer transform, all operations can be carried out with integer arithmetic and therefore result in no loss of accuracy. A scaling multiplication (part of the complete transform) is integrated into the quantizer in order to reduce the total number of multiplications. The integer transform can be written as [5]:

$$Y = CXC^T = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 1 & -1 & -2 \\ 1 & -1 & -1 & 1 \\ 1 & -2 & 2 & -1 \end{bmatrix} [X] \begin{bmatrix} 1 & 2 & 1 & 1 \\ 1 & 1 & -1 & -2 \\ 1 & -1 & -1 & 2 \\ 1 & -2 & 1 & -1 \end{bmatrix}$$

where X is the input, C is the transform matrix, and Y is the result of the transform. We choose a similar transform to that of H.264. The transform matrix is:

$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 3/2 & 1 & -1 & -3/2 \\ 1 & -1 & -1 & 1 \\ 1 & -3/2 & 3/2 & -1 \end{bmatrix}$$

We adopt the same quantization methods as H.264. A total of 52 values of Qstep are supported by our scheme and these are index by a Quantization Parameter, or a QP. The values of Qstep corresponding to each QP are shown in Table 3. Note that Qstep doubles in size for every increment of 6 in QP, Qstep increase by 12.5% for each increment of 1 in QP.

Table 3 Quantization step sizes

QP	0	1	2	3
Qstep	0.625	0.6875	0.8125	0.875
QP	4	5	6	7
Qstep	1	1.125	1.25	1.375
QP	8	9	10	11
Qstep	1.625	1.75	2	2.25
QP	12	51
Qstep	2.5	224

2.2.6 Interpolation

Interpolation is necessary since we adopt the quarter-sample-accurate motion compensation. In the proposed scheme, the accuracy of motion compensation is the same as that of H.264 and is in units of one quarter of the distance between luma samples. As in H.264, the prediction values at half-sample positions are obtained by applying a one-dimensional 6-tap FIR filter (1, -5, 20, 20, -5, 1) horizontally and vertically. Prediction values at quarter-sample positions are generated by averaging samples at integer- and half samples positions. More detail can be found in [1].

The operation of interpolation consumes the most run-time among all functions of a decoder. Since 6-tap interpolation filter is much more complex than 4-tap filter, so we adopt a 4-tap filter (-1, 5, 5, -1) in our implementation in the mobile devices for prediction value of half-sample position and bilinear filter for value of quarter-sample position. With this change in interpolation filter, we are able to significantly reduce the complexity of the decoder.

2.2.7 Entropy Coding

Since entropy decoding is necessary for any decoder, if entropy coding scheme is too complex, then, the scheme cannot be realized on mobile device. For this reason, in our scheme, we adopt the simple Exp-Golomb entropy coding instead of complicated entropy coding methods such as CABAC (Context Based Adaptive Binary Arithmetic Coding) and CAVLC (Context-based Variable Length Coding), which have been adopted in H.264. The detail of CAVLC and CABAC can be found in [5][6].

Exp-Golomb codes (Exponential Golomb codes) are variable length codes with a regular construction. Table 4 lists the first 9 codewords.

Table 4: List of Exp-Golomb codewords

code_num	Codeword
0	1
1	010
2	011
3	00100
4	00101
5	00110
6	00111
7	0001000
8	0001001
...	...

It is clear from this table that the codewords progress in a logical order. Each codeword can be constructed as follows:

[M zeros][1][INFO]

where INFO is an M-bit field carrying information. The first codeword has no leading zero of trailing INFO; codewords 1 and 2 have a single-bit INFO field; codewords 3-6 have a 2-bit INFO field; and so on. The length of each codeword is $(2M+1)$ bits.

Each Exp-Golomb codeword can be constructed by the encoder based on its index code_num:

$$M = \log(code_num + 1)$$

$$INFO = code_num + 1 - 2^M$$

A codeword can be decoded as follows:

1. Read in M leading zeros follow by 1.
2. Read M-bit INFO field.
3. $code_num = 2^M + INFO - 1$

2.2.8 In-the-loop deblocking filtering

One particular drawback of block-based coding is the possible generalization of visible block structures. Block edges are typically reconstructed with less accuracy than interior pixels and “blocking” is generally considered to be one of the most visible artifacts with the present image and video compression methods. This is known as blocking artifacts. At low bit rate, blocking artifacts are very obvious. For this reason, H.264 defines an adaptive in-loop deblocking filter. This technique can improve both objective and subjective video quality. A detail description of the adaptive deblocking filter can be found in [7]. In our scheme, a simpler version of loop filter than the one used in H.264 is adopted. We simplify the filter by the following aspects:

1. Modify the strength table of H.264 (change from 4 BS to 2 BS);
2. Rework Alpha, Beta & Clip_tab tables to fit the new strength table and reduce threshold;
3. Delete one threshold called “small_gap”. So we can reduce the filter complexity by 2 shifting and 2 addition operation;
4. Use simplified filter.

2.2.9 Drop Frame

To reduce complexity and delay, B frame has not been employed in our scheme. For wireless application, a video coding scheme should support temporal scalability. Without B frame, if needed, we can only drop P frame to support temporal scalability. This will unavoidably brings drifting error. To solve this problem, we adopt a special “p” frame that is not referred by other frames. This is demonstrated in Figure 2. The role of this special p frame is much like the B frame with only forward prediction. Such a p frame can support dropping frame since it is not referred by other frames.

The introduction of this special p frame will not increase the implementation complexity. However, it may cause some loss of the coding efficiency. To minimize the loss due to this type of p frame, we choose two reference frames technique in our scheme in order to make up the loss caused by the introduction of this special p frame.

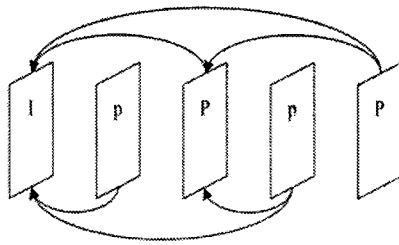


Fig. 2: Special p frame that is not referred by other frame.

3. Experimental results

Extensive experiments have been performed to verify the performance of the proposed video coding scheme. We compare the coding performance of the proposed scheme with that of JM76.

JM76 is the reference software of H.264 with version 7.6. The experimental settings are show in Table 5.

Table 5: Experimental settings for JM76 and this scheme

	JM76	our scheme
Inter Mode	seven	seven
Intra Mode	nine for Intra4x4 and four for macroblock	five for Intra4x4
Motion Vector	quarter-accuracy	quarter-accuracy
RDO	enable	enable
Transform	4x4 integer transform,	4x4 integer transform
Quantization	52 steps	52 steps
Interpolation	6-tap for half-pix and bilinear for quarter-pix	4-tap for half-pix and bilinear for quarter-pix
Entropy Coding	CAVLC	Exp-Golomb
Loop filter	enable	enable
reference frame	two	two

As shown in Figure 5, JM76 includes all techniques that have been adopted by H.264 video coding standard. JM76 and documentation for the reference software can be

downloaded from [4]. We choose four 15HZ QCIF sequences as the test sequences: paris, news, mobile and football.

Figure 3 shows the coding performance of the H.264 and the proposed scheme in terms of PSNR.

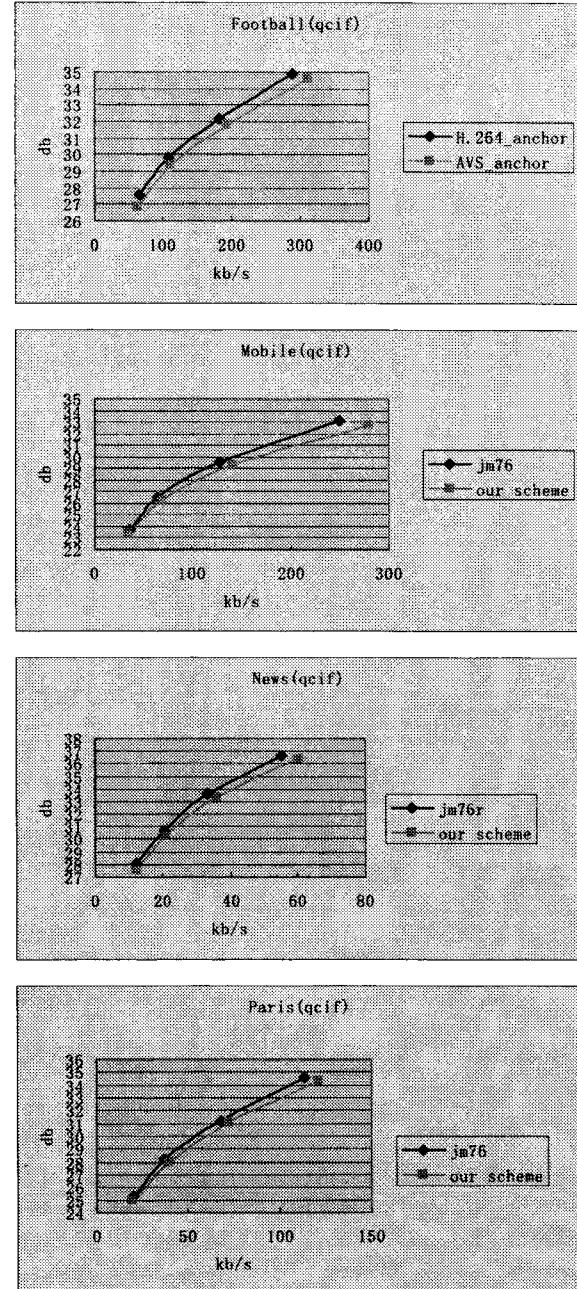


Figure 3: PSNR versus bit rate comparison between JM76 and the proposed scheme for test sequences of football, mobile, news), paris, all in QCIF format and at 15Hz.

From the performance curves shown in Figure 3, we can conclude that the proposed scheme lose on average

between 0.5dB and 0.6dB in PSNR, comparing with JM76 reference. At low bit rate, the loss can be slightly more. However, the complexity of the proposed scheme is much lower than that of JM76. With the necessary reduction in implementation complexity, the proposed video coding scheme can be implemented on mobile devices.

4. Implementation on Nokia 6600

To verify that the proposed reduced complexity video coding scheme is suitable for mobile devices, we have implemented decoder of our scheme on Nokia 6600. The Nokia 6600 image phone is the first device compliant with Series 60 Developer Platform 2.0. Researchers can develop application programming using Symbian OS v7.0s C++ APIs, or in JavaTM using MIDP 2.0 and related APIS for Mobile Media (JSR-135), Mobile Messaging (JSR-120), and Bluetooth (JSR-82). Nokia 6600 has a bright TFT display which supports 65536 colors (16 bit) with a spatial resolution of 176x208. Heap memory size of Nokia 6600 is 3MB and shared memory size for storage is 6MB + MMC. Nokia 6600 uses chip of ARM as its CPU.

There are several challenges in implementing the proposed video decoder on Nokia 6600 phone. One challenge is in managing the limited size in memory. Stack memory size is constrained under 8k by Symbian OS, hence we are unable to allocate large memory for local variables. We only place them in heap. This leads to speed loss. The most difficult challenge is to maintain adequate decoding speed. Video decoder needs to go through many computing steps, including complex steps such as inverse transform, inverse quantization, and interpolation. The overall complexity of decoder is therefore very high. Unfortunately, the speed of ARM adopted by Nokia 6600 is low comparing with that of Intel CPUs or AMD CPUs used in PCs. We adopt C++ as our implementation language, because C++ has higher running efficiency than Java. To overcoming the challenge of decoding speed, we adopt simple techniques in video coding and conduct some code optimization. To further improve the speed, we insert some assembly language in our C++ code. The results of implementation are shown in Table 6.

5. Conclusions

Based on the proposed scheme, we can play video at the speed of about 7 frame/s and the visual quality is similar to that of H.264 at the same bit rate. It is clear that the proposed scheme is able to achieve a balanced trade-off between coding efficiency and complexity and is suitable for implementation on mobile devices.

Our future work includes two aspects: one is to further improve decoding speed and the other is to improve coding efficiency without increasing implementation complexity.

We believe further improvement in decoding speed can be achieved when we fully consider the hardware structure of mobile devices. With appropriate rate control, we can also improve coding efficiency without increasing complexity. We also plan to add error control techniques, such as error resilience, error protection, and error concealment into our scheme. With these techniques, the scheme will eventually become robust for multimedia communication over mobile wireless networks.

Table 6 Implementation result

sequence	frame num	time(s)	speed (frm/s)
Bus	75	11.4	6.94
Foreman	150	21.105	7.11
Tempete	125	19.545	6.39
Football	130	18.975	6.85
Mobile	150	23.160	6.48
News	150	17.880	8.39
Paris	150	18.510	8.10

6. References

- [1] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," IEEE Trans. Circuits Syst. Video Technol, vol. 13, pp. 560-576, July 2003.
- [2] A. Tamhankar and K. R. Rao, "An overview of H.264/MPEG-4 part 10", Communications and Signal Processing Track, Texas Systems Day 2003, November 15, 2003, Dallas Hall, SMU.
- [3] "Mobile Profile Proposal", ISO/IEC, JTC1/SC29/WG11 and ITU-T SG16 Q.6, document JVT-C161, 3rd meeting, Fairfax, VA, USA, 6-10, March, 2002.
- [4] <http://bs.hhi.de/~suehring/tm1/>
- [5] "Draft ITU-T recommendation and final draft international standard of joint video specification (ITU-T Rec. H.264/ISO/IEC 14496-10 AVC," in Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG, JVT-GO50, 2003.
- [6] D. Marpe, H. Schwarz, and T. Wiegand, "Context-Based Adaptive Binary Arithmetic Coding in H.264/AVC Video Compression Standard", IEEE Trans. Circuits Syst. Video Technol, vol 13, pp.620-636, July 2003.
- [7] P.List, A.Joch, J.Lainema, g.Bjontegaard, and M.Karczewicz, "Adaptive deblocking filter," IEEE Trans. Circuits Syst. Video Technol, vol 13, pp. 614-619, July 2003.

Utilising Context Ontology in Mobile Device Application Personalisation

Panu Korppiä¹, Jonna Häkkilä², Juha Kela¹, Sami Ronkainen², Ilkka Känsälä²

¹VTT Electronics

P.O.Box 1100, FIN 90571 Oulu, Finland

Email: firstname.lastname@vtt.fi

²Nokia

P.O.Box 300, FIN 90401, Finland

Email: firstname.lastname@nokia.com

ABSTRACT

Context Studio, an application personalisation tool for semi-automated context-based adaptation, has been proposed to provide a flexible means of implementing context-aware features. In this paper, Context Studio is further developed for the end users of small-screen mobile devices. Navigating and information presentation are designed for small screens, especially for the Series 60 mobile phone user interface. Context ontology, with an enhanced vocabulary model, is utilized to offer scalable representation and easy navigation of context and action information in the UI. The ontology vocabulary hierarchy is transformed into a folder-file model representation in the graphical user interface. UI elements can be directly updated, according to the extensions and modifications to ontology vocabularies, automatically in an online system. A rule model is utilized to allow systematic management and presentation of context-action rules in the user interface. The chosen ontology-based UI model is evaluated with a usability study.

Categories and Subject Descriptors

H.5 [Information Interfaces and Presentation]: User Interfaces – evaluation/methodology, *graphical user interfaces, user interface management systems*

General Terms

Design

Keywords

Context awareness, ontology, mobile device, application personalization, user interface, rule, Context Studio

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MUM'04, October 27–29, 2004, College Park, Maryland, USA.
Copyright 2004 ACM 1-58113-981-0/04/10...\$5.00.

1. INTRODUCTION

The notion of adapting application behavior to context has received critique in the literature. Even though the goal of context-aware computing is well founded, developing devices that can sense the situation and adapt their actions appropriately faces one foundational problem; context awareness exhibited by people is radically different from that of computational systems [4]. Hard-coded fully automatic actions based on context are rarely useful, and incorrect automatic actions can be frustrating. Greenberg points out that it is not always possible to enumerate a priori a limited set of contexts that match the real-world context [6]. If such a set is found and is valid today, it may be inappropriate at any other time because of “internal and external changes in the social and physical circumstances”. Moreover, determining an appropriate action from a given context may be difficult.

However, it is not necessary to aim at fully automated actions as the only goal of context awareness. Mäntylä et al. propose to use an automation systems categorization for context-based adaptation [11]. Three different levels of automation of context-dependent actions can be distinguished: manual, semi-automated, and fully automated. Manual actions refer to actions made by the user based on context information, which is detected by the device (or the user). At the semi-automated level the user may predefine application actions based on context detected by the device, or choose from the actions proposed by the device based on context. At the fully automated level the application automatically takes (pre-programmed) actions according to the context detected by the device.

The semi-automated adaptation model partially overcomes the problem [6] of determining an appropriate action based on context. If the event-action behavior is defined by the end user instead of the application developer, a greater degree of personalization and flexibility can be achieved. Further flexibility is achieved by letting the user change the event-action configurations if it is required when the circumstances change over time.

Ranganathan and Campbell introduce a first-order logic-based rule model and distributed framework for defining context-based application actions [13]. As further work, the authors identify developing a graphical interface that lets the user choose the

elements of a rule from the available contexts and actions, instead of writing first-order logic. Sohn and Dey introduce an informal pen-based tool that allows users to define input devices that collect context information, and output devices that support response [15]. Inputs and outputs can be combined into rules and tested with the tool. The tool is not designed for small-screen mobile devices, nor are the user interface elements generated based on an explicit information model. The authors identify as future work the goal of enabling both designers and end users to create and modify context-aware applications. Dey et al. experiment with a programming-by-demonstration approach for prototyping context-aware applications [2]. The authors have developed a tool that allows the user to train and annotate models of context by example, which can be bound to actions. Again, the user interface of the tool is not designed for small-screen mobile devices. Furthermore, the focus differs from that chosen in this paper, which discusses a scalable model for representing the contexts of an ontology in the UI, where the contexts are freely selectable by the user.

The approach of learning contexts by example from a set of primitives from multisensor data has a fundamental problem: all primitives function as inputs for every context, acceleration, orientation, temperature, sound, light, humidity, heart-beat, etc. Therefore, when the user attempts to train the device in a certain context, the training data usually contains irrelevant inputs that are unintentionally included in the model. For example, if the user trains walking by walking near a road with a high rate of traffic, the model may learn that traffic noise is a part of walking and not recognize it without it. On the other hand, if walking is trained in silence, it may not be recognized in noise. With an accurately predefined ontology for predefined sensor and other inputs, and accurate rule definition by the user, context-action rules will only contain those inputs that the user wants to have.

Context Studio is an application personalization tool for semi-automated context-based adaptation. Application personalization is performed with a graphical user interface that allows the user to bind contexts to application actions. The Context Studio concept was introduced by Mäntyjärvi et al., who also presented a usability evaluation for a proof-of-concept prototype of the tool [11]. This paper utilizes the results and experiences from the study and further develops the concept; the usability problems found in the earlier study are corrected, and a user interface designed for small screens, especially Series 60 mobile devices, is evaluated with users novel to the concept of context-awareness. Furthermore, context and action vocabulary models, a rule model, and a context framework [8] are utilized in the new prototype. These models enable dynamic management of rules and extensible context vocabularies, and enable automatically generating elements of the user interface at runtime.

Context ontology can be applied for enumerating all the possible context events that can be used for activating actions. Many definitions for ontologies are available [5]; the purpose of ontology in this paper is expressing information so that it is easily understandable by humans and readable by machines. The human understandability of context information enables the mobile device end user to configure context-aware features. The machine readability of context information enables dynamically forming user interface elements based on the ontology. Moreover, the ontology facilitates describing rules as Context Exchange Protocol (CEP) XML scripts, which can be executed by an inference

engine [10]. Many context models and ontologies have appeared in the literature, e.g. [3,14,16]. The ontology model for a sensor-based mobile context [7,8] is utilized in this study. A more detailed vocabulary model is presented as an enhancement to the previous model.

Ontologies can be divided into lightweight and heavyweight, based on the degree of “depth” and restrictions (axioms, constraints) on domain semantics. The ontology applied in this paper is considered a lightweight ontology, including concepts, concept taxonomies (vocabularies), and properties (structure) that describe concepts. Moreover, ontologies can be characterized according to the level of formality: highly informal (natural language), semi-informal, semi-formal, and rigorously formal. The ontology applied in this study is considered semi-informal, since it is expressed in a “restricted and structured form of natural language” [5].

The main contributions of this paper are the following. Context Studio is further developed for the end users of small-screen mobile devices. Navigating and information presentation are designed for small screens, especially for the Series 60 UI. The context ontology and vocabulary models enable the generating of user interface elements automatically and dynamically. The ontology hierarchy is transformed into a directory model representation for the user interface, allowing easy navigation. Extensions and modifications to the ontology vocabularies can be automatically updated into the UI. The rule model is utilized to allow systematic management and presentation of rules in the user interface. Moreover, the results of a paper model user evaluation of the user interface are presented to evaluate the chosen UI navigation logic and appearance.

The paper is organized as follows. Context ontology with an enhanced vocabulary model is introduced. The ontology structure and vocabularies are used in a formal rule model, which is presented next. The user interface model is based on the representation of vocabularies and rules. The navigation in the UI is explained with examples. Finally, the UI is evaluated based on the results from the paper prototype usability study, followed by future work and conclusions.

2. CONTEXT ONTOLOGY AND VOCABULARY MODEL

Context ontology facilitates automatic generation of graphical user interface views for triggers and actions in Context Studio, and lets users navigate within the ontology hierarchies and choose contexts for context-action rules. The ontology contains concepts that are used as triggers and actions in context-action rules. For example, the ontology can describe a set of possible contexts that can be produced by a classifier from sensor data. In an online system the classifier produces instances of the ontology, which are used for triggering the user-defined rules.

The ontology is defined by a knowledge engineer or a computer scientist before use, and it should preferably be verified by a group of people in order to make it a shared conceptualization. The end user of Context Studio utilizes the ontology through the graphical UI in defining context-action rules.

The context ontology applied in Context Studio consists of two parts: structure and vocabularies. Structure defines the common properties of context that are used across different domains and

applications. Vocabularies are application- or domain-dependent expandable context conceptualizations, which aim at understandability and simplicity for the end user as well as the application programmer. New vocabularies are developed for new domains.

The ontology structure is defined as a set of properties. Each context (object) is described using six properties: Context type, Context value, Source, Confidence, Timestamp, and Attributes [7]. Defining context vocabularies concerns defining sets of Context types and Context values. Other properties are not discussed in detail in this paper.

Ontology vocabularies are designed according to the domain or application needs. Vocabulary design is a process of defining human-understandable domain- or application-specific values for the Context type and Context value properties. Those values should categorize and describe the possible real-world situations so that the information is useful for applications and understandable by humans.

Context types are defined for naming and categorizing context values. Context type resembles a variable name, and context values resemble the set of values for the variable. Figure 1a illustrates the vocabulary model for one context type and a set of context values for the type. Relations between concepts are only named for clarity, and are not used for, e.g., inheritance. Figure 1b shows an example of one context description for the vocabulary. Context type can have one or more concepts. Each context type can have one or more context values. The context instance at a certain time can have one of the values for each Context type. Different Context types can have one or more common concepts. Hence it is possible and useful to form a tree hierarchy of Context type concepts, where the leaf nodes represent the context values. The hierarchy can be utilized in querying and subscribing to branches of the context tree, instead of only one path.

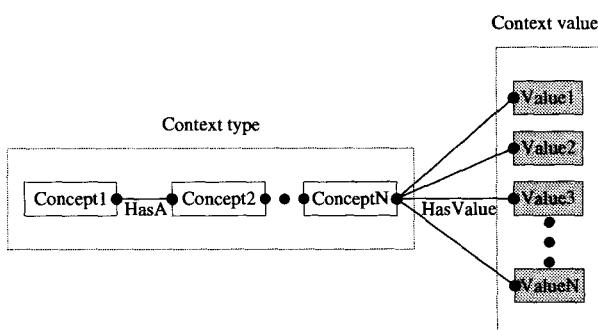


Figure 1a. A model for creating vocabularies consisting of Context types and Context values.

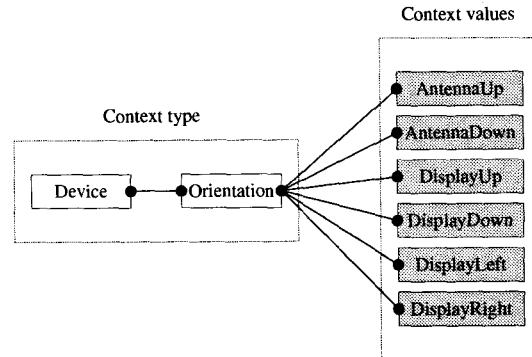


Figure 1b. An example Context type and a set of Context values.

The Context type concepts should be categorized from generic toward specific, so that more generic concepts are to the left, towards the root, and more specific concepts are to the right, toward the leaves. In this manner Context types can have common more generic concepts, but are separated by the more specific concepts.

For simplicity and ease of use, very long Context types should be avoided. At maximum, Context types should have three or four concepts. For example, if Context type concepts are modeled as folders in a user interface, navigating deep folder hierarchies is slow. On the other hand, Context types should be specific enough to allow having a small set of values. Too large a set of values is difficult to handle, at least in a user interface, if the user has to choose from a set of values. Each Context value should have a potential relevance to some application.

Table 1 presents an example of a context vocabulary describing device category contexts that are abstracted from acceleration and touch sensor data. The example in Figure 1b is described in text form in the first row of Table 1.

Table 1. Device category sensor-based context vocabulary example Context types.

Context type	Context values
Device: Orientation	AntennaUp, AntennaDown, DisplayUp, DisplayDown, DisplayRight, DisplayLeft
Device: Placement	AtHand, NotAtHand
Device: Activity	Still, Activity

Actions to be performed based on contexts can be represented by using the context vocabulary model. Actions are defined with two properties, Action type and Action value, which describe actions as Context type and Context value describe contexts. Table 2 presents an example of an action vocabulary. Moreover, external

devices can announce their actions, which can dynamically be included as Context Studio action vocabularies.

Table 2. Phone category action vocabulary example action types.

Action type	Action values
Phone: Applications: Profiles	Normal, Silent, Outdoors, Meeting
Phone:Joystick	JoystickUp, JoystickDown, JoystickLeft, JoystickRight, JoystickPress
Phone:Keypad	LockKeypad, UnlockKeypad

The ontology vocabulary model enables generating elements of the Context Studio user interface from the vocabularies. The hierarchical vocabularies are extensible and modifiable, and these qualities are inherited by the generated UI. Without the concept hierarchy, flat lists of Context values would easily grow too large to enable usable interaction.

3. NAMING CONVENTIONS

The context management of Context Studio is designed to utilize the context framework [8]. Appropriate naming is essential, particularly in describing the properties, Context type, Context value and source that are required for each context object added to the context manager blackboard. These properties are also used for accessing context information from the context manager. There are two main aspects to be considered in naming Context types. First, appropriate naming should reflect the meaning of context to the user, which is either the application developer or personalization tool user. Naming should reveal the use of the context. Second, correct naming convention ensures that the user of the context information can fully utilize the features provided by the context manager.

When Context types are built as paths consisting of elements from a generic to specific concept, context information can be accessed with partial Context types that represent a larger context information subset. This naming convention has also been utilized in CEP, where the reference to a subset of Context type hierarchy is called a wildcard [9]. Moreover, CEP recommends that vendor-specific context types are named starting with a prefix that names the vendor, e.g., "x-vendor_name:", followed with the normal Context type definition.

The set of Context values can be either numerical or symbolic. If Context values are described as numerical, for application use, they should have an explicit meaning to humans, such as environment temperature in degrees. If raw numerical measurement values are used, naming a Context value set is not required. Context type can be used normally. If Context values are defined for the purpose of application personalization by the user, they should be understandable by humans and symbolic, and the number of values in the set should be low enough to allow choosing a value from the set to function as a condition in a rule. When a small set of values is required, numerical values can be divided into named intervals - e.g. temperature value can be "over

20". Several other methods exist for abstracting numerical values into symbolic values [8].

4. CONTEXT STUDIO RULES

Context Studio enables the user to specify rules, which connect contexts to actions. Once the rules are active in the mobile device, contexts cause actions, as specified in the rule. Rule condition part elements are called triggers. A trigger can be any event or state that can be used for activating an action. Triggers are instances of the context vocabulary. Context type (variable name) and Context value (variable value) together define a trigger.

An action is any application, function, or event that can be activated when a set of triggers occur. The set may contain one or more triggers. A rule action part element is called action. For clarity, one rule may contain one action. Action type and Action value define an action.

A rule is an expression connecting a set of triggers to an action. Rules consist of one action element and one or more trigger elements. Rules are of the form IF trigger(s) THEN action. Based on the earlier user study [11], the rule structure is revised. To keep specifying rules simple for the user, the only operator for connecting triggers is AND. An example of a rule containing multiple triggers would look as follows: IF trigger1 AND trigger2 AND trigger3 THEN action. The operator OR is implemented by allowing the user to specify additional parallel rules for the same action. The user is allowed to make incomplete rules - i.e., the action or trigger part is allowed to be missing. Incomplete rules are represented with icons as disabled in the UI.

Context Studio adopts the CEP context script representation for rules [9,10]. Context scripts are semi-formal XML descriptions that can be used for describing rules. An example of a rule described as a context script is as follows.

```

<script
  xmlns="http://www.nokia.com/ns/cep/script/1.0/"
  xmlns:cep="http://www.nokia.com/ns/cep/1.0/">
  <if>
    <and>
      <equal>
        <atomRef name="Device:Activity" />
        <cep:string>Still</cep:string>
      </equal>
      <equal>
        <atomRef name="Device:Orientation" />
        <cep:string>DisplayDown</cep:string>
      </equal>
    </and>
    <actions>
      <notifyApp
        receiver="phone:apps/Activator">
        <cep:atom
          name="Phone:Applications:Profile">
          <cep:string>Meeting</cep:string>
        </cep:atom>
      </actions>
    </if>
  </script>
```

The example rule causes the device profile to be set to Meeting if the device orientation is display down and the device is still. Rules conforming to the script specification and Context Studio rule form can be displayed in the user interface. Hence it is possible to use rules defined by others, and exchanging rule scripts over the air is feasible. However, rules can be personal and thus not usable by others. For example, if the location of person's home is bound to coordinates, the description is not valid with anyone else's home.

Context management in Context Studio is based on the blackboard-based context framework for mobile devices [8] and rule scripts are evaluated with an inference engine, called script engine, [9] which uses context manager. Condition operations such as And, Or, Not, Equal, Not Equal, Less, Greater, Less Or Equal, Greater Or Equal, and In Range are supported by the context script model and the inference engine. In an online system context manager receives all context information from context sources and abstractors, and indicates script engine upon changes in those contexts that are used in the rule scripts generated by the Context Studio that have been defined by the user with the graphical UI. Script engine evaluates the rules for the changed contexts, and fired rules are used for activating application actions or platform events as defined in the rule. Details of context, rule, and application action management relating to Context Studio will be published later.

5. USER INTERFACE

Context Studio rules are specified by the user with a graphical user interface, following the notion of the semi-automatic adaptation model. The user interface is an application of context ontology. The representation of triggers and actions is based on the ontology vocabulary model. The Trigger, Action, and Rule view elements in the user interface can be automatically and dynamically generated based on the ontology and rule models. The ontology hierarchy is transformed into a folder-file model representation in the user interface, allowing easy navigation. Context and Action type concepts are represented as folders and subfolders, according to the vocabulary hierarchy. Context and Action values correspond to files in the concept folders. Extending and modifying vocabularies is possible at runtime, and the UI can be updated correspondingly. New Context and Action types are presented as new paths in the UI, and new Context and Action values are presented as new files in the folders.

Figure 2 shows three phases of navigating in the triggers view of the Context Studio UI. The UI corresponds to the vocabulary example given in Table 1. The UI is designed for small size screens, and the style follows Nokia Symbian Series 60 guidelines.

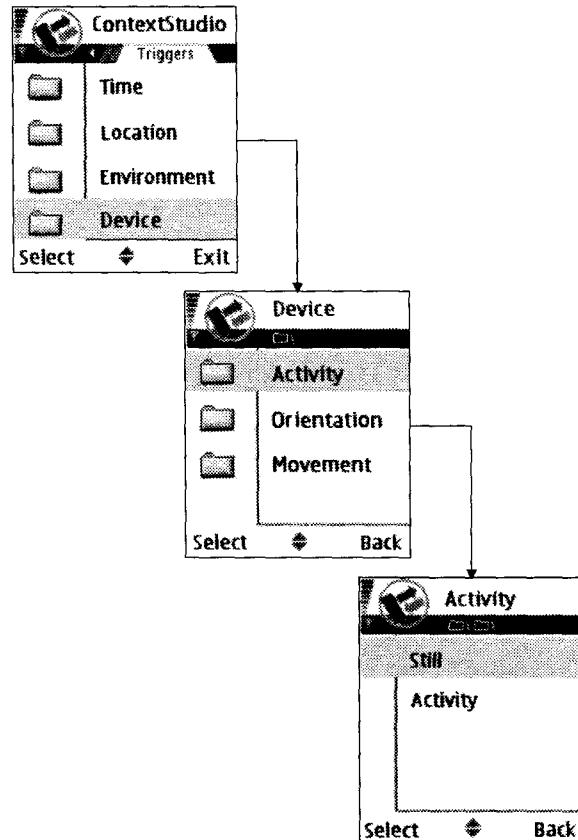


Figure 2. Navigating the triggers view in Context Studio.

In Figure 2 the user is navigating the Device category Context vocabulary in order to find the correct trigger for the rule in the earlier example. The user follows the path Device\Activity and, after choosing the context value Still, the view is returned to the rule edit view, Figure 3. Action hierarchy is displayed and navigated in a similar manner in the UI.

The rule scripts can be created, viewed and edited in the UI rule view. Figure 3 presents the view of the rule given in the example script.

Rule:

If the device is still and display down,

then set the phone audio profile to 'Meeting'.

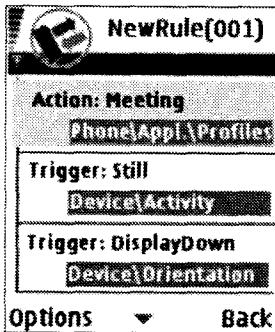


Figure 3. Context Studio rule edit view.

6. USER INTERFACE EVALUATION

The earlier Context Studio prototype [11] was evaluated with users. In this study the tool was improved based on the results from the earlier prototype, and redesigned to suit small-screen devices. Fewer operators were used for rule specifying, since overly complex rule structures have been found confusing by the users. The context and action navigation logic were redesigned and the rule view was simplified; the user may freely choose the trigger(s) for any rule, instead of marking predefined triggers. The hierarchy of contexts and actions is based on the scalable ontology vocabulary model, and the UI is generated accordingly.

6.1 Method and Participants

The Context Studio user interface was tested by paper prototyping with subjects who had no previous experience of context-aware systems. Each test session was conducted in controlled premises and lasted about 1.5–2 hours. A formative testing and evaluation approach was used for identifying problems and evaluating the overall design [1]. The test procedure was explained to the subjects and they were encouraged to voice their thoughts [12]. First, the subjects filled out a background questionnaire, charting their experience as mobile phone users. After this, they interacted with a paper prototype according to given scenarios, which included a short story and a task to be performed with the application. Between the scenarios the subjects were interviewed with questions related to each scenario. Finally, the subjects filled out a survey, where they assessed and commented on the test.

Eight subjects (4 male and 4 female), age 20–39, from different fields of study or work participated in the test. All eight subjects were users of mobile phones, five having previous experience with Nokia Series 60 phones. None of the subjects worked in the mobile phone industry.

6.2 Test Results

During the test the subjects constructed rules according to four scenarios, of which the first two are presented here as examples.

Scenario 1: You often call your best friend Mike. You want to be able to make the call quickly and without watching the phone, as you call him many times while driving a car or riding a bike. The

phone recognizes a 'shaking' movement. Now you want to determine that when you shake the phone it starts calling Mike.

Scenario 2: You quite often spend an evening in a nightclub with your friends. When you are clubbing you want your phone to turn to the 'Outdoor' profile so that you are able to hear when you receive a call or a message, and so you define an automated profile change.

A rule was constructed by selecting Context and Action values in the rule edit window (Figure 3). The selected values were found by navigating in the triggers and actions folder hierarchy (Figure 2). When constructing the rules, all subjects started from the rule edit window, where the *Action* and *Trigger* controls initially contained the value 'none'. Seven of the eight subjects first defined *Action* and then *Trigger(s)*, whereas one subject (#6) committed the tasks vice versa.

The concepts *Rule*, *Action* and *Trigger* were, in general, well understood. With the first task, six subjects could instantly complete the rule without any problems. Two subjects had initial difficulties in constructing a rule. Subject #5 failed to complete a functional setting, as (s)he defined only an action and no trigger. Subject #4 required some help with the application logic, although (s)he understood the idea of building a rule that would contain a condition and a consequence part. After completing the first task, none of the subjects failed to construct a rule, which suggests that the tool usage is easy to learn.

With the first scenario, four subjects (#1, 3, 6, 8) erroneously expected the gesture named 'shaking' to be the action as it described a physical action from the user. However, these subjects quickly spontaneously realized their misunderstanding when navigating in the action menu as the vocabulary referred to the consequential part of a rule.

The task given in Scenario 2 did not provide straightforward instructions for appropriate trigger selections but subjects had to infer them themselves. With this task the aim was to study the subjects' ability to define several triggers for one rule. Only two (#2, 8) subjects spontaneously realized that several triggers could be defined, and one of them (#8) still chose to use only one trigger. After completing the rule construction, the subjects were advised that defining several triggers was possible. However, they still chose not to add any more triggers but retained their original selections.

Generally, the vocabulary used in folder structure was perceived as intuitive and the navigation in the hierarchical structure did not cause difficulties. The presentation of both the trigger and action part in the same rule edit window (Figure 3) was perceived as a useful starting point for constructing a rule and for completing the editing. Some error steps appeared while navigating in the menu hierarchy. The errors were due to a misunderstanding with the vocabulary, leading to errors such as entering to the *InputEvents* folder instead of *Gestures* (#8). Two of the subjects mentioned being confused because of the length of the hierarchical menus: '*I have now long ago forgotten where I have gone*' (#4), and '*These seem to be behind quite a many selections...*' (#7). Some questions related to vocabulary were raised if terms were not commonly used in the subject's field of work or study.

At the end of the test the subjects were asked to assess the test on a scale from 1 (lowest) to 5 (highest). The ratings given by the subjects for the following items were *Usefulness of selected*

functions 3.6 (standard deviation 0.52), *Appearance* 3.9 (0.64) and *Logical structure* 3 (0.76). When asked '*Do you feel you would benefit from the Context Studio application?*', all subjects answered affirmatively. When interviewed, it was found that there was significant variation in which rules were perceived as most useful and which rules the users would probably use, depending on the user's lifestyle.

6.3 Analysis of the Test Results

The results of the user tests indicate that Context Studio is a potentially useful tool, and users could perform the given tasks well. The overall number of errors related to the interaction with the user interface was small. In particular, the performance with the vocabulary and hierarchical folder structure was good. Moreover, the idea of constructing a rule having a condition and a consequence part was well understood. Potential difficulties relate more to *the kind of rules* that are constructed than to the actual ability to do it.

Users have a tendency to avoid constructing long and complicated rules. The test indicates that this is affected by the user's low tolerance of required effort, elongated time spent with manual settings, and problems with outlining the increasing complexity.

Furthermore, the test results indicate that there are potential usability risks with users' subjective perceptions of context; they have varying opinions on the meaning of subjective concepts such as 'cold temperature'. Objective contexts should be preferred in the ontology vocabulary design [7]. The vocabulary should be verified with a group of people before use, and deep hierarchies should be avoided if possible.

The tests indicate that users are strongly influenced by what they perceive as useful device features for their particular lifestyle or usage preferences. The users would like to make personal context-based rules, which proves the usefulness of the chosen semi-automatic adaptation hypothesis.

7. CONCLUSIONS AND FURTHER WORK

Context Studio, an application personalization tool for semi-automated context-based adaptation, was further developed for the end users of small-screen mobile devices. The graphical user interface of Context Studio was designed to utilize context ontology. The ontology vocabulary hierarchy is transformed into a folder-file model representation in the UI. The ontology, with an enhanced vocabulary model, offers a scalable information representation and easy navigation of context and action information in the UI, and enables straightforward updating of the UI according to changes in the vocabularies. Furthermore, the ontology supports the utilization of context scripts as a formal rule model.

The tool was evaluated with paper prototyping. The usability study indicated that the tool was found useful, although the preferred context-aware features differed between the subjects. The general idea of constructing a rule with a condition and consequence part was quite well understood, and users had no problems in defining rules. However, they were not eager to construct rules matching multiple contextual conditions. Navigating hierarchical folder structures for selecting desired triggers and actions was intuitive, although care must be taken when designing vocabulary to keep it easily understandable and unambiguous.

A tendency to make simple rules was an obvious trend in the user test. Adding more rule operators is a consideration for the future, but the additional complexity of rules should not be allowed to reduce the usability of the tool. If making rules is too difficult, users will probably not use the tool.

Context Studio can be used for personalizing applications of a small-screen mobile device with a usable graphical UI. The range of possible context-aware features depends on the number and quality of the context values that can be acquired and abstracted from sensors and other sources, and the number of available actions. The usefulness of the self-configured features will be decided by the end user. Future work involves implementing Context Studio on top of the context framework in the target platform, and context abstractors to produce the contexts defined in the ontology vocabularies. The complete tool will enable study of the UI usability in the target device, and easily defining and evaluating context-aware application features in a real usage of a mobile handheld device.

8. ACKNOWLEDGMENTS

We thank Jani Mäntyjärvi and Urpo Tuomela for their helpful comments.

9. REFERENCES

- [1] Carroll, J.M. *Making of Use. Scenario-based design of human-computer interactions.* (Chapter 8). The MIT Press 2000, 368 p.
- [2] Dey, A., Hamid, R., Beckmann, C., Li, I., Hsu, D. a CAPpella: Programming by Demonstration of Context-Aware Applications. *Proc. CHI 2004*, ACM, 2004.
- [3] Henricksen, K., Indulska, J., Rakotonirainy, A. Modeling Context Information in Pervasive Computing Systems. *Proc. International Conference on Pervasive Computing 2002*, LNCS 2414. Springer-Verlag, 2002, 167-180.
- [4] Erickson, T. Some problems with the notion of context-aware computing. *Communications of the ACM*, 45(2), 2002, 102-104.
- [5] Gomez-Perez, A., Fernandez-Lopez, M., Corcho, O. Ontological engineering. Springer-Verlag, 2003, 403 p.
- [6] Greenberg, S. Context as a dynamic construct. *Human-Computer Interaction*, 16, 2001, 257-268.
- [7] Korpiä, P., Mäntyjärvi, J. An Ontology for Mobile Device Sensor-Based Context Awareness. *Proc. 4th International and Interdisciplinary Conference on Modeling and Using Context 2003*, LNAI 2680, Springer-Verlag, 2003, 451-459.
- [8] Korpiä, P., Mäntyjärvi, J., Kela, J., Keränen, H., Malm, E-J. Managing Context Information in Mobile Devices. *IEEE Pervasive Computing Magazine special issue: Dealing with Uncertainty* 2(3), IEEE Computer Society, 2003, 42-51.
- [9] Lakkala, H. Context Exchange Protocol Specification. 2003, 28 p. Available: <http://www.mupe.net>.
- [10] Lakkala, H. Context Script Specification. 2003, 22 p. Available: <http://www.mupe.net>.
- [11] Mäntyjärvi, J., Känsälä, I., Tuomela, U., Häkkilä, J. Context-Studio - Tool for Personalizing Context-Aware Applications

- in Mobile Terminals. *Proc. Australasian Computer Human Interaction Conference 2003*, 2003, 64-73.
- [12] Preece, J. Sharp, H. Rogers, Y. Interaction Design: Beyond Human-Computer Interaction. (Chapter 12) John Wiley & Sons, Inc. 2002, 519 p.
- [13] Ranganathan, A., Campbell, R. An infrastructure for context-awareness based on first order logic. *Personal and Ubiquitous Computing Journal* 7, Springer-Verlag, 2003, 353-364.
- [14] Schmidt, A., Aidoo, K.A., Takaluoma, A., Tuomela, U., Laerhoven, K., Van de Velde, W. Advanced interaction in context. *Proc. 1st International symposium on handheld and ubiquitous computing 1999*. Springer-Verlag, 1999.
- [15] Sohn, T., Dey, A. ICAP: An Informal Tool for Interactive Prototyping of Context-Aware Applications. *Ext. abstracts CHI03*, 2003, 974-975.
- [16] Wang, X., Zhang, D., Gu, T., Pung, H. Ontology Based Context Modeling and Reasoning using OWL. *Proc. Workshop on Context Modeling and Reasoning at IEEE International Conference on Pervasive Computing and Communication*, 2004, 18-22.

Design and Evaluation of mProducer: a Mobile Authoring Tool for Personal Experience Computing

Chao-Ming (James) Teng, Chon-In Wu, Yi-Chao Chen, Hao-hua Chu, Jane Yung-jen Hsu

Department of CSIE, Graduate Institute of Networking and Multimedia

National Taiwan University, Taipei, Taiwan 106

{jt, r92079, b89066, hchu, yjhsu}@csie.ntu.edu.tw

ABSTRACT

Personal experience computing is about computing support for recording, storing, retrieving, editing, analyzing, and sharing of personal experiences. In this paper, we present our design, implementation and evaluation of a mobile authoring tool called mProducer. mProducer enables a user to generate personal experience content using a mobile device anytime, anywhere. To address challenges in both limited system resources and user interface constraints on a mobile device, mProducer provides several innovative system techniques and UI designs. (1) The *Storage Constrained Uploading (SCU)* algorithm uploads large multimedia data to remote servers, in order to alleviate the problem of limited storage on a mobile device. (2) *Sensor-Assisted Automated Editing* utilizes a tilt sensor on the mobile device to automate the detection and removal of *blurry frames* resulting from excessive amount of camera shaking. This sensor-based solution requires small processing overhead, and it is considered a good alternative to computational-expensive image processing techniques for detecting shaking artifacts. (3) *Map-based content management interface* incorporates a GPS receiver on a mobile device to record location meta-data for each recording captured by a user, and enables easy, intuitive content navigation on a small screen. (4) *Keyframe-based editing* enables a user to edit content using only keyframes. We have conducted user studies to evaluate overall editing experience, user satisfaction in the editing quality, task performance time, ease-of-use, and learnability. The results of user studies have shown that keyframe-based editing works best with a storyboard interface. In general, users have found mProducer to be both fun and easy to use on a mobile device.

Categories and Subject Descriptors

H.4 [Information Systems Applications]: Miscellaneous; H.5.1 [Information Interface and Presentation]: Multimedia Information Systems; H.5.2 [User Interfaces]: Graphical user interfaces (GUI)

General Terms

Algorithms, Design, Experimentation, Human Factors.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MUM 2004, October 27-29, 2004 College Park, Maryland, USA.
Copyright 2004 ACM 1-58113-981-0 /04/10... \$5.00

Keywords

Personal Experiences, Multimedia Editing Tools, Sensors, Mobile User Interfaces

1. INTRODUCTION

The proliferation of camera-equipped phones and PDAs comes as a result of consumers' demand *not only* to be mass media consumers, but also content producers of their own personal experiences anytime, anywhere: where they go, what they do, and what they see and hear. The ability to record, edit and share footage of users' daily activities can be a strong selling point for these mobile devices with content producing capability.

Given the popularity of camera phones, it is expected that mobile content will be dominated by *personal experiences* produced by casual users. This is in contrast to the desktop computing world that targets professional content providers creating mass media content.

We believe that users are motivated to edit personal experiences directly on a mobile device, rather than to transfer content to a PC for editing. The motivations are that (1) they want to share their personal experiences *anytime, anywhere from a mobile device* – but prior to sharing them, they may want to perform simple editing functions to remove non-essential content or to add text or audio annotations; (2) they want to record important events as keepsakes – but given limited mobile storage, they want to keep only the essential content on a mobile device by removing unwanted recordings; and (3) typical users with little or no prior computing experience prefer to use a simple and intuitive user interface designed specifically for the mobile environment, rather than sophisticated PC-based tools that require a higher level of computer skills.

Specifically, the design of mProducer considers the following mobile challenges:

1. **Limited Storage:** Mobile devices have limited storage that restricts the length of recordings a user can capture.
2. **Limited Computing Resource:** Most image/video processing techniques for media editing are computationally intensive and demand the high computational power of PCs. They are beyond the limited computing resources on a mobile device.
3. **Specialized User Interface:** Small screens, inconvenient input methods, limited mobile user attention, and typical consumers with little computing experience require a different interaction model and user interface design, where

simplicity, ease-of-use, and good learnability are as important as the final quality of edited contents.

Although the idea of a multimedia authoring tool for mobile devices has been raised in [1, 9], we have yet to find a tool that address these challenges. In this paper, we describe our design, implementation and evaluation of mProducer, a mobile authoring tool which successfully addresses the challenges outlined. Our contributions include the following novel solutions:

- **Storage Constrained Uploading (SCU):** In order to overcome limited storage, video frames in content captured by a user are prioritized based on whether they will be needed during the user editing phase. When a mobile device runs low on local storage space, lower priority frames not needed for later editing will be uploaded to a server, allowing more contents to be captured on a mobile device. This technique can increase the size of contents captured on a mobile device by 14 times.
- **Sensor-assisted Automated Editing:** Existing desktop video editing tools use image-based processing methods to semi-automate editing on raw contents so that the amount of user effort can be reduced. Examples of these techniques include object recognition, location determination [11], lighting detection, and shaking artifacts removal [14]. Although these techniques are generally too computationally intensive to run on a resource-poor mobile device, sensors attached to the device can automatically achieve a similar result with relatively small computing cost. We describe how to use GPS and tilt sensor to automate editing for users.
- **Location-based Content Management:** When mProducer is used for capturing personal experiences at multiple locations (e.g., a trip covering multiple sightseeing venues), our studies have found that users mentally organize personal experiences based on the locations where the video clips were captured. To match users' *location-based mental model*, a simple, intuitive, map-based content management interface is designed to enable easy navigation and browsing of video clips. A GPS receiver on a mobile device is used to record location meta-data for each recording captured by a user, so that a user does not have to input it.
- **Keyframe-based Editing:** A keyframe is defined as a video frame that best represents a shot or a scene, i.e., a user can get a good understanding of what a shot is about by simply looking at its keyframe. Our user studies have shown that casual users can edit using only keyframes and produce satisfactory editing quality for the purposes of sharing and recording personal experiences.

The rest of this paper is organized as follows. Section 2 proposes the design of mProducer. Section 3 describes the storage constraint uploading algorithm. Section 4 explains how tilt-sensor is used in sensor-assisted automated editing. Section 5 explores the design of mProducer's user interface (UI). Section 6 discusses related work. Section 7 presents our conclusion and future work.

2. DESIGN

The current mProducer prototype covers two phases: *capturing phase* and *editing phase*. Typical usage of mProducer involves repeating patterns of capturing one or more clips, editing these

clips (which frees up space in the mobile storage), then refilling the freed space with newly captured clips.

2.1 The Capturing Phase

Figure 1 shows the execution flow within the capturing phase, starting with data captured from a mobile device and finishing with either storing the data on the mobile device or the server. In the 1st step, the camera and microphone on a mobile phone capture video and audio data in a buffer. The 2nd step applies the Shot Boundary Detection (SBD) algorithm¹ to divide a clip into disjoint shots² or scenes. In the 3rd step, data from a tilt sensor is used to automatically detect and remove blurred frames resulting from excessive amount of camera shaking (more details are described in section 4). In the 4th step, motion-JPEG encoding compresses incoming bitmap frames. In the 5th step, the Keyframe Selection Algorithm (KSA) [16] finds a representative keyframe for each shot, and keyframes are assigned higher priority than non-keyframes. In the 6th step, the SCU algorithm (described in details in section 3) uses the frame priority to either upload frames to the server or store them in the mobile device.

2.2 The Editing Phase

The editing phase consists of the three steps shown in Figure 1. In the 1st step, location-based content management organizes video clips based on their recording locations. A user starts editing video clips by first selecting a point on a map which represents recordings made there. In the 2nd step, when the user clicks on a map point, a list of clips is displayed to the user. The user then chooses a clip to edit. In the 3rd step, the user can edit the chosen clip using the keyframe-based editing interface. This interface design is described in more details in Section 5.

3. STORAGE CONSTRAINED UPLOAD

The storage constrained uploading (SCU) algorithm minimizes network communication (including both uploading and downloading) for content capturing and editing from a mobile device. We first describe the SCU algorithm in details, and then generalize this algorithm for different priority schemes.

The limited storage on a mobile devices' is barely sufficient for a user to record one complete experience. One solution to this problem is for a mobile device to upload recorded content to a server so that the amount of captured content is not limited by the mobile device's local storage. A naive approach would be to upload every piece of content to the server immediately after it is recorded, then download it back to the device whenever a user needs to edit it. The first problem with this approach is that transferring content that will later be deleted by the user is a waste of network bandwidth. The second problem is that limited wireless bandwidth is likely to result in slow content transfers, leading to a frustrating user editing experience. Therefore, we need a more intelligent mechanism to determine when to upload, and what portions of the contents to upload, from mobile storage to the server.

¹ We implemented SBD algorithm based on color histograms described in [6].

² A shot is defined as one or more frames generated and recorded contiguously and representing continuous action in time and space [7].

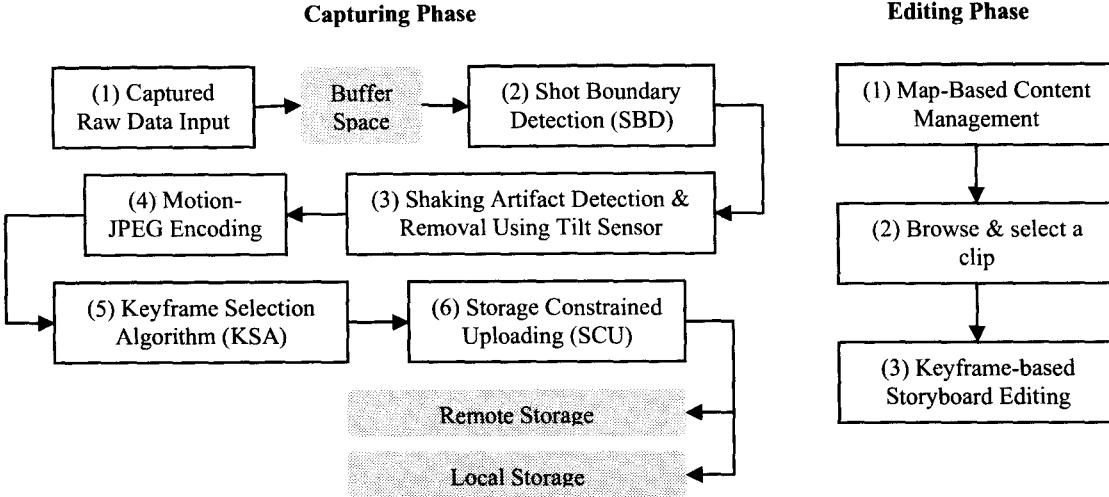


Figure 1: The capturing and editing phases

SCU will not upload contents to the server until the local storage space is nearly full. The reason for this is that we can avoid uploading frames that will later be cut by the user. SCU chooses frames for uploading based on the observation that there is a difference in quality requirements between personal experience editing tools targeting casual consumers, and editing tools targeting professional content providers. We believe that there is no need to provide a mobile authoring tool that can produce professional quality content. Fine-grain editing (e.g. frame-by-frame) used in a professional PC-based authoring tool for professional quality content is, in fact, unsuitable for a mobile authoring tool. This style of editing requires a significant amount of user effort, training and attention, high resolution screens, and high computational power.

When applying SCU in mProducer, frames were prioritized into two levels of importance: *keyframe* and *non-keyframe*. This prioritization is useful due to our observation that typical consumers are satisfied with editing using only keyframes. This allows a mobile device to provide editing functionality using only a subset of the total content being modified.

We define *editing granularity* to be the subset of frames used during editing. The finest editing granularity possible is frame-by-frame. The system may also only present keyframes or I-frames (for MPEG-encoded video) to users for editing. The editing granularity then becomes keyframes or I-frames. We have performed a user study to find out the granularity requirements of casual consumers. Our results have shown that typical consumers can edit and produce satisfactory quality (delete unwanted portions of video clips and add text to shots) when they were presented with keyframes only. This suggests that, for casual consumers, non-keyframes can be uploaded without degrading the editing experience. Further details about this user study are presented in Section 5.2.

3.1 SCU Algorithm

Initially, when mobile storage is empty, the SCU algorithm will store all frames including both keyframes and non-keyframes. The mobile storage is said to be at high storage granularity when it can store both types of frames. As a mobile user captures new frames, mobile storage may eventually run out of free space. The

SCU algorithm then enters low storage granularity when a new captured frame fills the remaining space in mobile storage. While mobile storage remains low, it will start uploading non-keyframes to the server in order to make room for incoming frames. In situations where network bandwidth (upload rate) is less than the content capture rate, a buffer on the mobile device is needed to temporarily store data. Eventually such a buffer would be filled and recording will be disabled on the mobile device. At this point, the user is notified to edit some clips, which are then uploaded to the server and removed from local storage. A mobile device re-enters high storage granularity again after local storage is cleared.

If mobile storage contains multiple clips, SCU uploads frames in round robin fashion among the stored clips, in order to maintain fairness among clips. When the storage granularity drops from high to low, the uploading of frames is done on an as-needed basis. SCU does not upload all non-keyframes at once to the server. The reason for as-needed uploading is to avoid unnecessary uploading of frames that will later be cut by users, as mentioned earlier.

Consider an uploading list that tracks the order of frames to be uploaded from the mobile storage. It sorts frames based on priority first then applies round-robin scheduling across the clips. Using this uploading list, mProducer can simply look at the head of the list to choose which frames to upload next. Note that the current policy in mProducer is to never upload keyframes, even when the storage is full. The main body of SCU algorithm is shown below. We denote the reserved space for mProducer in the storage as Z , the size of total frames in the storage is T , the i -th frame of clip # j as f_i^j , its size as $S_{f_i^j}$, the newly coming frame

as f_{new}^N , and N is the number of clips in the mobile storage. For more details for the SCU algorithm, please refer to our previous paper [2].

Algorithm 1 The basic SCU algorithm

Require: A new coming frame f_{new}^N (*size, type*)
Ensures: Frames to upload to storage server or save f_{new}^N

- 1: if $S_{f_{new}^N} + T > Z$ then
- 2: upload the frames in the order of the "Uploading List"
 until $S_{f_{new}^N} + T < Z$;
- 3: adjust the "Uploading List" accordingly
- 4: end if
- 5: if f_{new}^N is not uploaded then
- 6: save f_{new}^N and adjust the "Uploading List"
- 7: end if

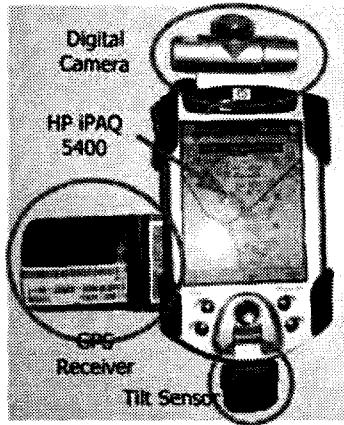


Figure 2: HP iPAQ 5450 with a digital camera, a GPS receiver and a tilt sensor

4. SENSOR-ASSISTED AUTOMATED EDITING

Existing video editing uses image processing to identify and extract meta-data *context information* at the time of production [8][11]. Sensors attached to a mobile device can achieve the same context information without high computational cost. This is ideal for a mobile device that has limited computing capability.

The current version of mProducer incorporates two sensors to automatically annotate captured contents with meta-data context information: (1) global positioning system (GPS) receiver detects location meta-data, and (2) a tilt sensor detects the amount of camera shaking. Note that excessive amounts of camera shaking results in blurry, unusable video, which can then be automatically detected and removed. A common example of unwanted video clips that can be detected by camera shaking is when a user forgets to hit the stop button after recording, leaving the device (while walking) in a pocket or a bag to continue capturing unwanted video clips. Figure 2 shows the hardware component of the prototyped system together with GPS receiver and tilt sensor.

GPS Receiver: it is the GPS-CF card from CHIPCOM Electronics. Each time a user records a video clip, mProducer will probe the GPS receiver for current location information. The GPS receiver has approximately 5 meters of accuracy outdoor. This clip will be annotated with location information. Our user studies have shown that typical consumers are more likely to merge video clips taken at the same location. This observation leads to the design of a location-based content management (described in

details in section 5.1), which organizes and groups contents based on their recording locations shown on a map. This enables users to easily and quickly navigate multiple video clips.

Tilt Sensor: it is TiltControl CF card made by ECER Technology as shown in Figure 3. It contains an accelerometer that measures the horizontal and vertical tilt of the device. Changes in the tilt are used to compute the magnitude of camera shaking and predict its impact on video quality. The sensor measures both direction and magnitude of tilt.

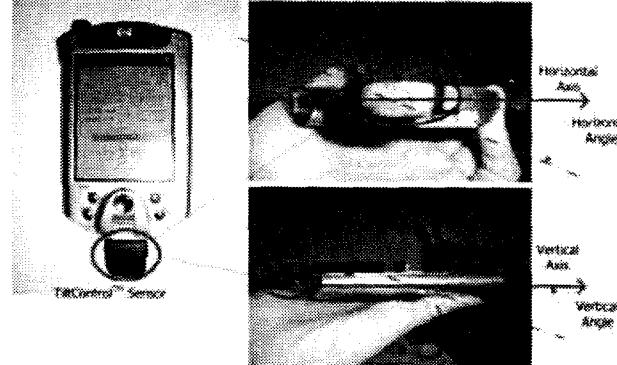


Figure 3: The TiltCONTROL Sensor

Experiment to Identify Camera Shaking Pattern

Tilt sensors can be used to detect camera shaking and automate the process of shaking artifact detection and removal. This is an ideal alternative to computationally intensive video analysis on a resource-poor mobile device. To determine the signature of camera shaking, an experiment was conducted to distinguish between excessive amount of shaking (e.g., resulting from putting the device in a pocket during walking) from moderate shaking that comes naturally with unstable hands when walking while filming. Our experiment is described below.

Data Acquisition: The TiltCONTROL sensor monitors vertical and horizontal tilt of the device throughout the experiment. A series of readings are recorded and analyzed to determine if camera shaking occurs. The sample rate of tilt sensing is set to be 200 milliseconds. The standard deviation of changes of device angles is computed for each sliding window of the most recent 10 readings.

Shaking Detection: Device shaking can be detected when changes in a device's tilt angles *oscillates* between two opposite directions. The intensity of shaking can be measured by calculating the *rate of change in device tilt angles* and the *oscillation rate*. Walking while holding a device by hands will create oscillations of smaller magnitude (see the middle graph of Figure 4). Walking with the device in a pocket will also create oscillations, but of larger magnitude (see the right graph of Figure 4). For the experimental setup, we measured three activities for each participant:

- (1) Holding the mobile device while sitting or standing still for 2 minutes (collecting 591 samples);
- (2) Holding the mobile device while walking for 2 minutes (collecting 591 samples); and
- (3) Putting the PDA in a pocket or a bag while walking for 2 minutes (collecting 591 samples).

Table 1: Standard deviations on the magnitude of oscillations and frequency of oscillations for three activities

Activities	Standard deviation on tilt angle degree changes		Frequency of oscillations (per second)	
	Horizontal	Vertical	Horizontal	Vertical
Sitting	2.62	3.00	1.36	0.76
Walking	5.27	7.13	1.89	1.97
Pocketing	64.72	75.96	1.73	1.85

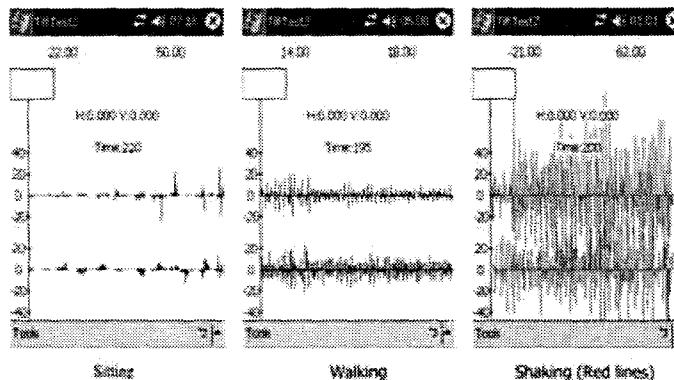


Figure 4: Measured oscillation magnitudes for three activities: (1) holding the mobile while sitting, (2) holding the device while walking, and (3) putting the device in a pocket. X-axis represents time. Y-axis represents the magnitude as change of degree per unit time.

Result: Based on empirical data shown in Table 1, we determine two conditions for excessive shaking: (1) the standard deviation of the tilt angles is larger than 20 (degrees) – it is calculated by 89.9% of actual shaking frames (externally observed) having higher standard deviation values than this threshold value, and (2) the frequency of oscillations in both directions exceeds 1.5 oscillations per second – again, it is calculated by 76.5% of actual shaking frames having higher value than this threshold value. In Figure 4, we depict a partial result of one participant’s experiment. We can see from this figure, under the normal case, that the standard deviation is small, and the vibration is moderate. Walking introduces constant vibration, but the standard deviation is below 20. When shaking, we can see that the standard deviation is high and the vibration is frequent. This pattern helps the system to detect camera shaking with a simple computation of standard deviation, which demonstrates how sensor measurements may assist in processing video content using simple computation.

5. USER INTERFACE DESIGN

The design of a mobile user interface needs to consider small screen size, inconvenient input methods, limited user attention, and limited user computing experience. A key design challenge is to understand the tradeoff between simplicity (ease-of-use, short learning curve, and reduced user effort) and *quality of edited production* (which provides a rich feature set but comes at a cost of increased user effort). In addition, the UI design needs to accommodate the system storage constraints on a mobile device.

The mProducer UI consists of two parts: location-based content management and keyframe-based editing. They are described in the following subsections.

5.1 Location-based Content Management

We have conducted an informal user study to find out the preferred manner, of casual users, to navigate or browse video clips. In general, there are two ways they mentally group clips: by *recording time* or by *recording location*. They reported that, in general, they prefer to navigate based on location instead of time. Users told us that they can make stronger mental associations between video clips and visual locations rather than times, i.e., they can better remember specific locations where they recorded video clips, rather than the specific times when they recorded video clips. We believe that the reason is location information is more visual (hence easier to remember and make associations), whereas time information is more abstract. With the help of the GPS receiver, we were able to automatically annotate each video clip with its recording location. This removed the need for a user to manually input the location meta-data. With location information, clips are organized and grouped by points on a map, rather than in directories for a file browser.

5.2 Keyframe-based Editing

There have been several applications that use keyframes extracted from video clips. One of these uses keyframes to expedite video browsing [5][10]. It has been shown that users can get a good understanding of video clip content by browsing only their keyframes [12]. We would expand on understanding to investigate keyframe editing, i.e., we would like to know if users can edit using only keyframes and still produce satisfactory quality for sharing personal experiences. We have performed a user study to investigate the effectiveness of keyframe-based editing, specifically:

- The reduction in user-perceived quality and whether the produced contents were acceptable to them;
- The reduction in user efforts or improvement in task performance;
- Keyframe-based editing’s effectiveness when combining with either a slideshow player³ or a storyboard player⁴.

If the editing quality drops only slightly and the task performance improves significantly, we can say that keyframe-based editing offers a good design trade-off for mobile computing environment.

5.3 User Study #1

The user study consists of testing the following three user interfaces:

³ A slideshow player displays an image to a user, waits a short period of time, and then displays the next image in a sequence, which may be random or ordered.

⁴ A storyboard player displays multiple still keyframe images at once, representing pivotal frames from a sequence, in order to understand a clip. The storyboard player differs from a slideshow player in that the storyboard player allows a user to see keyframes from adjacent shots the same time, whereas a slideshow player allows a user to see one keyframe (shot) at a time.

- **(UI-A):** Frame-by-frame editing with a video player (the scaled-down version of conventional desktop editing interface);
- **(UI-B):** Keyframe-only editing with slideshow player; and
- **(UI-C):** Keyframe-only editing with storyboard player.

Participants were asked to capture and edit video clips using each of three editing interfaces shown in Figure 5.

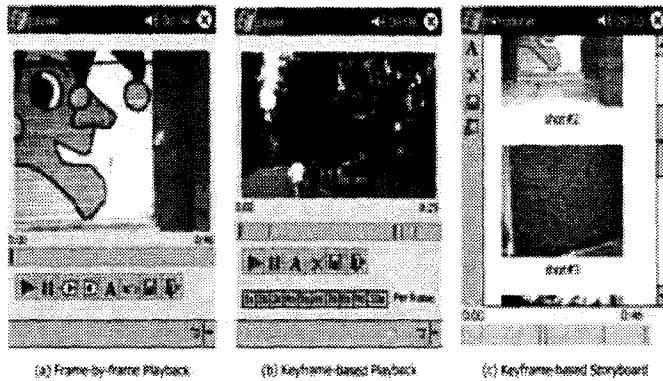


Figure 5: Screen shots for three editing user interface (frame-by-frame editing UI on the left, keyframe-only editing with slideshow player in the middle, and keyframe-only editing with storyboard player)

Independent Variables: The three editing interfaces detailed above.

Dependent Variables: Task performance measures the amount of time to complete editing tasks using a selected editing interface. Subjective satisfaction ranks the interfaces in terms of overall editing experience, the user's perception of quality of editing, ease of use, and ease of learning.

Participants: We randomly chose eleven participants (eight males and three females) on campus for this user study. Their ages range from 20 to 41 years, with a mean of 24. Three of them (all male) have previous experiences in using a PDA. Five of them (four male and one female) have previous experiences in using PC video editing tools. None of them had previous experience in using mobile video editing tool. All participants have used cell phones.

Procedures: The evaluation is consisted of four sessions: introduction/training, capturing video clips, editing video clips, and filling out a questionnaire as part of a face-to-face interview.

1. Each participant was asked to record a total of 6 minutes of video containing three 2-minute clips. Examples of content captured included scene-recording, self-introduction of people in a group, and a specific event.
2. The participants were asked to edit three clips, each using one of the three different editing interfaces. The editing task involved removing unwanted content from the raw video clips. We measured the length of time it took to complete each editing task for each participant. Note that the

assignment between clips and editing interfaces were chosen randomly to reduce the first clip bias⁵.

3. Each participant filled out a questionnaire with demographic information including age, sex, and experience with video editing tools. The questionnaire also asked each participant to rate the three editing interfaces in four terms defined in Table 2.

Results in Task Performance: We recorded the time each participant took to complete editing a two-minute video clip for each of three clips. The results are shown in Figure 7. The mean task completion time for each UI is: (UI-A) 4 minutes and 32 seconds, (UI-B) 3 minutes & 58 seconds, and (UI-C) 2 minutes and 48 seconds. Ten out of eleven participants completed the editing task fastest using (UI-C). All participants finished editing sooner using (UI-B) in comparison to (UI-A). The result shows that users can perform editing tasks *more* efficiently using a keyframe-only editing interface. In addition, the keyframe-only storyboard editing interface provided the best task completion time.

Based on our interviews with participants, they reported that the storyboard UI helped them by enabling them to see several keyframes at the same time. They could quickly identify which frames or shots they did not like and remove them. Some participants also mentioned that their problem with frame-by-frame editing was that it required uninterrupted, focused attention on the screen. However, many elements in the mobile

Table 2: Ratings on three editing interfaces

Questions (Rank three editing UIs)	
1	Perceived quality of editing
2	Ease-of-use
3	Ease of learning
4	Overall editing experience

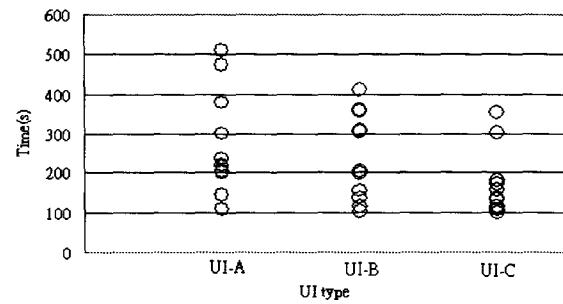


Figure 7: Task completion time

environment can be distracting and make it difficult for a user to maintain continuous attention for a long period of time. For examples, friends calling, people walking by, and surrounding noise can all temporarily distract user attention from the editing task. This makes frame-by-frame editing over a long clip difficult in a mobile environment.

Results in Subjective Satisfaction:

⁵ Participants may be least familiar with the first clip they recorded and might be less efficient in locating and removing unwanted portions of it.

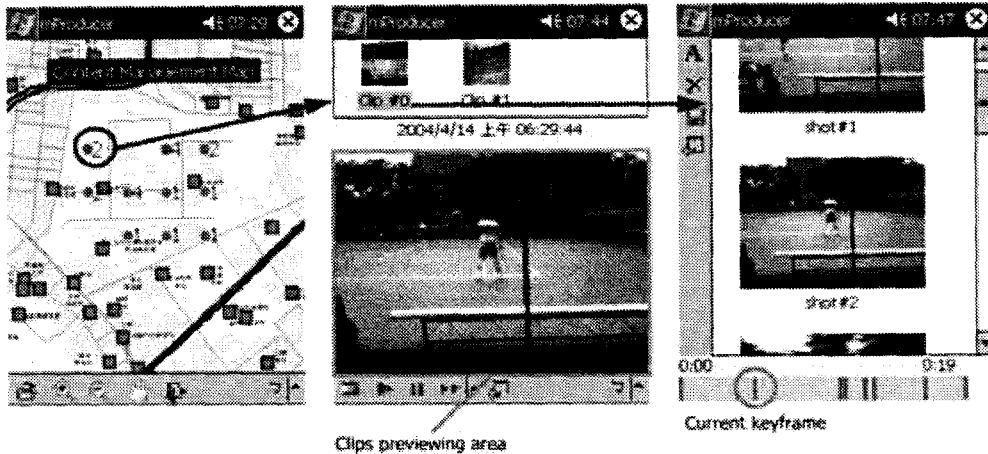


Figure 9: mProducer user interface showing the location-based content management (left screen), material pool (middle screen), and storyboard interface (right)

Participants answered the questions listed in Table 2. Their responses to the first three questions are shown in Figure 8. The results show that users rated keyframe-only storyboard editing as producing superior editing quality. Our explanation is that when using frame-by-frame editing, casual users are not willing to spend time to find good mark-in and mark-out boundary points for unwanted content. Because of this, they find our SBD algorithm can find better boundary points for both wanted and unwanted shots. The results also showed that users rated keyframe-only storyboard editing to have the best ease-of-use and ease-of-learning. We were told that the advantages of the keyframe-only storyboard interface were that (1) it allows users to quickly move among shots, which is useful during editing, and (2) it allows users to quickly delete unwanted shots by a single-click on the keyframes corresponding to these shots.

The results for overall experiences in the three editing interfaces showed that UI-C (keyframe + storyboard) was consistently selected as most satisfying from all participants (100%), and 64% (seven) of the participants found UI-B to be more satisfying to use than UI-A.

5.4 User Study #2

We conducted user study #2 to evaluate the overall experience of mProducer due to location-based content management interface and keyframe-only storyboard editing interface. The left screen of Figure 9 shows what a user sees when starting to edit video clips. On the map, dots are used to represent locations where contents were captured. Initially, we tried to use thumbnails instead of dots on map, but the PDA's small screen became cluttered with only a small number of video clips. Users can use the map interface to navigate (zoom in, zoom out, or move the map) and find clips to edit based on the location information. The middle screen of Figure 9 shows a *material pool* containing all clips captured at a specific location. The material pool screen is shown after the user clicks on a dot on the map. We provide keyframe previews for users to quickly decide which clip to edit. On the list of clips, one can see the time, date and the duration of the recorded content. The right screen of Figure 9 shows the keyframe-only editing UI.

Participants: We observed seven participants using mProducer to record video clips. Five were male and two were female. The ages

of users varied from 21 to 33 years old, with the average being 23.8 years. Three have had previous experiences using PDA, while all have used cell phones. Three had previous experiences with desktop PC video editing tools. One of them had previous experience with a mobile device's video editing tool. All were chosen randomly on campus.

Software and Hardware Equipment: Each participant was provided with mProducer running on an HP iPAQ 5450 mounted with a GPS receiver and a digital camera.

Procedure:

- (1) Participants were asked to shoot any type of footage they wanted. They were encouraged to walk around campus, and record what they found interesting. We asked them to record about 10 minutes of footage with any number of clip(s).
- (2) Participants used the editing component of mProducer immediately on the content they had produced. They were asked to edit two clips chosen randomly from the pool of clips they had recorded. During the editing sessions, participants were asked to "think aloud" in order to let us know their intentions and the cognitive process of using mProducer.
- (3) After the editing session, participants were asked to fill out a questionnaire and discuss their overall experiences using mProducer. The questionnaire included questions about demographical information, participants' previous experiences with mobile devices and video editing tools, their impression of the mProducer tool (before and after using it), their experiences of navigating among different clips and editing the two clips they chose, and any other improvements they thought we could make.

Result in Overall Experience: In general, participants' feedbacks were very positive. One of the participants described mProducer as "*a pretty cool tool to use.*" Another participant said that "*the keyframe-only storyboard is very helpful for me to delete contents that I do not like. Editing tools on desktop PCs should incorporate this feature too!*" "*Map based content management is very informative for choosing which clip to edit*", said the other.

All participants said that editing with a keyframe-only storyboard interface was fast and easy. Some of the participants mentioned

that the slideshow interface was better for getting a rough idea about the clip, while the storyboard interface was better for editing. Therefore, they suggested that the UI gives users the option to switch between these two interfaces. One participant suggested that we allow for location tracking of indoor recordings where the GPS receiver does not work. Some participants said that the content management map sometimes responds slowly⁶.

6. RELATED WORK

The Toshiba T-08 cell phone [13] is a commercial product that comes with its own video editing tool. Since it does not provide any uploading functionality, it only allows users to record 3 minutes of video clips at five frames per second on its 8 MB storage. Its UI is a smaller version of a frame-by-frame editing interface, but for a 3 minute video clip at a low frame rate, frame-by-frame editing is probably manageable. However, for long video clips recorded at a higher frame rate, a frame-by-frame editing interface would be difficult to use in a mobile environment.

Jokela presents an overview of the key opportunities and challenges in developing tools for authoring multimedia content in mobile environments [9]. However, no solutions were provided. Hitchcock [1] is a PC tool that uses keyframes to speed up editing of home videos. It displays keyframes in piles (based on color similarity of keyframes) for selection, and a storyboard to drag-and-drop keyframes (shots) according to the sequence of shots the user wants. Since mProducer runs on a PDA with a much smaller display, the idea of presenting shots in piles was not a workable solution. In addition, it is not possible to have both the keyframe presentation area and a storyboard on a small mobile screen at the same time.

7. CONCLUSION AND FUTURE WORK

We describe our design, implementation, and evaluation of a mobile authoring tool called mProducer that enables a mobile user to capture and edit personal experiences from a mobile device anytime, anywhere. MProducer addresses the challenges of both limited system resources and user interface constraints on a mobile device.

We have designed the Storage Constrained Uploading (SCU) algorithm, which uploads potentially large multimedia contents to servers, in order to alleviate the problem of limited storage on a mobile device. A GPS receiver was added to a mobile device to record location information for each piece of content captured by a user, and provide a map-based content management interface to enable easy, intuitive navigation from a small mobile screen. We incorporated a tilt sensor on a mobile device to automate the detection and removal of blurry frames resulting from excessive amount of shaking. This sensor-based solution requires small processing overhead, and is considered a reasonable alternative to computationally-expensive image processing techniques to detect shaking artifacts. We have designed a keyframe-only editing interface, and conducted user studies to evaluate the overall user editing experience (ease-of-use and learning curve), task performance time, and quality of the edited product. Overall, users found mProducer to be both easy and fun to use on a mobile device.

⁶ This was due to the limited computation power on the PDA

Since cell phones are more popular than PDAs, we are in the process of porting mProducer onto a cell phone platform. We are interested in finding out how well our UI design would work on a cell phone with an even smaller screen than a PDA.

Editing video clips is more meaningful if they can be shared with other people who are interested in viewing them. Our future work will exploit new methods to conveniently disseminate personal experience recordings.

ACKNOWLEDGMENTS

This research was funded by the NSC under NSC funding 92-2218-E-002-036

8. REFERENCES

- [1] A. Gergenoh *et al.* Home video editing made easy: Balancing automation and user control. In Human-Computer Interaction: INTERACT, 2001.
- [2] Chao-Ming Teng, Chon-In Wu, and Hao-hua Chu. mProducer: authoring multimedia personal experiences on mobile phones. In Int'l Conf. on Multimedia and Expo (ICME), 2004.
- [3] R. Sarvas *et al.* Metadata creation system for mobile images. In Int'l Conf. on Mobile Systems, Applications and Services (MobiSys), Jun 2004.
- [4] F. C. Li *et al.* Browsing digital video. In CHI 2000, pages 169–176, Apr 2000.
- [5] U. Gargi and R. Kasturi. An evaluation of color histogram based methods in video indexing. In International Workshop on Image Databases and Multimedia Search, 1996.
- [6] G. Davenport, T. Aguirre Smith, and N. Pincever. Cinematic primitives for multimedia. IEEE Computer Graphics and Applications, 67–74, July 1991.
- [7] H. Zhang *et al.* Video parsing, retrieval and browsing: An integrated and content-based solution. In Proc. of ACM Multimedia, 1995.
- [8] T. Jokela. Authoring tools for mobile multimedia content. In IEEE Int'l Conf. on Multimedia and Expo (ICME), pages 637–640, 2003.
- [9] A. Komlodi and G. Marchionini. Key frame preview techniques for video browsing. In Proc. of ACM International Conference on Digital Libraries, 118–125, 1998.
- [10] C. Snoek and M. Worring. Multimodal video indexing: a review of the state-of-the-art. Multimedia Tools and Applications, 2004 (In Press).
- [11] T. Tse *et al.* Dynamic key frame presentation techniques for augmenting video browsing. In Proc. of Advanced Visual Interface (AVI), pages 185–194, 1998.
- [12] Vodafone, Japan. <http://www.vodafone.jp/english/>.
- [13] W. Yan and M. S Kankanhalli. Detection and removal of lighting & shaking artifacts in home videos. In Proc. of ACM Multimedia, 107–116, Dec 2002.
- [14] W. Li. Overview of fine granularity scalability in MPEG-4 video standard. In IEEE Transactions on Circuits and Systems for Video Technology, Vol. 11, No. 3, March 2001.
- [15] H. Zhang *et al.* Video parsing, retrieval and browsing: an integrated and content-based solution. In Proc. of ACM Multimedia, 1995.

Mobile Kärpät – A Case Study in Wireless Personal Area Networking

Timo Ojala

Jani Korhonen

Tiia Sutinen

Pekka Parhi

Lauri Aalto

MediaTeam Oulu
University of Oulu

P.O.Box 4500, FI-90014 University of Oulu, Finland
{firstname.lastname}@ee.oulu.fi

ABSTRACT

The advanced smartphones entering the mass market are capable of playing audio and video files back, which paves the way for new types of rich mobile multimedia services. However, these services impose high data rate requirements on the wireless link, which can not necessarily be satisfied with the current mobile phone networks. This can be compensated with a wireless personal area network based for example on Bluetooth connectivity, or with a wireless local area network. This paper presents a case study demonstrating complementary distribution of static and dynamic multimedia content with Bluetooth equipped WPAN service points, a wireless local area network and mobile phone networks. The empirical evaluation conducted in the real environment of use shows that the proposed service is meaningful with commercial potential.

Categories and Subject Descriptors

H.5.1 [Information Interfaces and Presentation (e.g., HCI)]:
Multimedia Information Systems - Evaluation/methodology,
Video

H.5.2 [Information Interfaces and Presentation (e.g., HCI)]:
User Interface - Evaluation/methodology, Prototyping, User-centered design

General Terms

Design, Experimentation, Human Factors

Keywords

Mobile Multimedia, Empirical Evaluation, Usability

1. INTRODUCTION

A peculiar trend in the development of the technological capabilities of the mobile handhelds (phones and PDAs) has taken place over the past few years. While the computing, memory and display capabilities have been steadily increasing, just like in desktop computing, the data rates of the ubiquitous wireless connectivity have stayed around 10 to 40 kbps. With the ubiquitous wireless connectivity we refer to the wireless link that

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MUM 2004, October 27-29, 2004 College Park, Maryland, USA.
Copyright 2004 ACM 1-58113-981-0 /04/10... \$5.00

is available independent of place and time, e.g. in form of an operator hosted mobile phone GSM/GPRS network.

In theory, it is possible to increase the data rate of the mobile phone network, e.g. by using several simultaneous GSM channels, but the associated cost renders it unusable scenario for most end-users. EDGE and UMTS are bringing about an improvement to the capacity of ubiquitous wireless access [5]. However, it will still take several years before we can transfer data at rates of hundreds of kbps to the mobile phones dominating the consumer markets.

The limitations in the data rates of the mobile phone networks are partially compensated by the other forms of wireless interfaces embedded in the mobile phones, e.g. Bluetooth and IrDA, which typically provide data rates exceeding 100 kbps. Both can be employed in realizing a WPAN (Wireless Personal Area Network) [8] for interconnecting devices centered around an individual person's workspace, in which the connections are wireless. The typical spatial range of a WPAN realized with Bluetooth technology is about 10 meters.

Further, some new mobile phones entering the consumer market are equipped with wireless local area network (IEEE 802.11, abbreviated WLAN or Wi-Fi [7]) connectivity, e.g. Nokia 9500, which extends the spatial range of the high data rate wireless connectivity to about 100 meters.

Thanks to the increased computing resources of the mobile handheld devices, the supply of rich multimedia content targeted to these devices is increasing rapidly. This includes both informational and entertaining content such as brochures, tourist guides, games, micro movies, songs, etc.

However, downloading rich multimedia content imposes high data rate requirements on the wireless link, which can not necessarily be satisfied with the current mobile phone networks. This calls for employing WPAN (or WLAN) connectivity in distributing multimedia content.

Another major challenge in the domain of mobile handheld devices is human computer interaction (HCI). The usability of versatile and complex services can be seriously hampered by the rather limited input and output capabilities of a handheld device. A typical example is browsing the Internet, which is very convenient with the full-blown keyboard and the large screen of a desktop computer, but can be really awkward with the 12-key keyboard and the small display of a mobile phone.

To a certain extent the mobile access of a service, e.g. browsing

the Internet, can be enhanced with adaptation [16], filtering [11] and personalization [1], which may need extra resources in the network and/or the mobile device.

Further, a service can be provisioned in a context-aware manner, taking into account any information that can be used to describe the situation of a person, place or object that has significance to the interaction between user and the service [4]. The physical location of the user is one of the most obvious and useful types of context information. Combining the location with a suitable WPAN technology providing sufficient data rate, so that the WPAN's physical access point is associated with the context, we can offer rich multimedia services to mobile users of handheld devices in a user-friendlier manner.

Another important context attribute is the time dependency of the creation and consumption of content. Whereas static content can be created well beforehand its consumption, dynamic content is produced in real-time during the use of the service. This difference may have a great impact on the selection of the technology that can be employed for distributing the content.

This paper presents the Mobile Kärpät case study, where rich multimedia content is offered to users of mobile phones in two complementary manners in a context-aware fashion. Beforehand created static content is made available with a commercial Bluetooth based WPAN service called iJack. The static content contains hyperlinks to dynamic content produced in real-time during the use of the service, which is downloaded over commercial mobile phone network.

The motivation in employing the WPAN service in distributing the static content is two-fold. First, WPAN provides much higher data rate for download, which results in better user experience. Second, engaging WPAN service reduces the load on the mobile phone network hence higher capacity is available for the consumption of the dynamic content.

Further, for comparison purposes a wireless local area network based on IEEE 802.11a/b standard is employed to provide wireless access to users of PDA's.

The novel contribution of this paper is reporting the design, implementation and empirical evaluation of a rich multimedia service in the real environment of use. Based on a thorough user study, a genuine service provider and true end users are involved for the purpose of assessing consumer behavior and user experience, in addition to empirical performance characterization of the service.

This paper is organized as follows. Section 2 introduces the iJack WPAN service, section 3 presents the Mobile Kärpät case study and section 4 concludes the paper.

Related literature on providing mobile multimedia in sports events or as a WPAN service is rather scarce. Pham and Wong [17] discuss the different features and requirements set for a mobile device for the purpose of realizing mobile multimedia applications.

Song *et al.* [19] present an empirical study of user behavior in case of mobile streaming. The quality of the video stream rendered on the mobile device is altered by changing the frame rate, image resolution and bandwidth. Based on the QoS experienced by the test users the paper concludes that in case of

devices having low resolution displays, increasing frame rate does not increase the end user QoS. Consequently, with that kind of devices it is beneficial to divide the limited bandwidth over a number of users. The lowest bandwidth considered in the study is 384 kbps, hence the results can not be directly generalized to GPRS networks providing about 40 kbps.

Boll and Westermann [3] present a peer-to-peer system for distributing personalized multimedia events, such as sports related, globally to mobile devices.

Hunter [6] discusses telecommunications delivery in the Sydney 2000 Olympic Games. Considering the enormous scale of events such as Olympic Games, local WPAN solutions as the one described in this paper could be employed to provide consumer services, which would reduce the strain on ubiquitous telecommunication network.

Vilovic and Zovko-Cihlar [21] demonstrate that a Bluetooth link can be used for streaming video, which could be an interesting enhancement to our WPAN service.

2. iJACK WPAN SERVICE

TeliaSonera's iJack service [9] is based on WideRay's Jack Service Point [23], which is a solution for delivering local content to mobile devices over Bluetooth or IrDA. Jacks have been deployed in real end user environment, such as shopping malls and sports facilities.

Figure 1 illustrates the overall architecture of the iJack service. The user can download files from an iJack service point with a mobile device over Bluetooth connection. iJack service point acts as a master of the Bluetooth piconet, which according to the Bluetooth specification can serve at most seven slaves simultaneously.

The files are uploaded to the iJack service point over GPRS. Thus, content is uploaded once over a narrowband ubiquitous mobile phone network to make it available for multiple downloads over a much faster WPAN link. One major advantage of the iJack architecture is that it is a completely wireless solution, which can be deployed in a straightforward manner.

The iJack service comes with a service management system, which includes a web user interface for content providers. The business model is so that the content provider, e.g. a store wishing to advertise its goods, leases the service from TeliaSonera, i.e. the iJack service points and the service management system. The end-user typically can download the content free of charge.

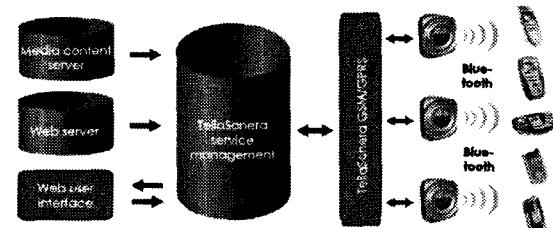


Figure 1. Architectural overview of TeliaSonera's iJack WPAN service.

From the end-user's point of view, a typical usage sequence of the iJack service goes as follows:

1. User enters the about 10 meter range of an iJack service point (user becomes aware of this with some visual aid such as a poster or a stand) and stays there for the duration of the download.
2. The user initiates a Bluetooth connection with the iJack service point. This is accomplished e.g. by sending a dummy contact information message to the iJack service point. Technically, iJack service point could also initiate communication, but it is not allowed in Finland.
3. If user does not have an iJack browser ready in his mobile device, he is asked to download one from the iJack. After download, the iJack browser starts up, connects to the iJack service point and displays the files available for download.
4. With the iJack browser the user can download files from the iJack service point to the mobile device.
5. User can stray away from the range of the iJack service point, having the files stored in the mobile device.

The iJack browser is a dedicated application needed for communicating with an iJack service point. The browser needs to be downloaded only once, i.e. in future encounters with an iJack service point the user just needs to fire up the browser to see which files are available for download. At the time of conducting this work the browser was available only for Nokia Series 60 mobile phones, Pocket PC and Palm.

3. CASE STUDY: MOBILE KÄRPÄT

Oulun Kärpät [15] is one of leading clubs in the Finnish ice hockey league, winning the league championship in the 2003-04 season. Oulun Kärpät relies in professional production of digital content (news, match reports, etc.), which is made available to the general public via the club's popular web site attracting tens of thousands of daily visitors at times. The club also offers limited WAP pages for mobile users.

The motivation of the Mobile Kärpät case study was to empirically assess, whether the match experience of the spectators at the arena could be enhanced by providing them access to rich match related multimedia content during the match. The reference point in terms of match related content was a match program printed several days before the match.

The implementation of the case study was perfected to the interactive system design process [12], extending from a versatile user study to a thorough empirical evaluation of the prototype in the real environment of use.

3.1 User study

The goal of the user study was to identify the potential users of the service, and the tasks they would like to carry out with the service. The user study comprised of contextual inquiries and interviews at a live ice hockey match, and two online surveys at the hockey club's web site.

3.1.1 Contextual inquiry and interviews

Beyer and Holzplatt [2] have introduced the method called contextual inquiry, which aims at extracting information of users in a particular context of use. The basic idea is to observe the user

in a real world situation and ask questions if needed. The observation can be augmented with an interview before and after the observation. Interviews can be productive by giving the user an opportunity to express feelings, for eliciting information about user preferences, impressions and attitudes, which may not necessarily come forward with plain observation.

The contextual observation and interviews took place in a single regular season hockey match in February 2004. Before the observation few preliminary questions were asked, for the purpose of surveying the subjects' background, interest in ice hockey and information retrieval channels. Additional questions were presented during the observation. Finally, after the observation the subjects were again interviewed on issues concerning adoption of new mobile services, preferred content types and pricing models.

In total 12 subjects were both observed and interviewed and 8 additional subjects were interviewed. On the basis of open structured questions the most desired content was: 1. statistics (10 out of 20 interviewees), 2. slow motion pictures (8), 3. background information (7), and 4. information from other matches (5). On the basis of the structured questions (options given) the most desired content was: 1. video clips and slow motion pictures from the current match (20 out of 20 interviewees), 2. players' statistics (13), 3. league table (9), and 4. video clips and slow motion pictures from previous matches (5).

The structured questions were presented after the open structured questions, during which it became apparent that not all subjects were aware of the possibility of viewing video clips with modern mobile phones. When video clips were offered in the structured questions, all subjects designated it as the most wanted content.

3.1.2 Online survey #1: Narrative open-ended questions

The first online survey comprised of five narrative open-ended questions, hence the respondent was engaged as the designer of the service. The photo of an upcoming multimedia device was employed as a stimulant, together with a textual description of the lavish technical capabilities of the device. The general public was motivated to reply with quality answers by a raffle, where one team jersey was drawn among all participants and another was issued to the respondent providing "the best story" as selected by the researchers. The survey was available at the hockey club's web site for 15 days, drawing 734 respondents.

The survey comprised of five open-ended questions, which are listed below, together with a selected representative answer

Question #1: "What would you like to do with the device during the match?"

"Look at the goals again, of course. During the intermissions I could check the statistics."

Question #2: "What would you like to save from the match to the device and take with you when leaving the arena?"

"Goals, screwups and fights."

"The smell and taste of hot sausages." (Now, this is a great challenge to mobile phone manufacturers ☺)

Question #3: "What would you like to send to your friend's device?"

"1. Comments from the match, 2. Meeting place at the next intermission."

"Fans of a competing team would get a summary of the goals of our team and the screwups of their team."

Question #4: "Why would the functions provided by the device be useful to you?"

"This device would be a dream come true ..."

"It would help me keep in touch with the match."

Question #5: "Free form description of the use of the device."

3.1.3 Online survey #2: "Traditional" questionnaire

The second online survey was "traditional" in the sense that it comprised of multiple choice questions grouped in five categories: demographic information, respondent's interest in ice hockey in general, the basic capabilities of the respondent's mobile phone, respondent's interest in different types of hockey related content and the respondent's assessment of the relative importance of various factors in terms of user experience.

The second survey was available at the hockey club's web site for 11 days. Of the 645 respondents 77% were male, 23% female, and 84% of them were at most 34 years old. The three most important sources of ice hockey related information were TV (34%), the web site of the hockey team (26%) and the web site of the hockey league (21%). The two by far most important types of information wanted by the respondents were news (79%) and match reports (71%).

The service can be used with a Nokia Series 60 phone, which has GPRS, Bluetooth and XHTML browser. The occurrence of these capabilities in the mobile phones of the respondents is summarized in Table 1. Only 3.5% of the respondents reported to have a Nokia Series 60 phone.

Table 1. Capabilities of the mobile phones of the respondents of the second online survey.

Capability	Have	Do not have	Could not tell
GPRS	53%	38%	9%
Bluetooth	20%	63%	17%
browser	37%	52%	11%

When asked to assess on scale 1 (totally disagree) to 7 (totally agree) their interest in downloading different types of content with a mobile phone, the respondents ranked highest the following five:

- real-time statistics of the ongoing match (5.29)
- real-time info of other ongoing matches (5.19)
- videos and replays of ongoing match (5.09)
- team roster and lineups in today's match (5.07)
- league table (5.03)

When asked to assess on the same scale the relative importance of various factors in terms of user experience, following five were deemed most important:

- the desired info is found easily and quickly (5.86)
- the service is reliable (5.85)
- the service is easy to use (5.82)
- downloading times are short (5.80)
- the system responds quickly to commands (5.79)

3.2 Implementation of the service

The main service was provided as a set of XHTML pages, which after downloading could be browsed locally without any server connection. Further, a number of prerecorded videos containing pre-match comments from the head coach and selected players were available as separate Real Media video files. This way the size of an individual file remained below 1 MB, which ensured reasonable download times.

The XHTML pages contained the following components: news, the match of the day (subsections: lineups of the teams, match preview, online match report available over GPRS, previous matchups between the teams), team roster (subsections: online player videos available over GPRS, goalkeepers' player cards, defenders' player cards, forwards' player cards), playoff fixtures and Kärpät Fanpack (description of Kärpät related fan material available).

The online match report contained a hyperlink to a web page with dynamic content created during the match and another hyperlink to the online scoring report of all ongoing matches provided by the hockey league. Assorted screenshots of the XHTML pages are shown in Figure 2.

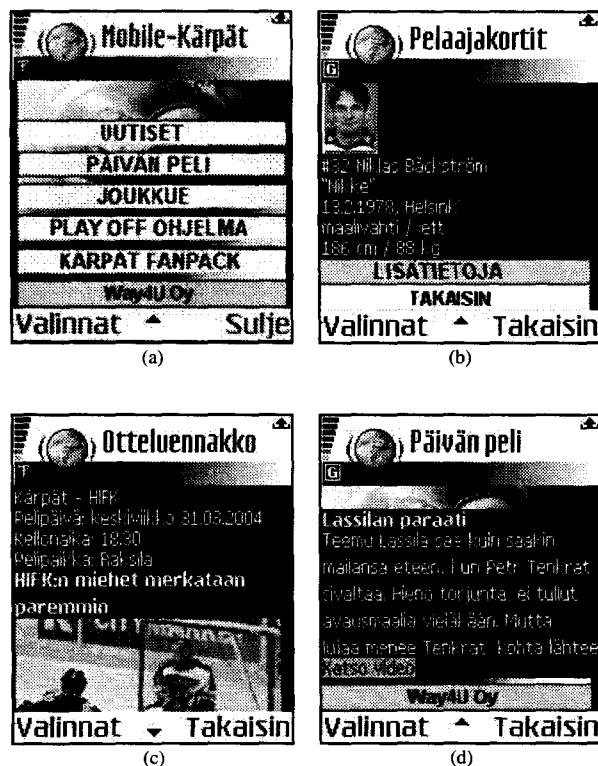


Figure 2. Screenshots: (a) main menu of the Mobile Kärpät package; (b) a player card; (c) an excerpt of the match preview; (d) an incident in the online match report, where highlighted "Katso video" link triggers GPRS streaming of the video clip related to the incident.

The service was provided in two different ways. For mobile phone users five iJack service points placed in lighted stands (see Figure 3(b)) were sited around the hockey arena, three at the main hallways accessible by the general public, and two at the VIP premises of restricted access. The stand contained a colorful advertisement of the service and usage instructions. The XHTML pages were packaged into a single .sis (Symbian Installation System) file of roughly 500 kB in size. The XHTML pages and the .sis file were generated separately for each match and uploaded into the iJacks before the match, together with the video files. Mobile phone users accessed the online components of the service using GPRS connection.

For PDA users the service was provided over a wireless local area network (IEEE 802.11a/b) realized with two Cisco 1200 series access points: the other was placed in the ceiling of the arena covering the stands, the other in the VIP premises. The XHTML pages and the video files were placed on a local web server.

According to the user study the most desired content was video footage from the ongoing match. To a certain extent this can be explained by the fact the hockey arena does not feature a large screen for showing replays of goals and other interesting incidents.

Video clips of the ongoing match were produced with an in-house video capture server, which buffered the incoming video feed so that clips could be generated retroactively, spanning also into the past at any given point of time. The server provided a simple graphical interface for configuring clip lengths and for initiating generation of different types of clips ('goal', 'save', 'tackle', 'penalty', 'skirmish' and 'other'). When the human operator pressed for example the 'tackle' button, the server generated a video clip spanning, relative to the time when the button was pressed, from the desired number of seconds into the past to the desired number of seconds into the future.

The video capture server automatically produced two versions of each clip. For mobile phones the data rate of 20 kbps was employed so that the resulting clip would fit into two GPRS time slots when streamed into the phone. The video files (50-70 kB in size) were automatically uploaded to a streaming server, where they were manually inserted into the online match report, together with a textual description of the incident (Figure 2(d)). For PDA's a higher data rate of 100 kbps was used and the resulting files (250-350 kB in size) were automatically uploaded to the local web server for HTTP download.

3.3 Empirical evaluation

The Mobile Kärpät service was empirically evaluated in the home playoffs matches of ice hockey team Oulun Kärpät in March 2004, which drew from 6000 to 7600 spectators each. Researchers were present in the matches, being on call at the iJack service points, guiding users, loaning out mobile phones and PDA's, and collecting feedback. A raffle was set up, to stimulate the users to try the new service and give feedback.

Figure 3 illustrates the different user groups and data collection methods employed in the empirical evaluation. Group 1 refers to those spectators who owned a suitable Nokia Series 60 phone and could use the service at their own will. Group 2 subjects were recruited at the matches by the researchers, from whom they were able to loan a suitable phone. Group 3 comprised of eight self-

selected long-term 'power users', who had expressed their willingness to participate in the test, discussions and observations. They were a very interesting group also because six of them used the service with both a mobile phone and a PDA.

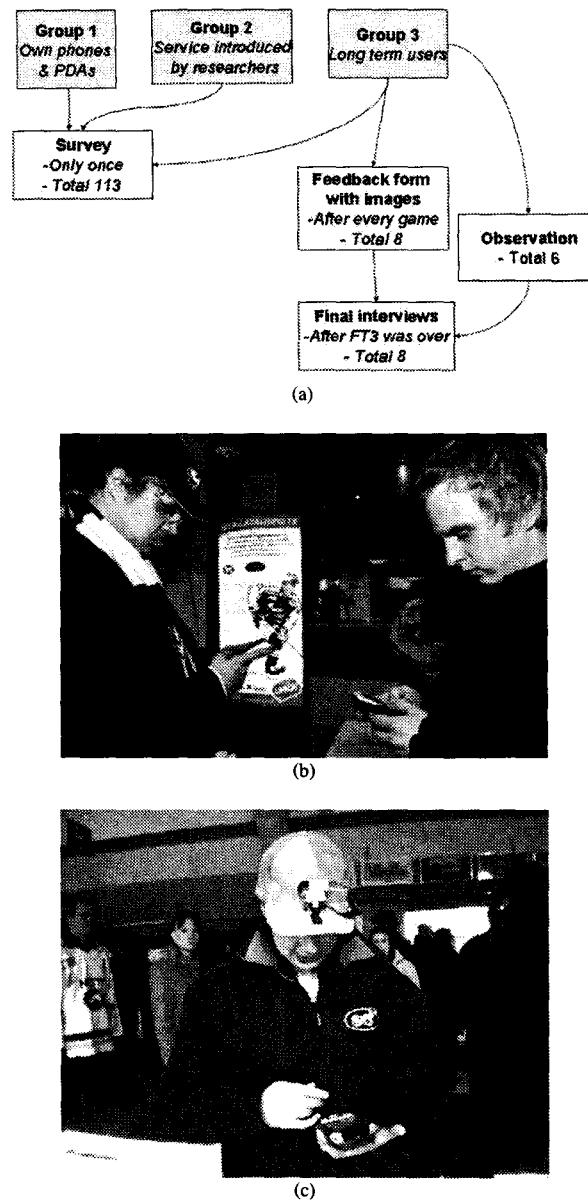


Figure 3. (a) User groups and data collection methods employed in the empirical evaluation; (b) two fans downloading the service from an iJack service point under the careful supervision of a researcher standing behind; (c) a test user is wearing the "helmet cam" recording her usage of the service.

Data was collected primarily with a questionnaire, which attracted respondents from each of the three user groups, and logging at the server side. Group 3 subjects were also interviewed and observed: Figure 3(c) shows a test user wearing the “helmet cam”, which captured the user’s interaction with the mobile device.

3.3.1 Quantitative analysis

In this section we study the server logs to assess the amounts of data downloaded at the first three playoff matches. We are particularly interested in the amount of static content (files uploaded to the iJacks before the match) downloaded from the iJack service points, which are shown in Table 2.

Table 2. Downloading of static content from the iJack service points over Bluetooth.

Match	Files downloaded	Amount of data (MB)
#1	262	142
#2	208	114
#3	136	75

The mobile phone users also had access to the online report and its video clips available at the streaming server. Table 3 shows the number of files downloaded and the corresponding amount of data.

We see that on average 110 MB of static data was downloaded during a match. We have to note that a vast majority of the downloading of static content took place during a thirty minute span before the start of the match, when spectators entered the arena and updated their Mobile Kärpät service.

Table 3. Downloading of dynamic online content from the streaming server over GPRS.

Match	Files downloaded	Amount of data (kB)
#1	87	592
#2	257	1271
#3	190	757

The amount of data downloaded over GPRS may appear very small. However, we have to keep in mind that in the matches 6000-7000 spectators were packed on a very small area covered by a single cell, many of them using their mobile phones, especially during the intermissions. The empirical observation was that during the intermissions the GPRS data connection was practically nonfunctional due to the heavy network load, regardless of the operator. We do not have access to the information how the operators had divided the network capacity between voice and data traffic.

For comparison purposes, Table 4 shows theoretical maximum capabilities of different mobile phone network technologies per base station per an hour as derived from [5][10][13][24]. We see that EDGE and especially UMTS bring about a welcome improvement in the overall transmission capacity with respect to GPRS. However, even these theoretical maximum capabilities are insufficient for providing rich mobile multimedia services for large user populations. Therefore, we have to look for complementary solutions such as the iJack WPAN service employed in this case study. It is trivial to show that via simple

replication this type of a WPAN service provides much better scalability than mobile phone network.

Table 4. Theoretical maximum capabilities of different mobile phone network technologies per base station per an hour [5][10][13][24].

Technology	Number of simultaneous users	Max . data rate per user (kbps)	Max. total data per an hour (MB)
GPRS	498	3,65	6548
EDGE	498	13,27	23784
UMTS	498	51,45	92242

3.3.2 Qualitative analysis

Next, we present a brief qualitative analysis of the questionnaire data obtained from the 113 respondents. The charts in Figure 4 visualize the test users’ assessment on scale 1 (totally disagree) to 7 (totally agree) of assorted claims addressing selected aspects of the service.

Most respondents found the service useful (Figure 4(a)). Also, despite of the frequent unavailability of the GPRS connection the majority of respondents thought that the service provided up-to-date information of the match (Figure 4(b)). Most of the respondents regarded the installation of the iJack browser easy (Figure 4(c)), although it was apparently the most difficult technical step in the mobilization of the service.

Having installed the browser the users had a bit easier time with downloading the content (Figure 4(d)), although some respondents found the downloading slow (Figure 4(e)). Most users had no trouble viewing the content (Figure 4(f)) and didn’t feel insecure using the service (Figure 4(g)). Finally, respondents regarded using the service fun (Figure 4(h)), which is an important prerequisite for the possible commercial deployment of the service.

A number of interesting observations were made during the user evaluation. For example, surprisingly many test users, including also tech types, had hard time understanding that they needed to stay nearby the iJack service points to download files over Bluetooth. Several subjects had trouble making a distinction between static and online content. Further, many subjects assumed that the content would be automatically updated when they entered the hockey arena.

Interesting trends in the usage and meaningfulness of different types of content were observed among long-term users. While dynamic content such as the match of the day and news remained as the most used and highest ranked content throughout the experiment, static content such as player videos, player cards and playoffs fixtures faded on both counts.

A pleasant observation was that the service and its content developed into a tool for social interaction. Video clips were shown to fellow spectators in the stands, which in turn spurred intense discussions.

Generally, following positive user experiences were observed: people like learning new things (“I knew how to do it! It was easy even for me. I normally can’t use anything like that!”), many subjects enjoyed trying out the new service and modern mobile

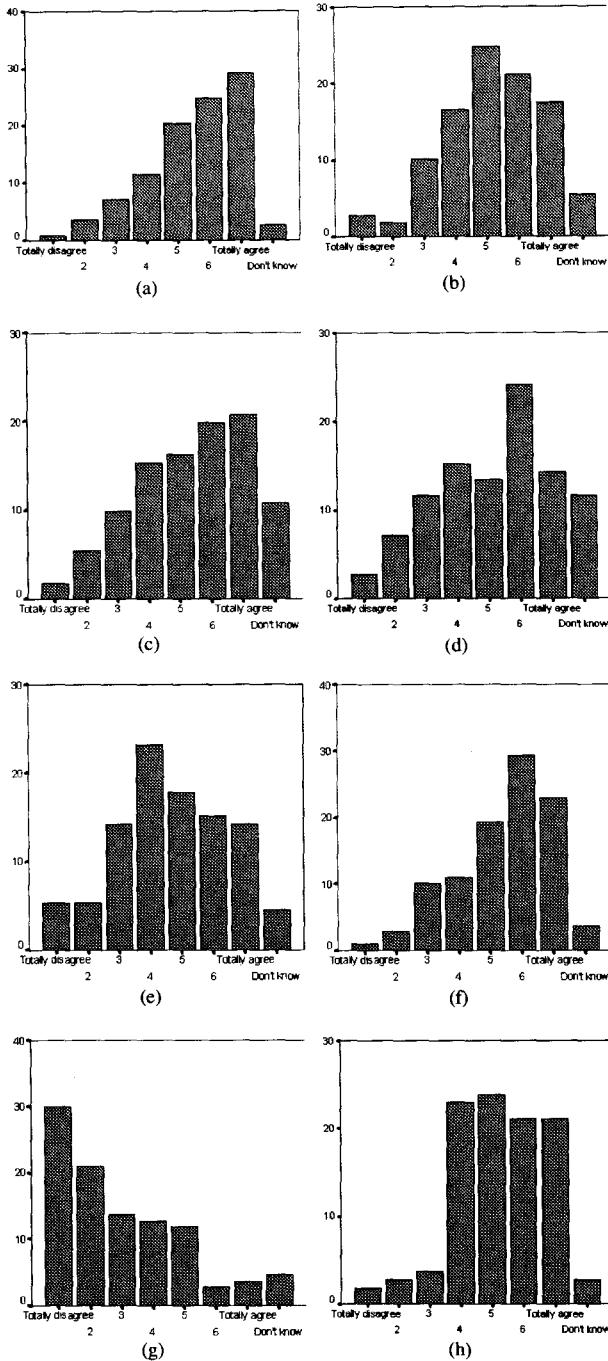


Figure 4. Statistics of the questionnaire data (N=113): (a) in my opinion this is a useful service; (b) with this service I can quickly get information of the match; (c) installing the browser was easy; (d) installing the content was easy; (e) downloading the content was quick; (f) viewing the content was easy; (g) I felt insecure using the service; (h) using the service was fun.

devices ("It was fun playing with this phone"), usefulness of the service ("It's a great time killing application for intermissions") and satisfaction with the content ("Now I can see the unclear situations I couldn't see from my seat").

Similarly, a number of negative user experiences were accumulated: technical difficulties, uncertainty, dissatisfaction with the content ("I can't distinguish the color of the shirts of the players and can only guess where the puck is in the video"), dissatisfaction with the availability of the content ("GPRS connection was slow or did not work at all during intermissions") and lack of a suitable phone ("I do have Bluetooth in my mobile phone. What do you mean? The service does not work in my phone?").

When inquired about their willingness to pay for the service, the long term users were not inclined to pay at the stadium. However, they were much more favorable towards payment, if the service were provided anywhere outside the arena, e.g. in form of video clips of the highlights of the match.

None of the long-term users were interested in buying a new phone just for the sake of using the new service. However, many test users reported they were going to consider a suitable model when renewal would become topical.

4. CONCLUSION

This paper presented a case study of a service, which is based on wireless distribution of rich mobile multimedia content. Beforehand generated static content was uploaded to a WPAN service point, from which the users could download it over broadband Bluetooth connection. Dynamic content produced in real-time during the use of the service was downloaded over the mobile phone network.

The rigid relationship between the nature of the content (static vs dynamic) and the distribution technology (WPAN/Bluetooth vs GPRS) is easy to understand. The slow GPRS backbone connection of the WPAN service point effectively prevents using it for distributing near real-time content. It would become more prominent, if the WPAN service point were equipped with a faster backbone connection, e.g. WLAN.

The service was evaluated with a large-scale field trial conducted with true end users in the real environment of use. The empirical observation was that the GPRS data network was badly clogged during the matches, especially during the intermissions. However, despite this and other assorted technical problems most test users found the service useful and fun to use.

5. ACKNOWLEDGMENT

The financial support of the National Technology Agency of Finland and the Graduate School on Electronics, Telecommunication and Automation is gratefully acknowledged.

The Mobile Kärpät case study reported in this section was a joint effort of the Oulun Kärpät ice hockey club [15], TeliaSonera Plc. [19] (provider of the iJack service), Nokia Plc. [14] (mobile phone manufacturer) and Way4U Ltd. [21] (publisher of Oulun Kärpät's web site) and University of Oulu's Rotuaari research project [18] employing the authors.

6. REFERENCES

- [1] Anderson C, Domingos P & Weld D (2001) Personalizing Web Sites for Mobile Users. Proc. Tenth International World Wide Web Conference, Hong Kong, 565-575.
- [2] Beyer H & Holtzblatt K (1998) Contextual Design. Morgan Kaufmann Publishers.
- [3] Boll S & Westermann U (2003) Meeting experience: MediaEther: an event space for context-aware multimedia experiences. Proc. ACM SIGMM 2003 Workshop on Experiential Telepresence, Berkeley, CA, 21-30.
- [4] Dey AK (2001) Understanding and Using Context. Personal and Ubiquitous Computing 5(1):4-7.
- [5] Halonen T, Romere J & Melero J (2002) GSM, GPRS and EDGE Performance - Evolution Towards 3G/UMTS. Wiley.
- [6] Hunter J (2001) Telecommunications delivery in the Sydney 2000 Olympic Games. IEEE Communications Magazine 39(7):86-92.
- [7] IEEE Working Group for Wireless LANs (2004) <http://grouper.ieee.org/groups/802/11/>.
- [8] IEEE Working Group for WPAN (2004) <http://www.ieee802.org/15/>.
- [9] iJack (2004) <http://www.i-jack.com>.
- [10] Korhonen J, Aalto O, Gurtov A & Lamanen H (2001) Measured Performance of GSM, HSCSD and GPRS. Proc. IEEE International Conference on Communications, Helsinki, Finland, 5:1330-1334.
- [11] Margaritis M & Polyzos GC (2000) On The Application of Continuous Media Filters Over Wireless Networks. Proc. IEEE Multimedia and Expo, New York, NY, 3:1241-1244.
- [12] Newman W & Lamming M (1995) Interactive System Design. Addison-Wesley.
- [13] Ni S & Haggman S-G (1999) GPRS Performance Estimation in GSM Circuit Switched Services and GPRS Shared Resource Systems. Proc. Wireless Communications and Networking Conference, New Orleans, LA, 3:1417-1421.
- [14] Nokia Plc. (2004) <http://www.nokia.com>.
- [15] Oulun Kärpät (2004) <http://www.oulunkarpat.fi>.
- [16] Pashtan A, Kollipara S & Pearce M (2003) Adapting Content for Wireless Web Services. IEEE Internet Computing 7(5):79-85.
- [17] Pham B & Wong (2004) Computer human interface: Handheld devices for applications using dynamic multimedia data. Proc. 2nd International Conference on Computer Graphics and Interactive Techniques in Australasia and South East Asia, Singapore, 123-130.
- [18] Rotuaari project (2004) University of Oulu, Finland. <http://www.rotuaari.net>.
- [19] Song S, Won Y & Song I (2002) Empirical study of user perception behavior for mobile streaming. Proc. 10th ACM International Conference on Multimedia, Juan les Pins, France, 327-330.
- [20] TeliaSonera Plc. (2004) <http://www.teliaonera.com>.
- [21] Vilovic I & Zovko-Cihlar B (2003) Performance of the Bluetooth-based WPAN for multimedia communication. Proc. 4th EURASIP Conference focused on Video/Image Processing and Multimedia Communications, Zagreb, Croatia, 2:783-788.
- [22] Way4U Ltd. (2004) <http://www.way4u.fi>.
- [23] WideRay Jack Service Point (2004) <http://www.wideray.com/product/hardware.htm>.
- [24] Ylianttila M, Pande M, Mäkelä J & Mähönen P (2001) Optimization Scheme for Mobile Users Performing Vertical Handoffs between IEEE 802.11 and GPRS/EDGE Networks. Proc. IEEE Global Telecommunications Conference, San Antonio, TX, 6:3439-3443.

Configuring Gestures as Expressive Interactions to Navigate Multimedia Recordings from Visits on Multiple Projections

Giulio Jacucci

Department of Information Processing Science,
University of Oulu,
P.O. Box 3000, 90014 Oulu, Finland
Helsinki Institute for Information Technology
(ARU), P.O. Box 9800, FIN-02015 HUT

ABSTRACT

The wide availability of digital recording devices leads to investigate how multimedia content can be navigated beyond a desktop computer set up. We present a system and a variety of applications, to navigate multimedia recordings from visits making use of: large multiple projections, location information to organise media and re-experience aspects of visits, physical interfaces as gesture-based interaction and other interfaces that render media more tangible and therefore more readily available. Analysing field trials we discuss the *expressiveness* and *experiential aspects* of gesture based interfaces as important features in navigating multimedia in "immersive" environments.

Author Keywords

Gesture-based interaction, visualising digital photographs, multiple projections.

ACM Classification Keywords

H5.m. Information interfaces and presentation (e.g., HCI):
Miscellaneous.

INTRODUCTION

Moving interaction from the virtuality of the screen of a desktop computer to the physical environment can result in diverse challenges and opportunities, according to the activity that computers seek to support. A particularly interesting activity to investigate for ubiquitous computing is digital photography. The widespread use of digital cameras is resulting, in leisure and work settings, in using the computer (also television screens, and home theatre technologies) to view and share pictures often taken at remote sites or during visits. This paper seeks to explore opportunities and challenges of providing novel physical

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MUM 2004, October 27-29, 2004 College Park, Maryland, USA.
Copyright 2004 ACM 1-58113-981-0 /04/10... \$5.00

Juha Kela, Johan Plomp

VTT Electronics,
Kaitoväylä 1, P.O. Box 1100,
90571 Oulu, Finland
juha.kela@vtt.fi, johan.plomp@vtt.fi

interfaces to visualise pictures and sounds from visits, moving the viewing and sharing of pictures from a desktop computer to a ubiquitous setting. In particular, we investigate three opportunities. Firstly, large multiple projections can provide more immersive environments to visualise digital media and sound can play a more important role. Secondly, location information can be used to organise media and re-experience aspects of visits. Finally, physical interfaces, as gesture-based interaction, can free users from the desktop and other interfaces can render media more tangible and therefore more readily available. A variety of questions and challenges accompany these opportunities. This paper explores novel interfaces to navigate collaboratively pictures from visits. In particular, we discuss the *expressiveness* and *experiential aspects* of gesture based interfaces as important features in navigating multimedia in "immersive" environments.

For this purpose, we use a distributed environment developed in the Atelier project¹ that provides: computational support to record location information of recorded multimedia during visits, a hypermedia database to store visits as *hyperdocuments*, and a variety of physical interfaces to navigate the hyperdocuments. Physical interfaces include: barcode scanner and print-outs with thumbnails and barcodes, an infrared remote control interface, and a gesture based interface. We organised trials, where pair of participants, after having recorded a visit, navigated the multimedia recordings using two back projection screens. The contribution of this paper is twofold: to present a variety of applications to support gesture based-interaction in navigating multimedia recordings from visits on multiple screens, and to provide insights from user trials on the advantages and challenges of gestures as an interaction mode in such environments.

The paper, first, briefly discusses related work in gesture-based interaction and novel interfaces for digital photography. We then discuss in detail the research questions and methodology. After that, we describe the Atelier environment clarifying which tools make possible to

¹ <http://atelier.k3.mah.se>

configure the navigation of multimedia recordings. A section is devoted to presenting highlights from the trials. The discussion summarises the lesson learned in constructing a more immersive environment to navigate multimedia recordings from visits, in particular addressing configurability and the advantages of gesture-based interaction.

Related Work

One of the ways that embodied actions or gestures are related to physical interfaces is movement. Benford et al [1] provide a framework centred on the notion of movement for physical interfaces: "new 'physical interfaces' considerably extend the repertoire of available movements to include whole body gestures, moving objects across surfaces, and moving the entire interface through space. Their aim in so doing is to make interaction "more natural, expressive, immersive or ubiquitous." Benford et al. [1]. Bodily movements have been considered to position or track a person's body, to track the movement of an object in the hands of a person, or to recognise gestures and facial expressions. Technologies to track and locate persons have been used to develop guides [2] and other location-aware information systems. Some aspects of nonverbal communication, e.g. visual expression (facial expressions, physical appearance, direction of gaze, physical posture), have been applied mostly to interact with virtual characters or for video conferencing. Gesture recognition systems, on the other hand, have been used to recognise sign language, for multimedia information kiosks, to control desktop applications and to interact with in-car devices [3]. Bodily expressions and interactive technologies have been the object of study in artistic performance [4]. Otherwise the expressiveness of gestures has been less the focus of research, which generally aims at making interactions more intuitive or usable and with a focus on single user interactions, e.g. in Bellotti et al. [5] "develop systems that can communicate more naturally and effectively with people" (p. 423). For example, Swindells et al. [6] present a system for identifying devices through a pointing gesture using custom tags and a custom stylus called the gesturePen. Similar pointing technology can also be integrated into different mobile devices as proposed by Ailisto et al. [7]. Sparacino et al. [8] developed "a "media actors" software architecture used in conjunction with real-time computer-vision-based body tracking and gesture recognition techniques to choreograph digital media together with human performers or museum visitors." Sparacino et al. apply this to dance, theater, and the circus, augmenting the "traditional performance stage with images, video, music, and text, and are able to respond to movement and gesture in believable, aesthetical, and expressive manners." Expression in interaction has been envisioned in what McGee [9] calls Contact-expressive devices, "...technologies that understand and use touch in meaningful ways—that can distinguish between a press and a caress." Most of the work in this area is oriented at developing the technology, while studies in real settings are hard to find. Also much work in tracking bodily movement has been applied to mixed reality environment supporting

remote collaboration (Luff et al. [10]), which is not within the scope of this work.

Novel interfaces for digital photography have been concentrating on information appliances more than in multiple large screens. Balabanovic et al [11] describe an easy-to-use device, "StoryTrack", that enables digital photos to be used in a manner similar to print photos for sharing personal stories. Other work focused on sharing and sending photographs through mobile devices [13]. Büscher et al. [12], in the WorkSPACE project, addressed with a comprehensive approach and technology the support for flexibility, mobility, and collaboration for landscape architects. They present one mobile appliance, the "sitepack", and two room appliances (a panel with pen interaction and a table with a horizontal display area). All three appliances run the Topos software in order to share workspaces and documents. The sitepack provides ways to index photographs and sound notes with GPS coordinates, and is also designed as a remote collaboration tool. The large display is rather used to dispose documents in a virtual environment.

Research Approach

Elsewhere, we have reported on how the development of the environment described in this paper, has been informed by field studies and trials on architecture students visiting remote sites [14, 15]. In this paper, we present more details on the configuration tools, new technological developments and findings from a new trial.

- 1) In previous trials we used two projection screens to display on one the pictures and on the other a 3D visualisation of the current position on a map. Participants could move forward and backward in the virtual 3D map between positions (or nodes) that contained media recordings.
- 2) In the new trials, both screens are used to visualise multimedia recordings or 2D maps according to user's preference. In particular, thumbnails of multimedia recordings and layouts of 2D maps can be printed out with barcodes, to provide a physical interface to the digital content.
- 3) We have implemented configurable interfaces to select the current multimedia output (separate media player on each screen or different sound outputs).
- 4) The mobile application had also support to store the compass direction of the taken picture. Now, this compass information can be used in the environment to navigate through a group of pictures that were taken in the same spot (forming for example a 360 degree panoramic view) by pointing in different directions.
- 5) The new trials focused more systematically on the use of gesture-based interaction in collaborative viewing of recordings of visits.

The approach adopted in the trials, included asking two participants to choose a place of interest. They would then visit this place, using Atelier equipment taking multimedia

recordings while performing paths, for example, walking. The participants were then invited in the Atelier environment, to navigate collaboratively the recorded “multimedia path”, and the trials ended with an interview. The scenario they were asked to perform, was therefore clearly defined, but allowed participant to choose a place, choose what to record, and also how to navigate it together. The trials were conducted as explorative and formative evaluations, trying to let users contribute with their own ideas how to use the technology. We did not evaluate particular usability aspects or measure specific performances. We were more interested in gathering insights on the appropriateness and opportunities of gesture-based interaction to navigate multimedia recording from visits on multiple media outputs.

AN ENVIRONMENT TO NAVIGATE VISITS

The use scenario we set out to support is of one or more persons visiting a remote site of interest, where they take pictures and record sounds. In addition, the system automatically records the path according to the user's movement (Figure 1). At a later time this “multimedia path” can be collaboratively navigated using multiple screens, gestures and printouts with thumbnails of the recordings. The environment is therefore described in two steps. First, the recording of the visit using a mobile application and devices is explained, also describing how multimedia paths (HyperDocuments) are stored in the Atelier environment. Second, we describe configuration tools to set up the environment for navigation. After the two steps, we present insights from the trials.

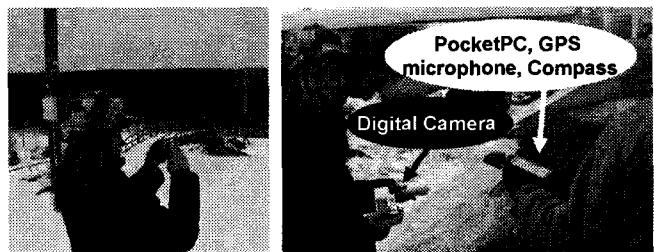


Figure 1. Performing a visit recording sounds, and pictures

Performing a visit, recording media

The path is recorded using timestamps and GPS (Global Positioning System) trace. These are created by a mobile application (eDiary), while visitors take pictures, videos, sounds, and text notes along the path. Back in the work environment the media files and the GPS log can easily be stored with an application (PathCreator) in a hypermedia database, creating a navigable and editable media path (a HyperDocument of the visit). Visitors can load a background picture as a map on which the path is visualised. Printouts with barcodes provide a physical support to navigate the visit. Using multiple projections and physical interfaces, the visitors can re-experience the visit, linking the media material to other physical artefacts (posters, models, objects etc).



Figure 2. The mobile application eDiary on a Pocket PC equipped with the SoapBox as digital compass and with a GPS receiver.

We have developed several components to support the scenario above. The eDiary is a mobile application to run on a PocketPC, which creates a HyperDocument containing a log of time and GPS traces, using a CompactFlash receiver card (Figure 2). The HyperDocument contains HyperNodes for each position; the user can set the recording interval of HyperNodes (e.g. 5 seconds, one minute) short or long, resulting in respectively more or less detailed recorded paths and less or more recorded media in each HyperNode. The GPS coordinates and time log are saved in the meta-information of the HyperNode (Figure 4). The interface of the eDiary displays the current GPS coordinates also on a calibrated map if available, provides a file menu to save the HyperDocument, and has three buttons: start recording; stop recording, and, with an additional button, users can record the direction of a taken photograph by using VTT's SoapBox as an electronic compass (Figure 2). SoapBox (Sensing, Operating, and Activating Peripheral Box) is a light, matchbox-sized device with a processor, a versatile set of sensors, and wireless and wired data communications [16]. With the iPAQ users can also record sounds or write text notes. These files and the created HyperDocument of the visit, containing the time log, the GPS trace, and the compass data, can be saved on an SD (secure digital) memory card. When returning from a visit the user connects the digital camera and the SD memory card to a PC running the PathCreator. This application is used to combine and synchronise the GPS trace and compass data, which are in the HyperDocument, with the multimedia files: pictures from the camera, sound files, and text files. As each media file has a time of creation, the PathCreator assigns each file to the HyperNode with the nearest time. The HyperNode is a container for multimedia objects, and has position information as we can see from its metainformation in Figure 4, and from its visualisation on a map visualised as a red circle in Figure 3.

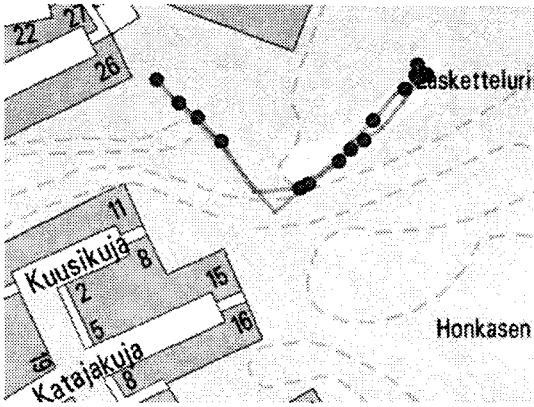


Figure 3. HyperNodes visualized on a map.

Contents of node node 14

Meta-information	
Location	64deg54.11553N 25deg33.7312E 67.3m
Date	2004-03-24 10:47:56.795
 PICT0870	
Picture information	
Filename	PICT0870.JPG
Date	2004-03-24 10:47:02
Orienteation	195.7
Thumbnail	
Content-type	image/jpeg

Figure 4. Printouts with barcodes can be used to play single pictures or whole *HyperNodes*.

With the iPAQ users can also record sounds or write text notes. These files and the created HyperDocument of the visit, containing the time log, the GPS trace, and the compass data, can be saved on an SD (secure digital) Also the compass data is assigned to the right media object based on the time. For these reasons the clock of digital recording devices needs to be synchronised with the eDiary. The PathCreator also provides loading and GPS calibration of maps, on which the visit can be visualized (Figure 2). After the visit has been saved into the database as a HyperDocument, users can print out HyperNodes with a web browser (Figure 4). Each HyperNode contains a 2D map with a blue trace of the path with a red circle on it positioning the HyperNode. The printouts contain information and thumbnails for all the related objects, and barcodes to play either the whole HyperNode or single

objects. The objects can be played on several *MediaPlayers* on different projections to create an immersive environment.

Configuring the Interfaces and Re-Travelling Visits

For the navigation users have a configurable environment with multiple projections and interchangeable physical interfaces. When a barcode from the print outs is scanned, the corresponding HyperNode or multimedia object is played on the current *MediaPlayer*. Users can configure specific physical interactions to change the current *MediaPlayer* or to play the next multimedia object in a HyperNode. The environment is built on an infrastructure with a HyperMedia database where input and output components can register and communicate through XML messaging [14].

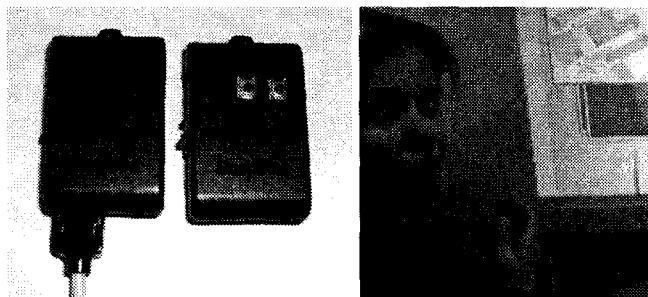


Figure 5. the SoapBox has two buttons and communicates through radio signals the data of sensors.

Several physical interfaces can be used:

- Gestures: users can train (configure) their own personal hand gestures by using the gesture recognition system based on VTT Electronics' SoapBox (Figure 5).
- Tilting: users can tilt the SoapBox resulting in four different control commands, front-right, front-left, back-right, back-left.
- Pointing: users can point in eight different direction with a electronic compass as input, north, north-east, east, south-east, south, south-west, west, north-west (e.g. to choose projections or retrieve pictures taken in a direction in a HyperNode).
- Infrared remote control: buttons on a remote control can be associated to events in the environment.
- Barcodes: all commands are available on barcode posters, barcode also turn print outs in physical handles for configurations and digital media.

Alternatively any other physical input can be configured, for example in the Atelier project we used Radio Frequency Identification tags and touch sensors.

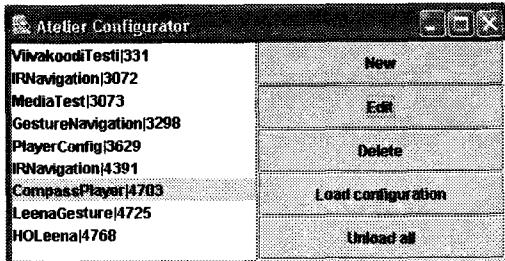


Figure 6: Storing and loading configurations for navigation

The hand gestures are detected by the acceleration sensors built into a SoapBox. The measured signal values are wirelessly transmitted from the hand-held SoapBox to a receiver that is connected to a Windows PC with a serial connection. All signal processing and pattern recognition is performed in the PC. The system is capable of detecting discrete 3-dimensional hand gestures and tilting of the device. The rotation (compass bearing) of the device is detected using electronic compass, which is also included in the sensor device [17]. Recognition results can be mapped to different control commands and transmitted using the ATELIER infrastructure. A configuration tool makes it possible to save configurations and load them “on the fly” by scanning a specific barcode (Figure 6). The motivation is to be able, with a barcode, to scan and re-configure the environment according to the preferences of the current user. This was extremely useful in test trials to be able to switch in seconds between different interaction styles. It enabled easy and fast system reconfiguration, for example, two different barcodes could be used for switching from gesture control mode to pointing mode. A GUI tool

projection screen) on which the next multimedia object will be displayed (for example by scanning the barcode next to its thumbnail).

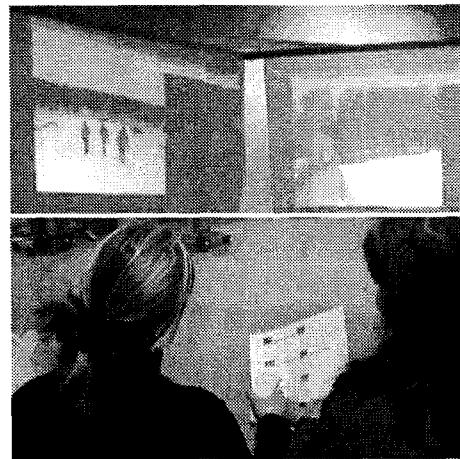


Figure 5. Navigating visits only with the barcode scanner

The creation of the configuration is a mixed interaction between physical interfaces and mouse clicks. Users do not need to write any text, as they first point to a direction with the SoapBox (e.g. south-east). This event results in an entry in the history list (see tab in Figure 6), this entry can be simply clicked and selected. In the same way users can scan barcodes next to projection screens to select specific MediaPlayer (or screens). The corresponding MediaPlayer-screen is then shown on the history list of the Output (Figure 7) and can be selected. Any physical input can be therefore configured to trigger specific commands,

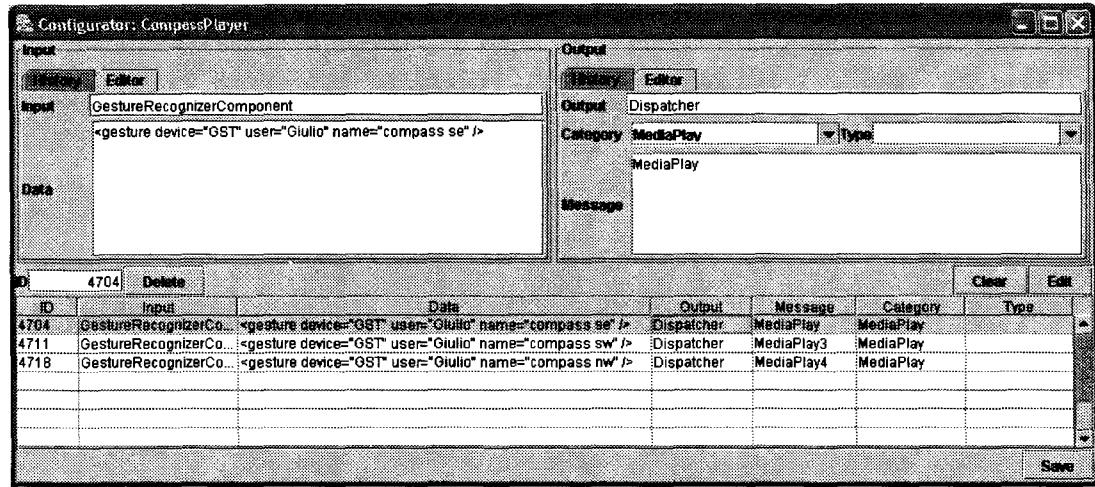


Figure 7. A GUI to edit configurations, events like gestures or scanned barcode or output components, appear in a history list to make configuration a matter of few mouse clicks.

provides a way to create, edit and store configuration tables. While most of the interactions in the Atelier environment do not need GUIs, in this case they were needed to make the management of configuration tables easier. The example of a configuration table in Figure 7 shows a configuration where pointing in different directions in the room results in changing the current MediaPlayer (or

for example, choose between large projection screens while playing the multimedia recordings from the visits, or moving to next or previous objects in a HyperNode. Pictures are also associated to a direction that can be sensed with the SoapBox as a digital compass during visits. After having scanned the barcode of a HyperNode that contains several pictures, users can also navigate pictures by

pointing the SoapBox to different directions and play media that was taken in that direction. This was particularly interesting when during visits a series of pictures form a 360 degrees panorama.

Evaluation in Trials

This analysis is based on a variety of trials that brought us, in the last session, to evaluate the environment as described in the previous chapters. The iterative design resulted in leaving the 3D-map navigation of the path on one of the two projections, adopting a more flexible set up, where the user decides moment-by-moment what to display on each projection screen. Before, to view multimedia recordings users had to move chronologically from one node to another. Now the printouts make the content tangibly available with no fixed scheme of navigation. A 2D map can still be visualised as each HyperNode and relative print out, containing a map showing the position of the HyperNode on the path. In this way the map of each HyperNode is treated as any other multimedia recording. In the last sessions two participants produced three multimedia paths of visits. The objective of this one-day visit was for one participant to show around her town and vicinities to her friend (the other participant). First they visited a skiing facility, producing the first path walking around. Reaching a height by a ski lift, they took panoramic pictures, forming a 360-degree view. Then they visited the town's spa and shopping centre, which were next to each other. Finally the third path was created by car driving to location also outside the town. The two participants were accompanied in their trip by two of our researchers. The participants were equipped with a digital camera while the researchers operated the mobile application eDiary. The participants requested to the researchers to record sounds or spoken notes using the Pocket PC in specific situations. The navigation took place in the Atelier environment arranging the space as depicted in Figure 8. Two back projection screens were used to display pictures each with a different MediaPlayer. Other two MediaPlayers were used as sound

the barcode scanner. These were used to change the active projection screen and move to next or previous objects (Figure 5).

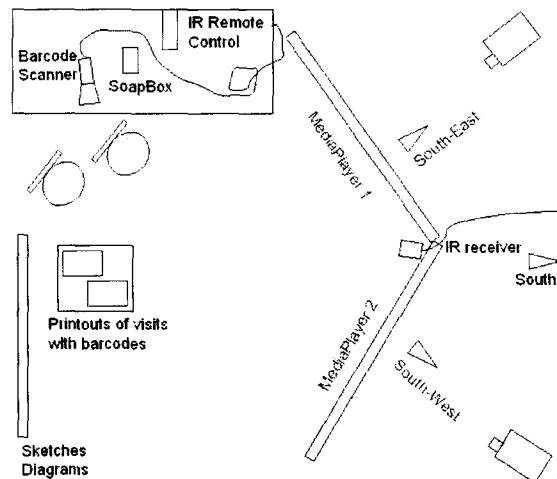


Figure 8. Physical setup of interfaces for the trial.

Then, we loaded a new configuration that enabled to choose the current MediaPlayer using the SoapBox as a compass pointing at different directions (South-East MediaPlayer1, South-West MediaPlayer2, North-East MediaPlayer3 for sound, North-West MediaPlayer4 for sound). Finally, each participant configured own gestures with the SoapBox gesture training system. The gestures for left screen, right screen, and next object, for example, were different from one participant to the other reflecting different personalities and individual ways of gesturing. This third configuration was particularly interesting. Both participants were choosing a MediaPlayer with a gesture, moving the SoapBox from left to right or right to left to respectively choose as current MediaPlayer the South-East and South-West projection (Figure 9). The gesture for next was a movement of the Soapbox from the chest towards the front, as using a fishing pole. This resulted in a “throwing”

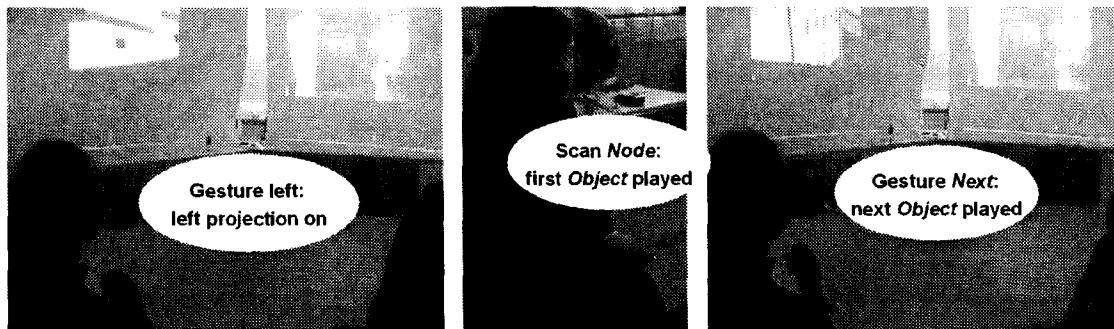


Figure 9. Choosing the left projection and moving to the next objects using gestures.

outputs. We organised the trial using three different configurations. The two participants navigated the three visits in the environment, using these three different configurations of physical interfaces, and pictures and sounds triggered discussions and memories of the trip. The participants started to navigate the visits (HyperDocuments) of the previous day, using the infrared remote control and

gesture for next, always towards the current screen that had been selected. Although gestures have no orientation (no absolute direction, also relative accelerations) and can be performed in any direction (Figure 9), this feature was particularly useful for allowing the other participants to follow what was happening. Several HyperNodes contained a group of pictures forming a panorama view. Participants

could navigate these HyperNodes by first scanning the barcode of the HyperNode and then using the SoapBox as an electronic compass to retrieve pictures that were taken in a specific direction (Figure 10). The availability of several sound outputs (MediaPlayer) resulted in users playing two sounds simultaneously, for examples voices recorded walking outside were mixed together with sounds recorded in a swimming pool resulting in an interesting mixing of "context". The interviews revealed interesting insights on the opportunity to use gesture based interaction in such an environment. Gesture based interaction was experienced as very positive, especially compared to the other modes: "With the gesturing I can see what the other is doing... when she was scanning the barcodes I had to go and check what she was doing ... The barcode was boring, and we made mistakes like scanning the same one twice, the gesturing was more active, better." And again one participant explained how she could understand what was going on: "Because this movement (gesture to the left) is so big that you do not have to watch it, you just see it (when the person next to you performs it)." Most of the problems that participants lamented were about the poor feedback of the system: "the feedback was poor to know if the system understood the gestures and if the current player has been changed". This was a problem as: "Sometime I choose a screen then we started talking with Marika and then when I wanted to use the system again I did not remember the current player." Commenting on the print outs as physical handles one participant explained: "It is easy to find what I am looking for, for example when I take a lot of digital pictures now it is hard to find the right picture when I need it because the name is only a number." The use of the SoapBox as an electronic compass was also received very positively, especially in navigating group of pictures forming a panorama. One participant commented on her navigation of the pictures taken on the height by the ski lift arrival. She observed how the acts of pointing in different directions, discovering again the different pictures, gave her a sense of being there even feeling "the cold weather". The participants also noted how the use of gesture based interaction and the barcode scanner to select the content was possible in collaboration with another person. The SoapBox in our set-up utilises both its buttons: left button activated gesture recognition, the right button activated the use of tilting, and finally pressing both buttons

simultaneously enabled the use of the electronic compass. In the discussion with the participants a device was envisioned that would merge all functionality, e.g. a barcode scanner that support gesturing tilting and pointing.

DISCUSSION AND CONCLUSIONS

This paper presented a variety of interfaces and applications to navigate, in an immersive environment, multimedia recordings from visits. The exploration focused explicitly on three opportunities of supporting the viewing of digital pictures in ubiquitous environments: the use of multiple projections and sounds, location information to organise media and re-experience aspects of visits, physical interfaces to render media more tangible and therefore more readily available. We described a distributed system, a mobile application and configuration tools, which embody such a vision. This gave us the possibility to investigate human-computer interaction issues in such an environment. In particular, we have investigated the opportunities and advantages of gesture based interaction. This interaction mode is very promising in such an environment. The results showed that participants configured gestures in different ways, becoming expressive of their own way of gesturing. More importantly, gesture based interaction was particularly helpful to render the interaction intelligible to co-participants. Participants in interview stressed the fact that with gestures they could better follow what was happening and also provide better accountability when the "system" was not responding properly. Finally, it embodied interaction, increasing the active and "physical" participation of users. This, according to participants, was an important feature to re-experience the visit for example by re-discovering pictures pointing in various directions.

ACKNOWLEDGMENTS

We are grateful to our co-researchers in the Atelier project, which is funded by the EU IST Disappearing Computer programme. We wish to acknowledge the contributions of Kari Kuutti, Anti Juustila, and Virtu Halattunen (University of Oulu), Infotech Oulu for supporting this research at the University of Oulu

REFERENCES

1. Benford, S., Schnädelbach, H., Koleva, B., Gaver, B.

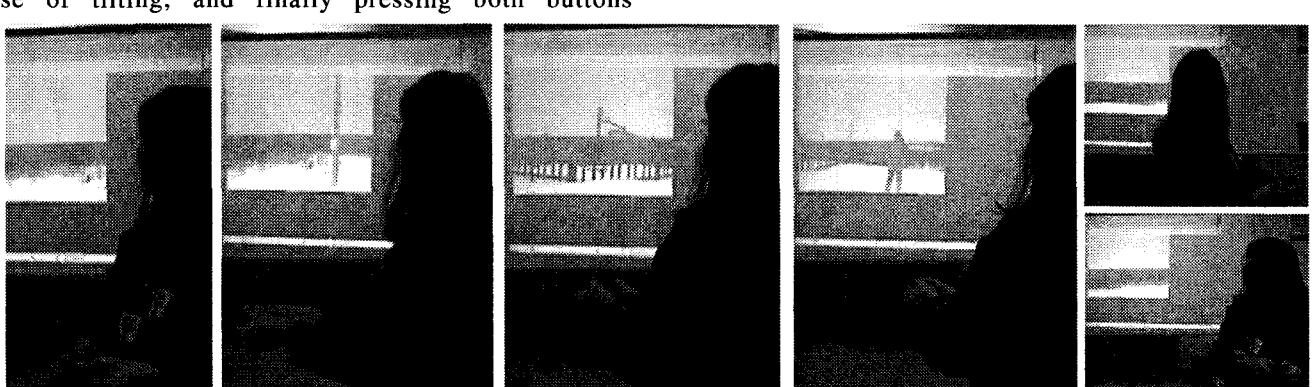


Figure 10. Navigating groups of pictures forming a panorama view using the SoapBox as an e-compass.

- Schmidt, A. Boucher, A. Steed, A. R. Anastasi, C. Greenhalgh, T. Rodden, and H. Gellerson. Sensible, sensible and desirable: a framework for designing physical interfaces. Technical Report EQUATOR-03-003, School of Computer Science & IT, Nottingham University, 2003.
2. Davies, N., Cheverst, K., Mitchell, K., Efrat, K., (2001) Using and Determining Location in a Con-text-Sensitive Tour Guide. *IEEE Computer* 34(8): 35-41.
 3. Camurri, A., Volpe G., Gesture-Based Communication in Human-Computer Interaction, 5th International Gesture Workshop, GW 2003, Genova, Italy, April 15-17, 2003, Lecture Notes in Computer Science, 2195, Springer 2004.
 4. Camurri, A., Mazzarino, B., Ricchetti, M., Timmers, R., Volpe, G., Multimodal Analysis of Expressive Gesture in Music and Dance Performances. In: Camurri and Volpe (2004).
 5. Bellotti ,Victoria and Maribeth Back and W. Keith Edwards and Rebecca E. Grinter and Austin Henderson and Cristina Lopes, Making sense of sensing systems: five questions for designers and researchers, Proceedings of the SIGCHI conference on Human factors in computing systems, 2002, ACM Press.
 6. Swindells, C., Inkpen, K.M., Dill, J.C. and Tory, M., (2002) That One There! Pointing to Establish Device Identity. In Proceedings of the 15th annual ACM symposium on User interface software and technology, Symposium on User Interface Software and Technology, 151 – 160.
 7. Ailisto H, Plomp J, Pohjanheimo L, Strömmér E (2003). A physical selection paradigm for ubiquitous computing. 1st European Symposium on Ambient Intelligence (EUSAIR 2003). Ambient Intelligence. Lecture Notes in Computer Science Vol. 2875. Aarts, Emile et al. (Eds.). Springer-Verlag. Berlin, pp 372 - 383.
 8. Sparacino, G. Davenport, and A. Pentland (2000) Media in performance:Interactive spaces for dance, theater, circus, and museum exhibits. In: IBM Systems Journal Vol. 39, Nos. 3 & 4, 2000, p. 479 –510.
 9. McGee, Kevin. A Touch of the Future: contact-expressive devices. In: *IEEE MultiMedia*, 11 (1). January-March 2004.
 10. Luff, P., Heath, C. Kuzuoka, H., Hindmarsh, J., Yamazaki, K. and Oyama, S. (2003) 'Fractured ecologies: creating environments for collaboration', *Human Computer Interaction* 2003, Vol 18 pp 51-84.
 11. Balabanovic M, Chu, L.L. & Wolff G. (2000) Storytelling with digital photographs. *Proceedings of CHI 2000*, 564-571. New York; ACM SIG-CHI.
 12. Büscher, Monika, Gunnar Kramp and Peter Gall Krogh, In formation: Support for flexibility, mobility, collaboration and coherence, *Personal and Ubiquitous Computing*, Vol 7, numbers 3-4, 2003, Springer.
 13. Mäkelä, A., Giller, V., Tscheligi, M. and Sefelin, R., 2000. Joking, storytelling, artsharing, expressing affection: A field trial of how children and their social network communicate with digital images in leisure time. *Proceedings of CHI'2000*. ACM Press, pages 548-555.
 14. Iacucci, G., Juustila, A., Kuutti, K., Pehkonen, P., Ylisaukko-oja, A., (2003) Connecting Remote Visits and Design Environment: User Needs and Prototypes for Architecture Design. In: the Proceeding of Mobile HCI 03, 8-11 September 2003, Udine, Italy,Lecture Notes in Computer Science, Springer Verlag. Pp. 45-60.
 15. Iacucci, G., Kela, J., Pehkonen, P., (2004) Computational Support to Record and Re-experience Visits, *Personal and Ubiquitous Computing Journal*, Volume 8, Number 2, Springer Verlag, London. May 2004, Pp. 100–109.
 16. Tuulari, Esa; Ylisaukko-oja, Arto, (2002) SoapBox: A Platform for Ubiquitous Computing Research and Applications. *Lecture Notes in Computer Sc.* 2414: *Pervasive Computing*. Zürich, CH, August 26-28, 2002. Mattern, F. Naghshineh,M. (eds.). Springer, pp. 125–138.
 17. Kallio, S., Kela, J., Mäntyjärvi, J. (2003) Online Gesture Recognition System for Mobile Interaction. *IEEE International Conference on Systems, Man & Cybernetics*, Volume 3, Oct 5-8, Washington D.C. USA pp. 2070-2076.

IP NETWORK FOR EMERGENCY SERVICE

K. K. A. Zahid

GITS

Waseda University

Tokyo, Japan

email: kalzahid@moegi.waseda.jp

L. Jun, K. Kazaura and M. Matsumoto

GITS

Waseda University

Tokyo, Japan

ABSTRACT

Internet has become the primary medium for worldwide communications. In terms of recreation, business, and various imaginative reasons for information distribution it is most used communication method today. Recently it is a big issue how we can make best use of it in emergency period. The goal of this paper is to ensure Internet's best effort service to help in emergency without any major changes in existing technology. Here we present a system architecture that reflects a general overview of how it can work parallel with the authorized Emergency Telecommunications Service (ETS). To detect location information in loosely coupled network environment is the most challenging aspect using IP's flat address model. Our conceptual view of the system architecture shows how the service can be provided extending DHCP, XML with existing technology. A simulation of the system is also shown to realize call setup delay and end to end VoIP packet delay over the IP network.

KEY WORDS

Location Information, i-PSAP

1 Introduction

Disaster can happen unexpectedly any time in human life. Quick response for rescue operation requires immediate access to public communication capabilities at hand. This includes several technology like PSTN, cellular phone, Wireless PDA, IP telephone etc. The commercial telecommunication technology is rapidly evolving to internet based technology. Therefore, the Internet community needs to consider how it can best support emergency management and recovery operations. But we still do not have end to end IP solutions to help us in emergency situation.

The main problem to migrate the IP based emergency network is to locate an end user (PC or any IP enabled device) in LAN/WAN environment. The user does not broadcast its location information (LI) to its attach subnet. So finding the position of an user in heterogeneous wired environment still represents an open issue. With the help of DHCP message transfer we present here how user's identification can be transferred to some directory server. It is not a good idea to put some static information in the user terminal. Because the user can move any time and can connect

different access point to reach the internet. The proposed method takes users information at runtime and dynamically updates LI in both end terminal and access point's associated location information server (LIS).

Another important factor in emergency situation is to reach the service rapidly. Internet does not use signaling to transfer message as fixed telecommunication does. But to ensure the emergency call setup we need some signaling mechanism which can establish the communication within an agreeable time limit. SIP and H.323 are such a protocol used for signaling in internet.

2 Motivation

Most developed countries have their own emergency service infrastructure. This facility however more or less closely coupled with traditional Public Switch Telephone Network (PSTN). For example "911" is the emergency number in US. On average 260,000 emergency 911 calls are made daily in US, accumulating to 95 million call per year [1]. To our best knowledge there are no existing services where IP network is used as major component. Different VoIP application services like net2phone is very popular for their low cost and easily accessibility. PSTN based service is already equipped but ES (emergency service) using IP still lacking some infrastructural components to provide LI of the user to the ESN (Emergency Service Network).

The current internet provides a simple best effort service where the network treats all packets equally. Although bandwidth of Internet is continually increasing, the backbone of Internet is still far from being able to support Quality of Service (QoS) without appropriate resource provisioning [2]. So we need some additional mechanism to implement fully integrated IP based solutions to handle emergency request.

3 Requirements

Several organization and standardizing bodies like IETF's GEOPRIV WG (Working Group), IEPREP WG and SIP & SIPPING WG are related with the development of emergency communication service. These groups have already proposed different location schemes, signaling protocols

and IP based solutions for emergency situation. Four basic topologies have been identified for internetworking with traditional circuit switched networks. They are IP bridging, IP at the start, IP at the end and End to End IP scheme. In following subsections we devoted our work to describe the methods to identify LI which fits in IP format and can transfer LI to local i-PSAP (internet based Public Safety Answering Point) for disaster recovery.

The common players for IP in Emergency Preparedness depend on TCP-UDP/IP stack. The other necessary protocols are DHCP, XML and signaling protocols (H.323, SIP etc.). To establish the connection between Emergency Caller (EC) and i-PSAP (Callee) both TCP and UDP based packet communication can be used. Sometimes UDP performs well to increasing the probability of reaching packets when the network is congested as in this case TCP usually backs off. For the end user we can choose any IP enabled terminal which supports multimedia in application layer protocols. Rather using PSAP we called it i-PSAP as it is mainly dedicated to respond for IP network.

3.1 Location Information

Location information is a description of a particular spatial location, which may be represented as coordinates (via longitude, latitude, and so on), or as civil addresses (such as postal addresses), or in other ways. There are different granularities for different objects location. IP enabled devices need to know their location to contact emergency centers during unexpected situation. A location of an object is a place where it is located physically. Location information can be managed in many ways. The most common method is to put it manually in Location Information Server (LIS) through web based form.

Civil information is useful since it often provides additional, human-readable information particularly within buildings. Also, compared to geospatial information, it is readily obtained for most occupied structures and can often be interpreted even if incomplete.

3.2 Why DHCP?

The dynamic host configuration protocol (DHCP), accepted as an internet standard by the IETF in 1997. It automated the host configuration of network devices that uses TCP/IP. DHCP servers act as agents for network administrator and automate the process of network address allocation and parameter configuration. DHCP clients and server interact through a series of client initiated request-response transactions [3]. DHCP message is a fixed section format followed by a variable-format section. The DHCP message format is shown in figure 1.

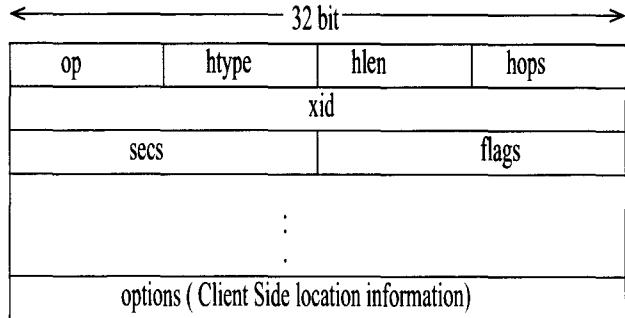


Figure 1. DHCP message format with LI

3.3 XML for LI encoding

The Extensible Markup Language (XML) is a textual markup language subset of SGML that is defined in 1996 by W3C (World Wide Web Consortium)[4]. One use of XML is that it can manage multiple views of data. It is human friendly and easily deployable irrespective to device capability and also platform independent. It is also a universal hub for information exchange.

3.4 SIP for Signaling

SIP is an application-layer control protocol that can establish, modify and terminate multimedia session such as internet telephony calls. It is text based and SIP messages can be carried in UDP or TCP. They are formatted similar to HTTP messages, which is plain text headers followed by an opaque message body and facilitates easy service creation by a very large community of software developers. It has event notification capability for example NOTIFY method which we used here for emergency alerting. Finally SIP is transport layer protocol neutral and involves less signaling, call setup is faster and capable to join in parallel search.

4 Network Architecture

Initially the system concentrates to find location information either from end user or from LIS or combination of both. In figure 2 we see that the stored location hardware is placed in user area to provide LI to the user terminal. This hardware could be any technology like RFID, IR, Bluetooth, IEEE802.11 access point with embedded application programming interface (API) which can transfer stored information to user through proper interface within its radio frequency range. The UI (User Information) is an emergency application running in user terminal can access any reachable API to get correct LI instantly. This hardware store LI according to IETF's GEOPRIV location object format [5].

The Civil location information is passed to the DHCP server through DHCP message options fields after a valid

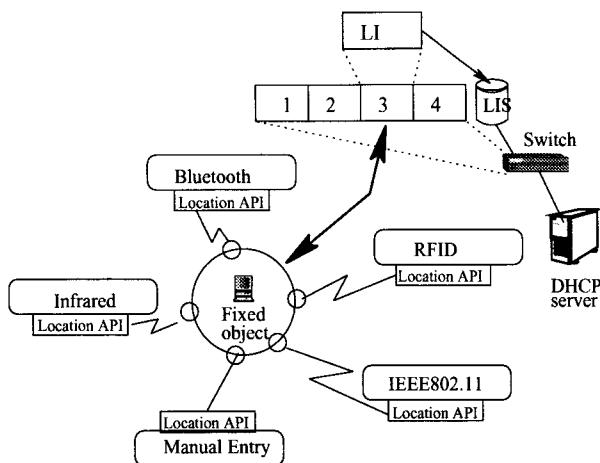


Figure 2. Location prototype for end user

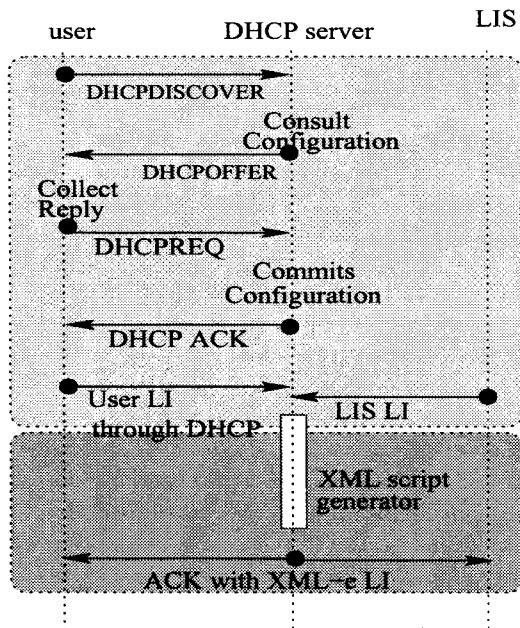


Figure 3. LI establishment using DHCP

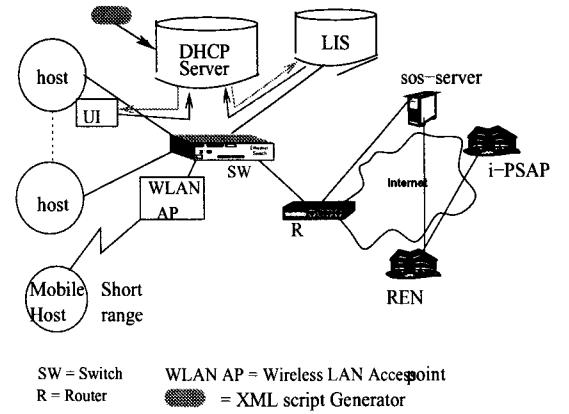


Figure 4. Model Architecture

IP address is assigned to the end user. In traditional DHCP setup process we require two more passes to establish the location information between the end user and LIS. The advantage of using LIS is, if the IP address of user is registered at a certain point of attachment then next time the user will get the correct LI even if the user does not send any information. In this case the MAC address of the host machine will play the key role to retrieve the appropriate LI from the LIS. The civil information of user is composed of different parameters and encoded in XML structure to identify his presence at certain location. Both user side and server side information is managed using some XML Document Type Definition (DTD) files. The server side DTD is stored in the LIS. After getting a valid IP address from the DHCP server user sends its client DTD through DHCP message options fields to XML script generator (XSG) located in each service provider's area. XSG takes both end user's XML DTD and server's XML DTD and generates complete XML-e-LI (XML encoded Location Information). It then sends this information to both UI and LIS for future use. The entire process of location setup using DHCP message options field is shown in figure 3. If UI sends a blank LI the XSG generates a LI based on LIS's predefined configuration stored in the IP to port mapping database (see figure 2). This time the mapping of area, building, floor, room etc. associated with the port will be used to generate LI.

An important component of IP based emergency service network is REN (Root Emergency Server) whose main task is to gather caller information, find the closest i-PSAP from hierarchical location database, and reply with LI to the Caller.

We considered a new gTLD (generic top level domain) server .sos in our paper. The question is why we need another gTLD server? One of the reason that we need the new .sos server is because it carry the normal request load, including a high proportion of unanswerable request from poorly configured or buggy DNS resolver. Another reason is to protect distributed denial of service.

Service#	Associated service
1	Police
2	Fire service
3	Marine
4	Ambulance
5	etc

Table 1. Emergency service options

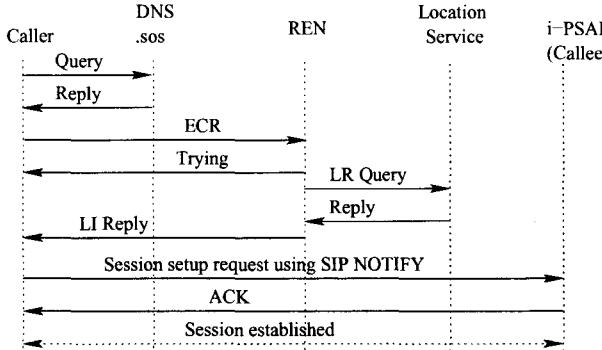


Figure 5. Sequence diagram of Session establishment

The caller requests the REN through any text or multimedia based application from IP enabled devices (in our case we choose VoIP application) with the number (described in Table 1) of emergency services he or she requires. Before transferring the VoIP packet to the callee it is necessary to find the appropriate REN for that region. So the call is routed first to the local DNS and if there are no information for sos domain in the cache it ultimately consults with the top level sos server. For this purpose the caller sends the simple Query message to the sos DNS server. After getting a valid response it sends the Emergency Call Request (ECR) to the REN putting the XML-e-LI in the TCP/UDP packet's payload field. REN sends a LocationRequest (LR) query including service number and XML-e-LI to the location services that can find the best i-PSAP based on the LI. In reply REN gets the URLs, IP address, scripts and other preferences to reach the closest i-PSAP and send back to the user as an LocationRequest Reply. Finally a NOTIFY (an event method of application level signaling) is issued using SIP from the Caller to the Callee. Whenever an ACK is available to the caller from Callee the session is established between them and transmission of media packet takes place. The interaction between Caller and Callee is shown in figure 5.

We extended the NOTIFY request to include the XML-e-LI and send it to the Callee. So the emergency responders can identify the victims' physical location. After a secured authentication process the Callee proceeds for further steps, for example sends rescue team in appropriate

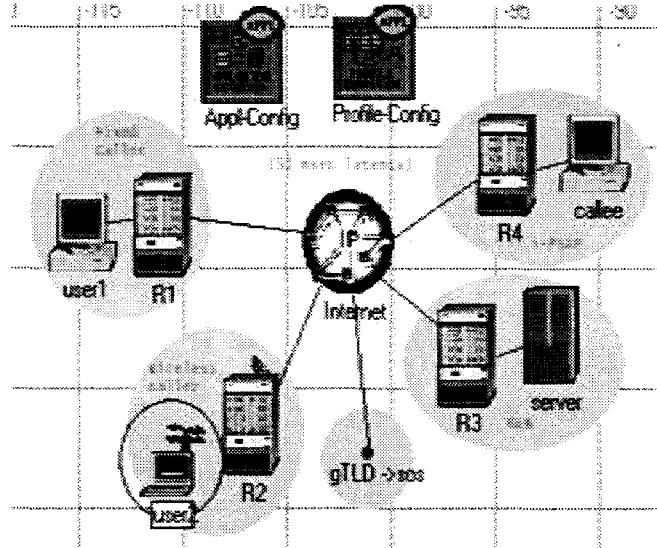


Figure 6. Network model for Emergency service

place.

5 Simulation

Here we present a simulation approach to find out the call setup time and the average delay for VoIP packets experienced by the fixed and wireless caller. For simulation we used OPNET simulator's discrete event simulation approach. The internet is designed using ip_cloud with a exponential packet latency of 50 msec. PPP-DS1(1.544 Mbps) link is considered for LAN to WAN connection. The fixed LAN user and i-PSAP are given 1000BaseX links (operating speed - 1000 Mbps) while the WLAN (IEEE802.11) user runs at 2 Mbps. End systems are general purpose computers capable of generating multimedia (VoIP) packets. The other component of the model is IP routers, WLAN routers etc. The network model is shown in figure 6.

Internet is composed of IP networks under different constraints. There will be many proxies and gateways between the internet and circuit-switched networks operating under a multitude of business models and organizations [6]. To contact i-PSAP we used VoIP application which comprises of two main components: signaling and media stream to carry the conversations. The delay of providing LI in SIP depends on numerous factors. The most obvious one is the message propagation delay on the transport network. As the SIP protocol is an Internet protocol, it also depends on flow related parameters like the utilization and resulting queuing delays. And SIP uses its own timer for unreliable transport protocols. Performance can also be affected by other factors such as delay of DHCP and security

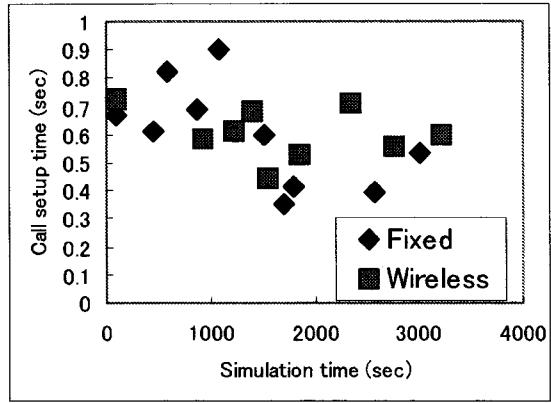


Figure 7. Call setup time

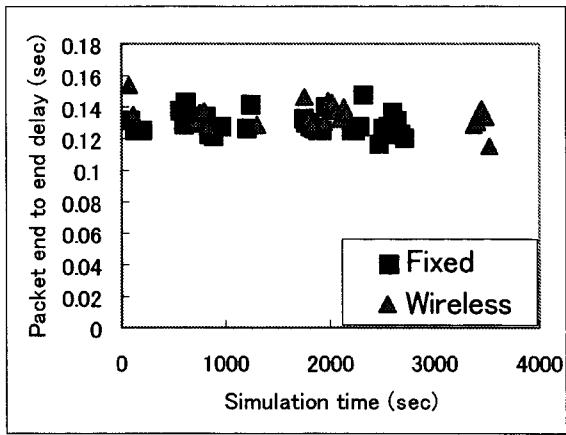


Figure 8. End to end VoIP packet delay

operation if applied. In order to simulate the network we monitored the following parameters to evaluate the system performance,

- 1) Call setup delay between Caller and Callee
- 2) End to End media packet delay experienced by the caller

The simulation is performed for both fixed and wireless callers separately using the same network parameter. An average one way delay of around 150 ms is considerably acceptable for media packets in VoIP transmission [7]. Figure 7 and 8 depicts the call setup time and end to end VoIP packet delay for wired and wireless caller respectively.

6 Analysis

In summary the time required to reach i-PSAP depends on two factors. First to resolve the .sos TLD to find the authoritative address of REN in the requested region. Second, the time to connect from Caller to Callee. IP to Host resolution varies from location to location due to local differences in available bandwidth, network architecture and proximity to the element supporting DNS resolution such as root servers, generic top-level domain (gTLD) servers, as well as other servers like country-code top-level domain (ccTLD) servers and the authoritative servers for the domain names being resolved [8]. The mean response time for completed lookups in DNS varies from 0.95 seconds to 2.31 seconds [9]. The time to retrieve the authoritative server depends also on the DNS caching. Without caching in local server DNS Query would have to begin at the root, continue to TLD server, and then end at the authoritative DNS servers.

An easy way to speed up this process is to cache the emergency server (REN) information locally thereby eliminating the need for repetitive queries to the remote DNS server. DNS caching reduces the name resolution time by more than two orders of magnitude[10]. So rather than using the remote DNS server that is provided by ISP we need local DNS caching for emergency server lookup especially in each user side. This local connection can be performed much faster and does not depend on the response time of the remote DNS server across the Internet. As the REN server address does not change frequently we can make the TTL for REN server a long period so that the entry does not destroy from local DNS server frequently.

In general the Emergency services requirements are that an approximate position, such as that offered by PSTN or Cellular phone, be available in less than 15 seconds, and that an accurate position estimate be available in less than 30 seconds [11]. So the time required to reach the Emergency Service Network is $\sum(\text{time to resolve REN either from local DNS or gTLD (.sos)} + \text{Call setup time between caller and Callee})$. Our simulation satisfies this requirement and indicates adopting the system we can open a universal interface for big community to access emergency centers using various application.

7 Conclusion

The need to able to support emergency users in public Internet is obvious. But how it can be practically achieved is still a complex issue. This paper shows some positive indication to adopt IP for lifeline. The described scenario is simple; deployment cost is low and scalable. Our simulation study has indicated a considerable acceptability of adopting the system. At the end the usefulness of the system should be realized by mass people so that they can take

forward step to make life more safer with accessible emergency service.

References

- [1] “FCC Docket No. 96-264: Revision of the commission’s rules to ensure compatibility with enhanced 911 emergency calling system.” [Online]. Available: <http://www.fcc.gov/>
- [2] A. Striegel and G. Manimaran, “Dynamic dscps for heterogeneous qos in diffserv multicasting,” *Global Telecommunications Conference, 2002. GLOBECOM '02. IEEE*, vol. 3, pp. 2123 – 2127, Nov. 2002.
- [3] R. Droms, “Automated configuration of tcp/ip with dhcp,” *Internet Computing, IEEE*, vol. 3, no. 4, pp. 45 – 53, July–Aug 1999.
- [4] T. Bray, “Extensible markup language (XML) 1.0 (third edition),” February 2004. [Online]. Available: <http://www.w3.org/TR/REC-xml>
- [5] J. Peterson, “A presence-based geopriv location object format draft-ietf-geopriv-pidf-lo-01,” 2004.
- [6] K.S. King and S. Bradner, “Internet emergency preparedness in the IETF,” *Applications and the Internet Workshops, 2003. Proceedings. 2003 Symposium on*, 27-31, Jan. 2003.
- [7] M. K. Ranganathan and L. Kilmartin, “Performance analysis of secure session initiation protocol based voip networks,” *Computer Communications*, vol. 26, pp. 552–565, Jun 2003. [Online]. Available: www.elsevier.com/locate/comcom
- [8] C. E. Wills and H. Shang, “The contribution of DNS lookup costs to Web object retrieval, Tech. Rep. WPI-CS-TR-00-12, 2000. [Online]. Available: citeseer.ist.psu.edu/wills00contribution.html
- [9] R. Liston, S. Srinivasan, and E. Zegura, “Diversity in dns performance measures,” in *Proceedings of the second ACM SIGCOMM Workshop on Internet measurement*. ACM Press, 2002, pp. 19–31.
- [10] A. Shaikh, R. Tewari, and M. Agrawal, “On the effectiveness of dns-based server selection,” in *INFOCOM 2001, Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies*, vol. 3, 22-26 April 2001, pp. 1801 – 1810.
- [11] J. Malenstein, “Co-ordination group on access to location information by emergency services,” May 2001. [Online]. Available: <http://www.telematica.de/cgalies/index.html>

Mobile Multimedia Service's Development

- Value Chain Perspective

Jari Karvonen

University of Oulu

Department of Information

Processing Science

P.O.Box 3000

FIN - 90014 University of Oulu

Jari.T.Karvonen@oulu.fi

Juhani Warsta

University of Oulu

Department of Information

Processing Science

P.O.Box 3000

FIN - 90014 University of Oulu

Juhani.Warsta@oulu.fi

ABSTRACT

This paper describes and analyses the mobile value chain development, value generation and mobile application development processes. The study bases on the eight small multimedia and software companies that uses the latest technological possibilities provided by the mobile Internet. The analyzed development processes give good understanding how the mobile value chain is composed as seen from the perspective of the case companies developing state of the art mobile multimedia service and content applications (MMS). The developed mobile multimedia applications were games, short films, music videos, still pictures, and advertisement videos. The processes were scrutinized in order to find out how the different groups of developers act together with their value net partners according to the preset project plan as they develop new applications and contents in mobile multimedia environment. The essential elements, actors as well as their roles in the MMS development processes were analyzed in order to depict the structure of mobile value network.

Keywords

Value chain, mobile multimedia service, mobility, MMS

INTRODUCTION

Mobile and mobility opens up many opportunities to serve and to make our daily life easier. We are willing to carry our mobile terminals wherever and whenever and we have extended our reachability, if we want to be available. We have ubiquity access on different databases and places and we may combine simultaneous tasks together, like jogging and transmission of our heart beat telemetry. (Tsangalatidou & Pitoura, 2001) We have adaptive channels that merge the companies' production and sales information and this is done in real time. All these possibilities are included in the terminal that we carry with us. The nascent mobile solutions makes this possible, but only if these solutions have been created. By improving these mobile web applications developing capabilities, the definition of the

required value chain and the analysis of developers strengthen the service creation processes.

Value generation

Companies' primary existence is based on the value increase that they offer to customers utilizing the product or service. If these companies do not offer enough value for customers or the value is less than the customer is willing to pay, the normal consequence is that the company exits from the markets like Coase's law has vindicated. The basic function of the firm is to increase the value for the customers and for the shareholders. The value chain model describes this value increase process and it was first presented by Porter (1985). In the model Porter describes two major entities, primary activities and support activities. The primary activities form the apparent value chain including five steps – inbound logistics, operations, outbound logistics, marketing and sales and service. The support activities – infrastructure, human resource management, technology development and procurement – include functions that support and help the primary activities to operate properly. The primary activities characterize the concrete and apparent value chain while the support activities are more aimed on the company's internal structures. Porter's original purpose were to identify the value creating processes of products and services within a firm. (Maitland et al. 2002). Porter's value chain framework defines four steps in firms. These steps are to identify the strategic business unit, identify critical activities, define products and determinate the value of an activity. (Amit & Zott, 2001) Later on has this concept being broadened to describe the entire industry level value chain. Industry level value chain model observes the firms in the market independently. Industry level value chain describes the positions of firms in the overall industry.

Value generation, relationships and network effects

Value creation analysis and the concept of the value chain broadened on the form of value nets in strategic management literature. (Gulati, Nohria & Zaheer, 2000; Möller & Svahn, 2003; Parolini, 1999) The original concept of 'value net' or on the longer form 'the value creating network' were introduced by Kothandaraman and Wilson (2001). The network has built upon relationships, capabilities and superior customer values of key firms within the value chain. The value incensement of network generates by the capabilities of firms within the network. In other words, the core capabilities of all firms together create the superior customer value. The way the firms combine to create this value captures the nature of relationship that firms have between themselves. Amit and Zott (2001) found out the four sources that created value in e-business. E-business' value drivers were efficiency, complementarities, novelty and lock-in.

Value generating networks build up and stay together by various of reasons. These networks may create strategic alliances between each other or the networks can compete together as a separated value generating entities. Value generating networks stay stable only as long as these have strategic importance for the participants of the network. (Amit & Zott, 2001) Strategic networks impacts on business in many ways. Strategic networks offers the potential to share risks and possibilities to generate the economies of scale and scope (Katz & Shapiro, 1985; Shapiro & Varian, 1999), shares knowledge and facilitates learning (Anand & Khanna 2000) and shortens the time to market (Kogut, 2000)

In this context, to where there has a strong impact with technology, the Kothandaraman and Wilson's (2001) model that has been drawn in Figure below, offers a good base for value analysis. Kothandaraman and Wilson pointed out that the network's value generation has a combination of core capabilities, relationships and superior customer value. Core capabilities defines the firms ability to deliver services and goods that satisfies customers needs and wants coming from the market. Relationship aims at the cooperation of the firms within the value network. Together these two, core capabilities and relationships, create the value network that generates the value offerings for customers being also the third part of the model.

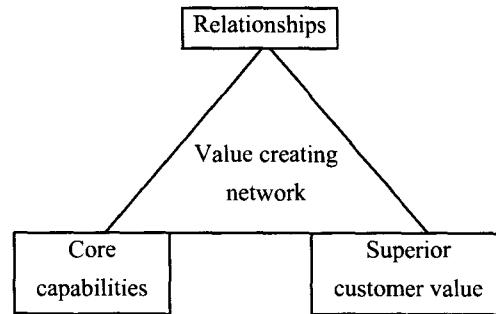


Figure 1: A model of value-creating network (Kothandaraman & Wilson, 2001)

Relationship, value generation and strategic network create the environment to where the forthcoming products will be defined, provided and created. The mobile products and services belong on the own and special economical environment that should be considered during the development. Mobile applications, like mobile multimedia services, based on the combination of content, software and transactions that are deliverable in networks. Software as a digital and easily copied commodity offers a possible to force the network effects. The network effects have long history in economics. Katz & Shapiro's (1985) article defines the direct- and indirect network effects to where the utility that consumer forces based on the number of the users of certain network. Typically when the number of users of a product or service or network increases, the value of the product or network to the other users changes. (Cave, Majumdar, Vogelsang, 2002) The direct network effect have been generated trough a direct effect of the number of the purchaser on the value derived from a product. Mobile subscriptions is a typical example of the direct network effect. The more subscriptions, the more valuable it is for users. The indirect network effect becomes from the community of certain service, product or network. Famous software, like WordPerfect, may create user community that benefits from the possibility to ask help in defeated tasks within community. The indirect network effect forms out by the complementary commodities to where the utility is collectively dependent of those dependant commodities.

Mobile value chain

Wireless communications and value systems have long history starting on the Marconi's wireless telegraphy and continuing through out 1G, 2G until the present 2,5G from where the development continues to forthcoming 3G and 4G. Historical value chain of wireless industry were simple. In the pre-cellular time (until -83) the network operators managed the whole value chain of wireless markets. The 1G brought equipment manufacturers among wireless value chain. The development from the simple

market structure to the complicated and fractured markets have begun. (Steinbock, 2003)

The second generation mobile phones brought the value added services among the market. These networks were digital¹ that enables the data transactions delivery through out the mobile network. This have been seeing as a start of the content services. The mobile data services combines features from mobile telecommunications, from data communications and from internet revising value addition process to be a combination of voice and data services. Maitland et al. (2002) represents the simplified model of the traditional mobile value chain in Figure below.

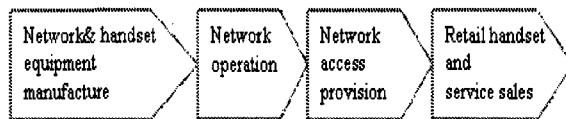


Figure 2: Value-chain for 2G mobile telephony (Maitland et al. 2002, look also Li & Whalley 2002)

Maitland et al.'s simplified model presents the operative functions of mobile value chain. If the analysis have been transformed in the content services providing process, would it brought forward the complexity of the value chain of mobile industry. The 2G value chain included separated technical solutions in different continents. In Europe the GSM² rises in a role of standard. Barnes (2002) appraise that half of the mobile users uses GSM standard³. The Asia-Pacific used on-line typed PDC⁴ that succeeds in the mobile services area. The United States follow up by two lines, TDMA and CDMA⁵. Many standards and several

¹ GSM, TDMA, CDMA and PDC

² GSM (Global system for mobile communications) network has circuit switched data transmission structure that offers 9,6 (later on 14,4) kbps/s data transaction speed for one channel. As a digital network, the GSM possibilities short messages sending throughout the network. GSM network is updatable as HSCD, GPRS, EDGE or E-GPRS networks

³ GSM World association headlined at February 2004 that more than one billion people users GSM phones. (<http://www.gsmworld.com/index.shtml>, 24.2.2004).

⁴ The Asia-Pacific has the PDC, Personal Digital Cellular, standard that offered 'always-on' data communication and 28,8 kbit/s packet-data transactions. PDC offered also a possibility to send few seconds videos throughout the network

⁵ TDMA (Time division multiple access) uses IS 54 and IS 136 standards and as a digital network it possibilities the

development roots complicates the providing process of mobile services in the 2G world

Mobile value chain, value generation and strategic networks create a robust based to analyze mobile multimedia service's creation and development. During the mobile services development time we noticed the pilot's to be deeply absorbed in the service development process. This opened up the need to deepen the development process analysis. The improved tool to deepen the mobile multimedia service's analyze were got from the article of Barnes (2002) to where he concerns on mobile commerce's specifications, possibilities and value creation. Barnes defines also the value chain model for m-commerce. Barnes (2002) value chain model comprises *two paths* where both are further divided into *three phases*, Figure 3. The first path describes the content development and the second one describes the infrastructure and services provided as well as needed by mobile commerce. The *content path* expands further into content creation, content packaging and market making phases, whereas the *infrastructure and services path* consists of mobile transport, mobile services & delivery support and the mobile interface & applications phases.

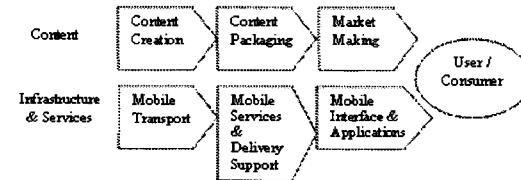


Figure 3. The m-commerce value chain, Barnes (2002).

Mobile markets and wireless value chain have got forms that were able to identify the basic functionality of the market. The wireless value chain have become a complicated entity that includes many members. Marconi's time value chain, to where the network operator handled the market, have gone. The number of members within value chain have increased and this development will continue together with technological development. Each of the new properties of the market requires new operational member in the value chain. Steinbock (2003) presents the

short messages sending and improved voice service's secrecy. Data transaction capability is 14,4 kbit/s. TDMA network has commonly being compared on Europe's GSM. TDMA were commonly used in Central- and South-America. CDMA is a circuit switched network that has been build upon analogue network. Uses IS-95, IS-95A and IS-95B standards. The latest version, IS-95B, is packet based. CDMA has development root continuing until 3G.

3G value chain to include 7 members⁶. Complicated market structure drives the business branch to specialize within the market. Technology or technical infrastructure should transform in the position to where the application development and the developers are not in the core of the value chain. Previous should point the development more on the applications value creation and market's needs satisfaction than pure technical items.

Mobile multimedia value chain as a business branch

Value chains and value chain analysis presents a model to characterize and analyze business' systematically. Value chain analysis emphasis on the partitioning of the business functions e.g. forms the required positions of producers so that the market can provide goods as a whole. Mobile value chain displays the natural partitioning of business functions from software creation until the use of customers. Value chain defines the major functions that must be brought and combined together so that the software applications can be constructed. Value chain-formed market structure does not itself define of how the industry is organized to provide these functions. Messerschmitt and Szyperski widen the scope of the value chain to become as a part of the ecosystem in the book of software ecosystem (2003). As a software ecosystem, the markets scrutinizes as completeness that includes supply side value chain, requirement side value chain and industry cooperation and competition.

Messerschmitt et. al (2003) presents software development to include eight natural business functions. The *industry consultant* analyzes and conveys the needs of a vertical industry segment or horizontal business functions. The results of vertical and horizontal analysis has to be taking into account in application software features and capabilities. The *business consultant* spread these results in practice when the same or similar applications have been applied in other companies. Basically the industry consultant focus on the needs of all firms and the business consultant focus on adapting application in certain firms. The *applications software supplier* develops the application. The application software supplier try to maximize the market share by attempting to meet the needs of the multiple end-user markets, and emphasizing the core competencies, like technical or project managerial skills in software development. The *infrastructure software supplier* is a member that has knowledge from wide range of applications, application's requirements and the needs of application developers. The infrastructure supplier-member benefits from economics of scale. The *system integrator* specializes in provisioning of software. System integrator role includes required software acquiring from application and infrastructure supplier, make all software to work

together, installs and tests software. System integration emphasizes more on the technical side of the development. The Business consulting on the other hand responds on the organizational and need issues. An *application service provider* licenses and operates the required applications. *Infrastructure service provider* purchases and operates required software and hardware infrastructures, like computers, operating systems, network and storages. Infrastructure service provider shares the common infrastructure over current applications. The business partitioning model shares the market on two parts. The upper part is a composition of software application development. The lower part defines and describes the requires infrastructure of the development.

Value chain Implications from the mobile multimedia development

The research cases comprised from eight companies that create mobile multimedia service pilots. The companies were mostly small start-ups. Only few companies have experiences from mobile value-added services business. These case companies were monitored from the content creation level until the product launch level. The generated ideas for the services and contents were from a wide area including professional business to business services, entertainment services, infotainment services and leisure services. Technological goal was a working MMS application, although many of the final products are also functioning on the Internet. The cases were examined several times during the eight-month development period. Afterwards the product success and pilot's networking were examined by the aid of internet and interviews.

The infrastructural side of the development

Considering on the Messerschmitt et al. (2003) model in the mobile application development, the market will be divided in two segments. The upper segment of mobile market consists on the application provide and business creation problems including also variables from business branch items. The lower part of the mobile ecosystem defines the infrastructural environment. The Infrastructural environment defines the developers technical environment to where they produce these mobile applications.

The impacts of infrastructural chosen in the mobile multimedia development

Infrastructural technology chosen in the mobile multimedia development focus on four key elements. These elements are operating systems, development platform, networks and terminals. The economical effects of infrastructural technology chosen appears by the network effects. The developers strive to find the environment that open up the possibility for positive network effect. Present infrastructural technology chosen on mobile multimedia development mainly become as a given solutions for case

⁶ Contractors, Equipment manufactures, Platforms, Chips, Application software, Location specific software and content aggregations.

companies. There has a possibility to make a chosen on certain areas, like on operating systems, but mainly the technical environments are same for all of which guides the development to follow up certain roots. Example of similar development environment arrives from the MMS production to where the GPRS network offers only feasible environment for the MMS-services delivery. This naturally forces the developers to choose GPRS-network and development tools.

The first key element on the infrastructural chosen is the wireless terminal's *operating system*, later on OS. Present market have three major operating systems⁷ for wireless equipment. Symbian OS bases on the mobile manufactures development and support, Palm OS bases on the Palms development and support and the Pocket PC based on the development and solution of Microsoft. The chosen of OS defines also the set of possible development platforms of which creates the second element of the infrastructural chosen. The *development platforms*, like Series 40 or Series 60 have on the Symbian OS based operating systems, create the environment where the developer make the application. Development platforms economical impacts becomes from the direct and indirect networks effect. Direct network effect measures the number of the users. Indirect network effect becomes from the community of the applications and development tool users, like from the community of Series 60 developers. Another example from the possible network externality arrives from Palm OS and from the development platforms of Palm of which have generated over 20 000 commercial applications⁸ for Palm PDA equipment. The third element on the infrastructural chosen arrives from the used *mobile network*. Mobile networks, like GSM,GPRS or WCDMA⁹,

creates autonomous environment that has own development environments. These networks are incompatible of which means the services and the development platforms to be network dependant as well as the provided applications are. Each of these separated standards create their own design and development environment. Nevertheless this fragments the markets, there has compatibility that have been made by the terminals of which supports more than one of the used standards. Present mobile phones may support GSM 900/1800/1900 and GPRS standards simultaneously¹⁰. This forms the last key element of the infrastructural chosen to be the mobile phone. Mobile phone's chosen become mainly defined by other arguments than developers free and rationale chosen. The chosen of operating system, the chosen of development platform and the chosen of network dictates the suitable terminal. The position to be a key element within infrastructural chosen becomes from the users and furthermore from the potential business environment. Operating systems, development tools and mobile networks may create good environment for mobile application development, but only the user and the use create the business around applications.

Analyzing 8 case companies from the infrastructural chosen point of view gave expected results. Mobile network defined to be GSM and GPRS for each of the developers of which follows the natural development line of Europe. The operating systems defined to become Symbian based, after the companies were made their development platform analysis. The development platforms rose up on the key role in the infrastructural decision process. The development platforms represents the environment to where the practical application creation were done. The development platforms also caused definitions of properties within applications. From the application development point of view, the development platforms direct application development stronger than what for example the business or the commercial need does. Mobile phones determined to become from the present smart phone markets supporting Symbian OS and Series 60-development platform.

Implications from the mobile value chain

Analyzing the eight case companies during their different phases on the service and content creation process gave varied and rich information of the mobile multimedia value chains, value networking and the roles of the participating

⁷ Smart phone's and PDA's operating systems have spread on separated development roots. Mobile manufactures supports Symbian OS based systems. Microsoft brought own mobile operating system, the Pocket PC, in the markets. Palm that has very popular on PDA's have brought Palm OS 5 with wireless and multimedia features. Linux has own mobile operating system named Mobile Linux. Further on have these operating systems got development platforms, like Series 40, 60, 80 or 90 have in the Symbian OS. (Liikenne- ja Viestintäministeriö, 2003)

⁸ Look at <http://www.palmsource.com/index.html>

⁹ The technical infrastructure's share on the modular structure offers a possibility to vary the structure when the development environment changes. These changes happens e.g. when the network, the terminal or the operating systems changes. Sanmateu, Paint, Morand, Tessier, Fouquet, Sollund & Bustos (2002) article 'Seamless mobility across IP networks using mobile IP' defines and introduces infrastructures, environments and

introduces the imputativeness to notice the technical changes on the service's architectures.

¹⁰ Like Nokia 6610i or SonyEricsson S700 or Siemens SX1 Sources:<http://www.nokia.com/nokia/0.8764.55637.00.html>, <http://www.sonyericsson.com>, http://www.siemens-mobile.com/cds/frontdoor/0.2241.hq_en_0_15803_rArNrNrN.00.html)

developers. We used Barnes (2002) presented m-commerce value chain in the analysis. Barnes's presented m-commerce value chain model supports both sides, technical and commercial, of the mobile services creation. The focus pointed on the content creation, content development and content packaging phases that presents the new forms of mobile service development. The model's role was to be the development analysing tool. The model has been introduced in the Figure 4.

The *content creation* phase includes the initial birth of the idea. During this phase the developers form the basic idea of the services and contents to be later introduced to the customer. Questions like, how to serve the customer, what customers' needs must be satisfied and where the value addition is built up during the process are introduced and asked.

The role of the *content development* phase is to further refine the visionary idea to become a feasible application in the mobile MMS context. During this phase the required answers are sought and solutions are found in the mobile service contexts for the *technical* and the *commercial* aspects of the required service supply. At the end of the content development phase is made the decision for possible further continuation of the development process. The studied cases showed that proper definition of this content development phase turned out to be important for the success of the whole project.

The *content packaging* phase includes technical solution of the product or service. As the idea has passed the content creation and the content development phases it has taken the form that technically can be carried out. This phase creates the software application for the mobile multimedia service.

The findings from the service production process varied between general and detailed. General findings brought subjects that were typical for all pilots. These typical findings presents the general development phenomena of the mobile multimedia service development. The dependence of the technology and the awareness of mobile service creation's possibilities were typical for the general mobile service development. Detailed development findings turned on the certain technical solutions within mobile services, like flash-presentation's implementation or the combining of text, sound and pictures. Both of these findings, general and detailed, have technological background that lead the development into situation where content creation, content development, and content packaging phases work closely together. The finding follows Barnes's m-commerce value chain model, but only in general level. Mobile service development practices required further development and the m-commerce value chain were shaped up to come on the form that has presented on the figure 4.

In Figure 4 presented model shapes and smelts the content phases to work closely together. Development process open up a need for a new phase. Content development phase were taken in the model to cover original idea's and mobile possibilities space in the development. The content creation, content development and content packaging phases creates the natural and required environment for the mobile service development. This new formed and together smelted phase *guides* and *lines* the possibilities and capabilities of mobile service development.

The last driver of the mobile value chain, the market making phase, includes information from the present and from the forthcoming target markets. The case companies were not interested to find out the detailed information from the target markets. The main interest were emphasized on the applications and on the contents feasible development than business'. Business interest were mainly on the general side including total market development items than in detailed segment information. This market development information belongs more on the content development phases, indicated in Figure 4 with grey area than in market making phase.

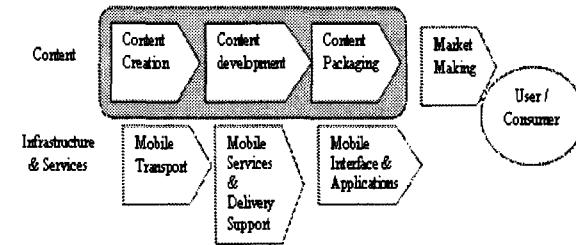


Figure 4: Mobile multimedia value chain

Implications for mobile business models

Those pilots and companies that were able to managed and reach the practical business in the market used three different kind of business models¹¹. The *first business model* includes traditional business forms. The MMS-services and sales materials were produced and presented to operators and application resellers e.g. third party members that took the practical services delivery and business management care. Mobile service providers role defined to become a subcontractor of these resellers.

¹¹ Business is the architecture for the product, service and information flows, and included a description of the various business actors and their roles and the source of revenue (Timmers, 1998) or the simplified that Rappa (2000) presents: It is the way to do business. More information of business models, please look Olla and Patel (2002), Tsalgatidou and Pitoura (2001) or Maitland, Bauer & Westerveld (2002)

Business were global e.g. each country creates their own markets and own resellers. Business revenue bases on intensiveness of the sales. The *second business model* become around partner development. Two ore more members created together a development project that goals on the MMS-service's development. MMS-service developers develops services together so that there has properties from each of them. Usually there were a combination of content and technical development of which also shares the separated interest of developers to be autonomous apart. Developers goal were to create a product that solves their own areas problems. Business revenue becomes from the extended resources, lower risks, from the potential of the forthcoming product sales, from the increase of the knowledge of the developers and from new IPRs. The *third business model* become around Internet's and mobile network's convergence. this model used strengths of both of these networks. Internet with PCs were used in a role of improved data transaction and processing capabilities. Mobility were join in the model by the payment process and a channel that supports services traditional ideas. Mobility brought new features for Internet products, like extended reachability, ubiquitous access or sophisticated payment processes.

Conclusion

Mobile multimedia development's value chain and development practices based on the experiences of this study have got a form that has been presented in the Figure 5. The picture brought out the impact of technology and the difficulty of adding content in the digital form. Technology dependence and difficulty of the digital content creation drives the market to follow on the technical development roots and passes the business and the value generation over. This kind of business behavior belong on the early stage business that still works with the practices established and development. Mobile value added business and applications creation still belong on the early stage business area. Customers are leaning to use mobile services in their daily living. Similarly the service developers are learning to create these services. Later on will the efficiently of the creation improve.

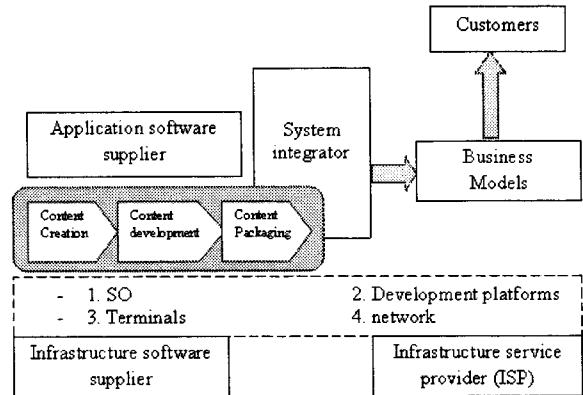


Figure 5: Mobile multimedia development's ecosystem

Mobility and mobile data services have approximately ten years history. During this ten years time the market have become a complex combination of many separated members. There have operators, application developers, programming languages, development platforms, customers etc. of those work together to create valuable services for customers. This market and market's application development has four chosen to be made. The operating system, the development platform, the terminal and the network lines the application development. Presently the value generation in the mobile services concretized via mobile multimedia services. These mobile multimedia services includes possibilities and limitations on technology, art, and music, as well as on information connected to databases. All these elements have various possible ways to be combined in a MMS application and various of ways to generate business. There were identified three separated business models; traditional sales, partner model and network convergence model. Nevertheless the primary interest should not be in the business model, but in the value generation for customers that will lately show up the success of this business branch. For market point of view it would be recommended to point more interest on the value generation of services. Sophisticated services serve customers well, but unsophisticated will drive the market in the chasm.

Mobile multimedia service's development within companies brought out the importance and dependency of technology. The development processes such as the idea development were compound in the technical possibilities of current mobile technologies. The developers value chain got three key drivers that smelted together during the development process. These drivers: content creation, content development and content packaging, formed out to become the critical drivers of MMS development and production. Knowledge within these drivers opens the possibilities of mobility and mobile services. The

development time of the services brought out also the similarity of the problems within companies. During this 8 months period, the developers were mainly working with the same problems albeit their service ideas differs. Same problems highlight the underdevelopment of the mobile market and mobile value chain. The roles, technical tools and platforms have not reach so sophisticated level that the market would be able to work fully efficiently. Well defined value chain creates the environment to where the market members have clear roles and duties and to where the competition concretes on the capability of own areas management.

REFERENCES

- Amit, R. and Zott, C. (2001): Value creation in E-business, Strategic management journal, 22, 493-520.
- Barnes, S. J. (2002), The mobile commerce value chain: analysis and future developments, International Journal of Information Management, 22, 91-108.
- Buellingen, F. and Woerter, M. (2002): Development perspectives, firm strategies and applications in mobile commerce, Journal of Business Research, 5830.
- Cave, M., Majumdar, S. K. and Vogelsang, I. (2002): Handbook of telecommunications economics, Elsevier science, Netherlands 2002.
- Gerstheimer, O. and Lupp, C. (2002): Needs versus techonolgy-the challenge to design third-generation mobile applications, Journal of Business research 5800/2002. In Press
- Hämeen-Anttila, T. (2002): Mobiilipalveluiden tuottaminen, Tummavuoren kirjapaino 2002. In finnish.
- Katz, M. L. and Shapiro, C. (1985): Network externalities, competition and compatibility, American Economic Review 75, 424-440.
- Kothandaraman, P. and Wilson, D., T. (2001): The future of competition Value-creating networks, Industrial marketing management, 30, 379-389.
- Kothandaraman, P. and Wilson, D., T. (2000): Implementing relationship strategy, Industrial Marketin Management, 29, 339-349.
- Liikenne- ja Viestintäministeriö (2003): MONA-Mobiilipalveluiden kehittämishohjelma, Liikenne- ja viestintäministeriön julkaisuja, 47/2003. In Finnish.
- Olla, P and Patel V. P. (2002): A value chain model for mobile data service providers, Telecommunications Policy, 26/2002, 551-557.
- Paavilainen. J. (2001): Mobile business strategies: Understanding the technologies and opportunities, IT-press, Great Britain 2002.
- Porter, M. E. (1985): Competitive advantage: Creating and Sustaining Superior Performance, the Free Press, New York 1985.
- Maitland, C. F., Bauer, J. M. and Weterveld, R (2002): The European market for mobile data: evolving value chains and industry structures, Telecommunications Policy, 26/2002, 485-504.
- Möller, K. and Svahn, S. (2003): Managing strategic nets- A capability perspectice, Marketing theory articles, Vol. 3 (2), 209-234.
- Li, F. and Whalley, J. (2002): Deconstruction of the mobile telecommunication industry: from value chains to value networks, Telecommunication Policy, 26/2002, 451-472.
- Rappa, M. (2000): Managing the digital enterprise, Business models on the web, <http://ecommerce.nscu.edu/topics/models/models.html>, 17.8.2001.
- Sabat, H. K. (2002): The evolving mobile wireless value chain and market structure, Telecommunications Policy, 26/2002, 505-535.
- Sanmateu, A., Paint, F., Morand, L., Tessier, S., Fouquart, P., Sollund, A., Bustos, P. (2002): Seamless mobility across IP networks using mobile IP, Computer Networks, 20/2002, 181-190.
- Shapiro, C. and H. Varian (1999), Information rules: A Strategic guide to the network economy, Harvard Business School Press, Boston.
- Steinbock, D (2003): Globalization of wireless value system: from geographic to strategic advantages, Telecommunications Policy, 27/2003, 207-235.
- Timmers, P.(1998): Electronic commerce -Strategies and models for business to business trading, John Wiley&Sons LTD, Chichester, United Kingdom.
- Tsalgatidou, A. and E. Pitoura (2001), Business models and transactions in mobile electronic commerce: requirements and properties, Computer Networks, 37, 221-236

User Experiences on Combining Location Sensitive Mobile Phone Applications and Multimedia Messaging

Jonna Häkkilä

Nokia Corporation

P.O.Box 300, Yrtipellontie 6
90320 Oulu, Finland

jonna.hakkila@nokia.com

Jani Mäntyjärvi

VTT Electronics

P.O.Box 1100
90571 Oulu, Finland

jani.mantyjarvi@vtt.fi

ABSTRACT

Quickly emerging usage of multimedia messages offers new approach for mobile and collaboration between users and services by providing a mature and easy accessible technology. This paper investigated possibilities of location related messaging, where pictorial and textual information are combined. We present a model involving contextual information delivery for a mobile user, and evaluate it by user tests with location sensitive multimedia messaging representing different application categories. The results show that the functionality and social acceptance varies between different message categories. It is suggested that distinct information elements should employ sharing and access right management, and a commonly agreed categorization system is required for successful information filtering.

Author Keywords

Multimedia messaging, mobile communication, location, context-awareness, mobile applications

ACM Classification Keywords

H5.m. Information interfaces and presentation (e.g., HCI): Miscellaneous.

1. INTRODUCTION

Mobile phones have become an integrated part of our everyday life. In addition to conventional voice call messaging is a strongly represented communication mode with mobile phones. Highly adopted short messaging services (SMS) messaging is a common interaction method among various user groups, especially teenagers, who use messaging not only to chatting, but also to coordinate media and times to interact, revise and adjust arrangements [7]. SMS has recently been accompanied with multimedia messaging services (MMS) offering an easy accessible technology for exchanging pictorial information between mobile phone users and different services.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MUM 2004, October 27-29, 2004, College Park, Maryland, USA.
Copyright 2004 ACM 1-58113-981-0/04/10...\$5.00.

Exchanging pictures online enhances the communication and creates new forms of mobile data and collaboration between mobile device users. So far, the research for the user behavior and potential applications related to multimedia messaging has been negligible. Mobile technology capable to pictorial information exchange has been found potential for children's collaboration during children's activities in story telling, adventure gaming and for field trip tasks [3]. As the MMS technology is still in the early adaptation phase, the ways users utilize it in mobile communication are still looking for their form. Research indicates that multimedia messages are used in various communications between people [11] and to enable novel applications, which include, e.g., tagging multimedia (MM) documents with location information during document creation [[17]], and expressive messaging using avatar animations [14].

In this paper, we propose novel type of MMS functionality obtained by combining MMS technology, positioning of mobile phones and some location based mobile applications. Particularly, study presented in this paper focuses on user experiences. Study is carried out with a mobile phone. The location based applications linked to MMS include notification, reminder and presence that are potential ones to support location sensitivity [5, 6, 10]. A simple model for combining location sensitive mobile phone applications and multimedia messaging to novel type of MMS functionality is also provided.

The study is based on our earlier research work providing background knowledge for model development and for selecting the application categories [12, 9].

2. BACKGROUND

Previous research on the location-aware mobile devices has introduced a number of studies on applications such as shopping assistants and tour guides, see for instance [2, 4]. In [5], user's personal data, such as calendar notes, are combined with location information and used for location-based messages to the user. Typically, the research so far employs specific devices, which are additional gadgets the user has to carry with, or equipment modified specifically for the purpose, for instance a PDA with an additional GPS module.

Current mobile phones are able to determine their own location information, and applications for sharing it exist. Information of the current physical distance may provide useful data e.g. for time management. Also, many of the device features vary according to the usage situation: mobile phone ringing tone profile setting to silent for meetings and loud for outdoors represent a common

usage case. Users' availability and devices' status offer important information when interacting with other people. If another person's mobile phone is set silent, one can reason that the person is unable to have a conversation at that moment. Availability information has been used in context-aware phonebook and context call [15, 16].

MMS have potential to combine the benefits of synchronous and asynchronous communication. When an MM message is received in the certain place, a user is notified of receiving the message. Notification can be set to occur according to her/his personal settings. Thus, (s)he has opportunity to react immediately. However, delayed reading of the message may not be as crucial for the user as with a text-based message. The pictorial information enhances the possibilities to memorize the connection to the message content, for example, the place is easier to recognize. Typically, MMS also includes other than pictorial information, for instance text or audio. It can be accompanied also with other type of information, such as various applications, which can be tied to a place in which the picture is taken and to an object it represents, as proposed in this paper.

The applications chosen for our study are notification, presence and reminder, as they form a comprehensive group of applications relevant for a mobile user. To be straightforward to employ into everyday usage, the selected applications offer intuitive cases for end-users and have potential for real life applications for large user groups. Previous results on context-aware functions for mobile phone applications indicate that reminders and notifications are perceived functional and valuable from the end-user viewpoint [12]. Here, location sensitive applications are combined with MMS information to novel type of MMS functionality. Integration of elements is described in a model.

3. MODEL

To examine the user experiences and acceptance of novel type of MMS functionality obtained by combining location sensitive mobile phone applications and multimedia messaging we have created a simple model. Schematic of our model is presented in figure 1.

In the model a user (sender) explicitly adds a recipient, the location, in which the MM message with application functionality is triggered, MM document, e.g. picture, describing an object (taken in certain location) to which application functionality is tied to, and type of phone application, of which functionality are attached to MM message. Applications include presence, reminder and notification, which represent potentially useful location or context based applications [[18], [19], 12, 5]. The user sends the MM message composed. Finally, recipient receives a MM message enhanced with application functionality.

Now, there are various types of information attached to a MM message; pictorial, location and application specific. This is a novel concept generating research questions on user perceptions and reactions when receiving and interpreting such a message. In our study the focus is on examining user experience when (s)he receives such a message and on projecting results to requirements of technical implementation.

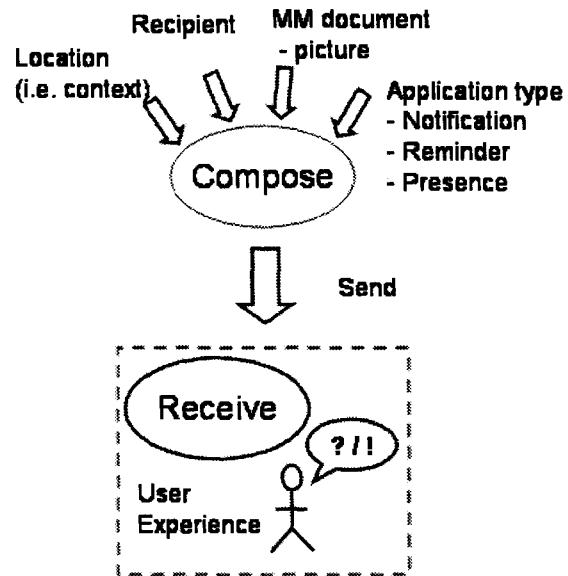


Figure 1. Model of using multimedia messaging for location sensitive mobile phone applications.

4. USER TESTS

4.1 Method and Participants

The number of subjects participating into user test was 9 (3 male, 6 female). The subjects represented different fields of study or work, and they all were mobile phone users. Two of the nine users reported to be familiar with the idea of the location based messaging, and in addition three had briefly come across with the concept. The test session with each subject lasted for 1.5-2 hours.

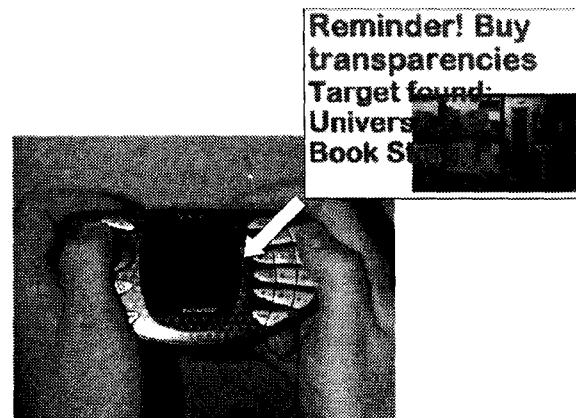


Figure 2. Location sensitive reminder received as an MMS.

User tests involved location sensitive MMS prototyping within a real life environment, interview and a written questionnaire. Tests were conducted at a university campus area, where the subject performed a walk along predetermined path. Thinking-aloud protocol was used as subjects were encouraged to talk their

perceptions during the walking route where the messages were received. The scenario used for the test was as follows:

You are going to meet your friend Marko at 12.00 in the room TS366 as you have to do a seminar presentation together. You have promised to bring transparencies, on which you will print your presentation. You have to buy these on the way. The time is now 11.45. You have a location sensitive mobile phone with you! In the beginning of the interview, the subjects were given a paper sheet with pictures of received messages as a reminder and support for their commentary.

4.2 Test Set-Up

The subject was given a MMS capable mobile phone (Nokia Ngage) for receiving messages, figure 2, and (s)he was accompanied by a monitor during the walk to whom (s)he could tell his/her perceptions during the experiment. The MM messages were sent by a test organizer to be received at the locations marked in the map, figure 3. Figure 3. illustrates the walking route on the campus area and MM messages received at each point along the path. The length of the walking route was altogether approximately 1km.

The received MM messages, illustrated in figure 2, are categorized to presence, reminder and notification types, see table 1. Notifications are categorized by their function to commercial and non-commercial information, as well as private and public notes. Contrary to the other messages, message no. 4 is an advertisement of a shop that was not located on the walking route but nearby, out of the sight. The visual information included in the messages 1, 3, 4, and 6 was aimed to enhance the mapping the information into the specific location, and to help the users to orientate themselves in the physical surroundings. In the message 5, the accompanying image considered the oncoming happening, and with message 2, the purpose of the appearance was to focus user's attention to the advertisement.

Table 1. Location sensitive messages used in the study

No	Category	Function
1	Presence	Notifies of presence of a contact in user's phonebook
2	Notification	Commercial information; Public information; notification of a current event, located on the walking route
3	Reminder	Personal reminder
4	Notification	Commercial information; Public information; notification of a shop, not located on the walking route
5	Notification	Noncommercial information; Public information; notification of a current event
6	Notification	Noncommercial information; Private note; set by a specific person to be received by a certain person

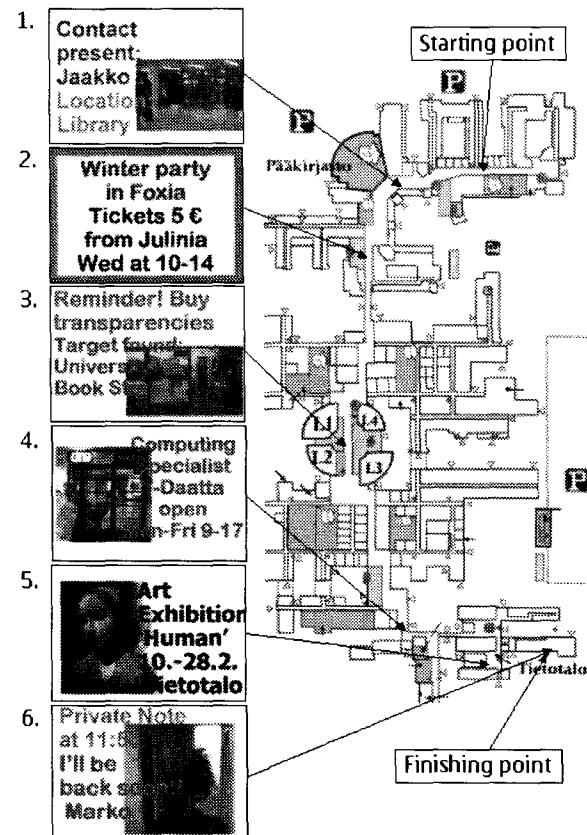


Figure 3. Messages received on the walking route at the university campus area.

5. Results

Results of the user tests are processed from two viewpoints. Firstly, user acceptance and general comments on various message types are examined. Secondly, subjects' feedback is evaluated in respect to the presented model, i.e., each category and particular characteristics are examined separately. In the following, female and male participants are referred with F and M, respectively.

5.1 Most Liked and Disliked Functionality

Subjects were asked to comment on each message and select a message, which they especially liked and disliked. The collection of answers is presented in figure 4. The sum of answers exceeds the number of subjects (9) as some subjects commented on more than one message.

Figure 4 shows that some messages are perceived strongly positive (message no. 3, 6) or strongly negative (messages no. 2, 4). Messages 1 and 5 awoke mixed comments. Positive feedback (message no. 3, 6) indicates that personal reminder and a private notification are seen as potentially functional and meaningful applications, and they were described as *'useful* and *'useful'*. Arguments for personal reminder included such as '*'Useful, right time and right location'* (*Subject #8*) and '*'Relevant, since I have a*

bad memory' (#6, #9). Arguments for private note included e.g. '*It is nice to know that Marko remembers our meeting and will be coming [to the meeting place]*' (#6). Negative feedback (message no. 2, 4) indicates that MM messages commercial information or public notifications were perceived unnecessary and annoying. They were experienced as '*spamming*', and main concerns related to overwhelming and uninteresting information flood and unnecessary interruptions, for instance '*[This is] spam advertisement. Exhausting to watch all the time what I just received*' (#1), and '*If every shop will send messages, how can I do anything else but reading them?*' (#8).

Messages no. 1 and 5 received both positive and negative comments. Presence information was perceived useful for social collaboration '*Nice to know if friends area around*' (#2), but on the other hand it aroused privacy concerns: '*Is my location going to be told to everyone?*' (#2) and '*One has to be able to turn this off*' (#1). Message 5 included a notification of a current event, and although it was regarded as spam, it was also perceived more interesting and acceptable than commercial advertisements.

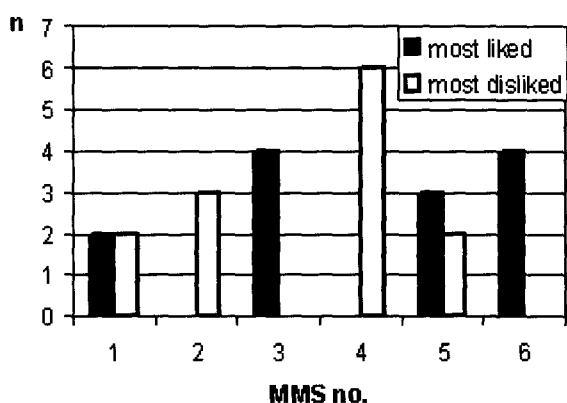


Figure 4. Number of references given for each multimedia message to the question 'Was there a specific message which you especially liked/disliked?'

The visual information provided in MM messages was perceived to support orientation in the current location that was considered important. Particularly this was useful when confirming the right meeting place (message 6) and establishing the connection between reminder or notification content and the physical environment, as commented for messages 3 and 5. The other perceived benefit information was the additional memory support the graphics gave, as the messages could be reviewed later on.

5.2 Assessing Application Groups: Presence, Reminder, Notification

Subject's comments on each message are grouped in message categories *Presence*, *Reminder* and *Notification* in the following sections.

5.2.1 Presence

The comments mentioned for message 1, i.e. informing the user of Jaakko's presence, are collected into table 2. The most perceptions relate to privacy concerns. Even as the application is generally perceived as a useful functionality, there are immediate questions of if the function is bi-directional and whom the presence information is shared with.

The concerns related to system infrastructure and application functionality relate to the time and distance range of the application. Timing is critical, as the information usually has value to the user only when received close enough to the other person. This is affected also by the distance from which the presence notification is given. Another issue is whom the presence information is shared with. Selection of sharing parties requires some kind of trusted or privacy filter, which can be defined by the user. Allowing information sharing by subscribing to the service was also mentioned.

Table 2. Comments for message 1.

Comments for Presence message (MMS #1)
Good information. Hopefully Jaakko gets the same message? How far is this scanning? (M1)
Does Jaakko also know that I'm close, or can the messaging work one direction only? Can I have influence on who is getting my information? (F2)
Most probably I wouldn't give my information to others. (M3)
Annoying. (M4)
This is cruel. (F5)
It might have been irritating if a lot of other contacts here nearby to receive all the messages. Perhaps it would be better to have a choice who to look for. (F6)
Good service, one can use this for occasional meetings. (F7)
I'd like to have it only if I've subscribed to this information or if they have different sound (alarm / message received sound). (F8)
If you want to read the messages right then, this should be timed exactly at the right position or little earlier. Depends where the phone is, is [the information] critical or is the message useless. Better functionality in wrist watch or glasses than in phone – one wouldn't have to dig it from anywhere. (M9)

5.2.2 Reminder

Comments on the personal reminder, message 3, are presented in table 3. Reminder functionality is seen as valuable and usable feature. Questions here relate to the reliability of the application: *Who knows if the paper shop sells transparencies, or what happens if you forget to go by the specific place*. Also, the collaboration aspect was pointed out as setting reminders for someone else was considered. Reminder type messaging emphasizes the role of the sending entity and triggering conditions. Sharing reminders and setting them to others requires access to user specific information elements in a model (figure 1).

Table 3. Comments for message 3.

Comments for Reminder message (MMS #3)
Good reminder with location information. How easy this is to set? Maybe only a reminder of the place [would be enough]. (M1)
Reminder is quite good. Have I set this up myself, or maybe 'Marko' set it? It may be irritating if others can remind you too much. Reminder itself is good, but if the shop only advertises itself, that would be annoying as I see the shop anyway. On the other hand, if I'm in a new environment, the information might be welcome. Can the ads be filtered, for instance so that only the places which I hadn't visited would advertise to me? Is there any limit for investigating consumer behavior? (F2)
If reminders are location sensitive, is there a risk that one forgets the place and never get the message or reminder? (F2)
Function, which I would find useful. (M3)
Good. Most important [message]. (M4)
Good! (F5)
Very nice to remind me! Who decides that transparencies can be bought in a paper shop? (F6)
Good service for own use. Bad thing is that you become very dependent on the phone when everything is in its memory. (F7)
Very useful. (F8)

5.2.3 Notification

The comments on notification messages are here presented so that comments of messages 2, 4 and 5 are grouped (table 4), as they received very similar comments. These messages can be seen as 'Public notifications', as they were not directed to any specific user.

Message 6 can be treated as 'private notification' as it was directed to the subjects and was not to be received anyone else. Comments on message 6 are gathered into table 5.

Messages in notification category gained lot of doubts and criticism. The subjects who were ready to receive notifications only accepted it conditionally, as they wanted to be able to select what kind of messages they would receive. Spamming with unnecessary advertisements and information flood were common concerns, and filtering was suggested to prevent undesired messaging. There were no differences in comments referring to the distance if the message sender was on the sight of the user.

Users' perceptions on acceptability and usefulness are significantly higher with private notification than with a public notification. Location's role as a trigger was emphasized, as the message was important only if the user entered to the specific location. Otherwise receiving the message was not compulsory. This feature is different to location sensitive reminders, as reminders for their functionality must go on even though the location trigger would not match.

Table 4. Comments for messages 2, 4 and 5.

General comments for Public Notification (MMS #2, 4, 5),
Irritating if [the information] comes as a message, [as] one has to always check what was received. I wouldn't use for instance in a city (spam). (M1: #2, 4, 6)
It would be good to be able to turn the ads 'off'. See also my answer for Message 3. (F2)
I wouldn't want to my phone (M3: #2, 5)
Annoying (M4: #2, 4)
Personalization would work better so that not all [existing] ads would be received (ads about anything). (F6: #2, 5)
Good service, but one should be able to restrict what kind of notifications one wants to receive. (F7: #2, 5)
Same as message 1: I'd like to have it only if I've subscribed to this information or if they have different sound (alarm / message received sound). (F8: #2, 4.)

Specific comments for MMS #2:

Message was late! Also, [it] should be received only at the time (Wednesday between 10-14 o'clock) or earlier. (M9)

How much in advance do these messages come? Is the same message coming every time I go pass the place? (F2)

Specific comments for MMS #4:

This could be handy if the information search was done in active manner. (M3)

This would work well for instance so that if I was looking for a [new] TV, I could set a search 'find TV shops'. The phone would then notify me when such a shop was nearby (for instance in a foreign city). (F7)

Specific comments for MMS #5:

How about copy rights [with this type of messaging]? (M1)

When and where? Is this [exhibition] really going to be on? (M9)

Table 5. Comments for message 6.

Comments for Private Notification (MMS #6)
Good information. On the other hand, this would have been done previously with [sending] a [text] message. (M1)
Can I get in contact with Marko if needed? Do I get the message if my phone is not turned on? What is the delay? Good, if I can ask information about Marko's location. On the other hand it is oppressive if someone could get my own information. (F2)
Interesting application. (M3)
Good. (M4)
Good service. Quick and comfortable way to inform of little changes in plans. (F7)
Very useful. (M8)

5.2.4 Perceived Message Categories

Subjects' perceptions of different message classes were evaluated with a question '*How would you categorize the previous messages?*'. The most typical way of categorizing messages was to distinguish who had sent the message. The categorization based on the following classes was done by 7 of the total 9 subjects (subjects no. 1, 3, 4, 5, 6, 7, 9):

- Own, self-set messages
- Messages initiated by a friend or a personal contact
- Advertisements

Within these answers and the category 'Advertisements', three subjects (5, 6, 7) distinguished two subclasses, 'Advertisements' and 'Current events'. Subject F2 proposed two ways of categorizing the messages:

- Public / Private
- or
- Entertainment / Benefits / Personal.

Subject F8 combined the focus audience and temporal factors dividing the messages as follows:

- Things which are useful personally for me right now
- Messages personally for me, but not necessarily now
- Messages for everybody who is nearby

Interesting detail in answers was that when subjects mentioned messages containing location information of a friend (like in message #1), these type messages were described also as general location notifications, i.e. not for indicating presence or nearness. This phenomenon was described e.g. with words '*Who is where – messages*' (subject 6).

6. DISCUSSION

Generally, subjects' perceptions of location sensitive messaging were quite undivided. Privacy concerns were dominating when subjects presented their negative expressions of location sensitive messages with presence indicators. Information flood and unnecessary advertising raised doubts especially with public notification messages. These results are consistent with previous research [9].

Pictorial information included to multimedia messages was perceived useful as it helped users to orientate themselves in the physical environment and connected the message content to the physical surroundings. It was also seen as memory support as the message could be reviewed afterwards.

Authors' proposed categorization to reminder, notification and presence was relatively close to what the subjects' perceived. The dominating argument for categorizing suggested by the subjects was to classify messages according to the message initiator: user, friends or equivalent, and an advertising entity. Also, there was a tendency to distinguish subcategories between advertising shops and enterprises versus current events even if they were both commercial notifications. Some subject perceived the messages informing of a friend's location more as general location indicators than remote presence, which perception they expressed when categorizing the messages. There was also a trend to categorize the messages to private and public according to their focused receiver. User defined message classification is coherent and indicates that people have strong sense of different functions of the messages. As obtained from the results, the message

categories have differentiating priorities, and the acceptability of them varies.

The characteristics of each message category can be taken into account when considering them in respect to the model and critical points can be found. Thus, the model can provide guidance e.g. in application design phase, as special care can be devoted to specific parts of the application interaction design. The following findings can be derived from the user test results: When evaluated in respect to the presented model, the contextual triggers for reminders were found to be critical. Presence applications also require acceptance from both the sender and the recipient in order to ensure users' privacy. With public notifications, the target group needs to be especially considered in order to avoid unnecessary and irritating spamming. When refining the model, adding a filter for privacy or for profiling the user should be considered.

Location based messaging can be implemented either as a pull or push based system, which also sets specific requirements for the application design. Pull type of messaging, where user makes request to gain information, reduces the amount of unnecessary messaging, but on the other hand requires active actions from the user and since more effort. The former suits better for 'search information' rather than to 'showing present information' types of applications.

The reliability needs to be considered in application design and is emphasized in situations, where the proper functionality of an application is needed even without location sensitive information. For instance, the reminder should still go off at some stage, even if the location is not optimal.

The requirements the notification application category sets to the infrastructure and service must be carefully considered, as there exists a potential risk to implement disturbing functions. Filtering needs to be employed, and successful implementation requires consistent and commonly accepted categorization of services and enterprises. In addition, introducing a search function requires some kind of database of available services and commonly agreed vocabulary with which the services can be mapped to user's profile or personal needs.

7. CONCLUSIONS

MMS has recently been introduced to the wide audience of mobile phone users, and they enable development of new application concepts and mobile services. In this study, we have explored user experiences on combining location sensitive mobile phone applications and multimedia messaging to novel type of MMS functionality. The selected message categories under investigation were presence, reminder, and notification (public and private), which were selected as they were seen to provide a representing sample of potentially useful and realistic location-related messaging applications. The function of this paper was two-fold. Firstly, we have presented a simple model, which defines elements for composing multimedia message enabled location based application functionality. Secondly, the users' perceptions of location sensitive multimedia messaging were investigated with user tests.

Reminders, which are activated in certain contextual conditions, here by location, were perceived as the most useful application and gained high acceptance, although also concerns related to the inferring logic in which conditions the reminder was to be

triggered. Messages informing of another person's presence were felt practical and fun, although these opinions were mixed with concerns of privacy. Private notes set by certain person targeting to a specific user were considered useful, and value was seen especially in social cooperation when rapid changes in plans occur. The main concerns with the location sensitive applications were unnecessary advertisement and privacy threads. Subjects also valued a possibility to profile and filter messages.

Subjects' perceptions of message categories followed the message initiator, as the classes were divided as follows: the user him/herself; friend or equivalent; and an advertising entity.

When examining the results in respect to the model including elements for multimedia message enabled location based application functionality, it was found that different message categories emphasize different elements of the model. This can be seen as useful information for application design process.

Since there is clearly a need for filtering and profiling the messages, there is also a need for a commonly agreed categorization for different message types. Collaborative distributed mobile data management is also needed in order to enable seamless co-operation between the service providers, operators, and application developers. In addition, there is a need to manage the information sharing and access rights for distinct information types. This is required for instance with mobile collaboration tasks, writing private and public location specific notes, and sharing presence information with other users.

Next research steps include refining the proposed model and more extensive user testing on the topic. Implementing a prototype with technical features implemented would provide more usability information, and is considered as future work.

8. REFERENCES

- [1] Bharat, R., and Minakakis, L. Evolution of Mobile Location-based Services. *Communication of the ACM*, 46, 12 (Dec 2003), 61-65.
- [2] Bohnenberger, T., Jameson, A., Kruger, A., and Butz, A. User Acceptance of a Decision-Theoretic Location-Aware Shopping Guide. In *Proceedings of the Intelligent User Interface 2002*. ACM Press (2002), 178-179.
- [3] Cole, H., and Stanton, D. Designing mobile Tehcnologies to Support Co-Present Collaboration. *Personal and Ubiquitous Computing*, 7, Springer-Verlag London Ltd (2003), 365-371.
- [4] Davies, N., Cheverst, K., Mitchell, K., and Efrat, A. Using and Determining Location in a Context-Sensitive Tour Guide. *IEEE Computer*, 34, 8 (2001), 35-41.
- [5] Dey, A.K. , and Abowd, G.D. CybreMinder: A Context Aware System for Supporting Reminders. In *Proceedings of HUC 2000*.
- [6] Dey, A. K., Salber, D., Futakawa, M., and Abowd, G. The conference assistant: Combining Context-Awareness with Wearable Computing. In *Proc of the 3rd International Symposium on Wearable Computers (ISWC '99)*. IEEE Computer Society Press, 1999, 21-28.
- [7] Grinter, R. E., and Palen, L. Instant Messaging in Teen Life. In *Proceedings of CSCW 2002*, 21-30.
- [8] Grinter, R. E., and Eldridge, M. A. y do tngrs luv 2 txt msg? In *Proceedings of the Seventh European Conference on Computer-Supported Cooperative Work ECSCW'01*. Kluwer Academic Publishers, 2001, 219-238.
- [9] Häkkilä, J. and Hexel, R. Interaction with Location-Aware Messaging in a City Environment. In *Proceedings of OZCHI 2003*, 84-93
- [10] Kaasinen, E. User needs for location-aware mobile services. *Personal and Ubiquitous Computing* 7 (2003), 70-79.
- [11] Kurvinen, E. Only When Miss Universe Snatches Me: Teasing in MMS Messaging. In *Proceedings of DPPI'03*, ACM Press, 2003, 98-102..
- [12] Mäntyjärvi, J., Tuomela, U., Känsälä, I., and Häkkilä, J. Context Studio – Tool for Personalizing Context-Aware Application in Mobile Terminals. In *Proceedings of OZCHI 2003*, 64-73.
- [13] Marmasse, N., and Schmandt, C. Location-Aware Information Delivering with commotion. In *Proceedings of HUC 2000*, Springer-Verlag, 2000, 157-171.
- [14] Persson P., ExMS: An Animated and Avatar-based Messaging System for Expressive Peer Communication. In *Proceedings of GROUP'03*, ACM Press, 2003, 31-39.
- [15] Schmidt, A., Stuhr, T., and Gellersen, H.-W. Context-Phonebook Extending Mobile Phone Applications with Context. *Third Mobile HCI Workshop*, 2001.
- [16] Schmidt, A., Takaluoma, A., and Mäntyjärvi, J. Context-Aware Telephony Over WAP. *Personal and Ubiquitous Computing* 4, 4 (Aug. 2000), 225-229.
- [17] <http://www.nokia.com/nokia/0,6771,59033,00.html>
- [18] <http://www.nokia.com/nokia/0,8764,43089,00.html>
- [19] www.sonera.fi

Digital Rights Management & Protecting the Digital Media Value Chain

Marvin L. Smith

HumanCentric Technologies, Inc.
111 James Jackson Ave, Suite 221
Cary, NC 27513
msmith@humancentrictech.com

ABSTRACT

Digital media that is readily & illegally distributed over the Internet and related digital networks has posed major problems for the members of the digital media value chain. Ubiquitous mobile communication devices such as media capable handsets and PDAs have made the problem even larger.

Technical approaches to controlling illegal distribution—commonly known as Digital Rights Management (DRM)—have been varied and inconsistent since the shift from analogue media to digital media; but in recent years, the Open Mobile Alliance (OMA) has made huge contributions to the efforts to standardize the DRM effort, especially as it pertains to the sharing of media using mobile devices. The OMA has released DRM Enabler Release 1.0 and 2.0. DRM 1.0 was a first attempt to apply control to digital media and DRM 2.0 is a more sophisticated continuation of 1.0, allowing the owners of digital assets to control their use while attempting to provide the end user with the perceived rights and privileges acquired over the evolution of digital media. While challenges lay ahead for the developers of DRM, OMA DRM is a move toward better standardized control of digital media.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MUM 2004, October 27-29, 2004 College Park, Maryland, USA.
Copyright 2004 ACM 1-58113- 981-0 /04/10... \$5.00"

Categories and Subject Descriptors:

C.2 [Computer-Communication Networks]; C.3 [Special-Purpose and Application-Based Systems]; D.2.12 [Interoperability]; D.3 [Programming Languages]; H.5.1 [Multimedia Information Systems]; I.3 [Computer Graphics]; K.6 [Management of Computing and Information Systems]

Keywords:

Digital media, Digital Rights Management (DRM), Open Mobile Alliance (OMA), forward lock, separate delivery, combined delivery, Rights Expression Language (REL.)

1. INTRODUCTION

Protecting the intellectual rights of the creators and owners of intellectual property has been a difficult task since early advancements in the printing press allowed for the mass production of printed documents, and legal efforts to insure compensation for such work date back to the first copyright laws in England in the early 18th century [Chiarglione 2003].

In the subsequent 300 years, and especially in the 20th century and present, these efforts have been complicated exponentially by the introduction and perfection of technology that allows the end user to exactly duplicate—and proliferate on a mass scale—all types of media content.

In the early days of recording and distribution of “analogue” media [Chiarglione 2003], the owners of intellectual assets had a greater advantage in controlling their revenue streams because of the way in which media was delivered to and consumed by the end user. That is, the media was limited to the device or medium that contained it and/or the device required to consume it and it could not be duplicated by the end user. Using music as an example: the phonograph record represented a song or collection of songs recorded on a vinyl disk. To consume the product, the end user was required to purchase the disk and have access to a device upon which to play it. At that time, duplication and further distribution of the sounds recorded on the disk by the end user was not possible. However, a couple technological generations later, music was also distributed via cassette tape. This advance created new problems for content owners because end consumers could copy content to other cassette tapes and then give or sell unauthorized copies to other users. Though the full quality of the original recording was not preserved and mass distribution was difficult, the ability to duplicate and distribute did exist and was

established in the minds and habits of consumers who have consequently become accustomed to such rights and freedoms related to the use, copying, and sharing of such media [Chiarglione 2003].

More recently, the fight by content owners has become more difficult because of the most recent generation of recording and distribution of music in the form of the compact disc (CD). CDs now allow the end consumer to perfectly duplicate the CD content in the same quality in the same compact disk form. But it doesn't stop there, users can now store the CD content to any number of storage devices and the content may be played on a number of players including PC media players and other more portable music players. The compact disc is the 'first example of digitized consumer media' [Chiarglione 2003] and since its introduction, practically every form of media is now available in digital form and most are duplicable and distributable via the internet and related digital networks.

Distribution (also known as "swapping" or "sharing") consists of users downloading files from P2P Web sites or applications which provide the individual with the ability to download complete songs, albums, music videos, movies, software applications etc. Practically any medium that can be digitized. Other means of file sharing include sending of files via email, transferring files via IM, transferring files wirelessly via IR and Bluetooth, and saving of pictures and other files directly from Internet Web sites.

In the 1990s, most unauthorized distribution of media was from "fixed" computer terminals. Now, with the advent and rapid evolution of the mobile communication sector, the mobile communicator using a phone or PDA with access to the digital mobile network (or even WiFi, if the handset is so enabled) can access and share practically the full array of digital files (images, photos, videos, streaming videos, and sounds) with anyone with a computer or mobile device with Internet access and using content-based communication applications native to the handset (email, IM, Multimedia Messaging Service (MMS), Enhanced Messaging Service (EMS), Infrared (IrDA), and storage to universal serial bus (USB) device).

Essentially, mobile communication allows digital media to be duplicated and superdistributed to hundreds, thousands, or millions of people—practically anywhere in the world—very quickly through "viral" distribution. Consequently, content owners are highly motivated to implement a solution to control content and insure compensation. Current approaches to control of pirated digital media are basically two-fold. The first is a legal approach rooted in the legacy of intellectual property rights and recorded media and is based on "levies"; the second, and focus of this paper, is a technology-based effort called Digital Rights Management (DRM).

Briefly, the original intent of "levies" was to protect the rights of the consumer while at the same time compensate the artist/owner of the media [Chiarglione 2003]. This was and is accomplished by collecting the levy on the sale of each device or product that may be used in the copying, sharing, or storing of any protected digital content. The proceeds are then distributed to the content creators/owners. In spite of its limitations, this practice is still in place and quite active in many countries. For example, Canada imposes levies "on blank digital media to compensate recording artists for acts of piracy." In addition, there is an effort to standardize such practices among "Interested countries," also known as the World Intellectual Property Organization [Kapica 2004].

DRM, on the other hand, is a technical solution being implemented in mobile devices by mobile handset manufacturers

at the behest of members of the digital media supply chain. DRM is "a system for protecting the copyrights of..[files]...circulated via the Internet or other digital media by enabling secure distribution and/or disabling illegal distribution of the data" [Webopedia 2004] In other words, DRM is a means by which content providers can prevent illegal distribution of copyrighted media files –and thus protect revenue streams –by encrypting and packaging media files, with rules that dictate the playback/distribution of the file.

One of the first efforts to employ a DRM-type solution was in the form of the Content Scrambling System (CSS) used by DVD distributors. In this approach, the DVD content is encrypted and can only be "decoded and viewed using an encryption key, which the DVD consortium kept secret [Wikipedia 2004]. As a part of CSS, DVD manufacturers signed an agreement with DVD consortium promising not to provide "features in their players such as digital output which could be used to extract a high-quality digital copy of the movie" [Wikipedia 2004]. Ultimately, CSS failed to insure the rights of the content owners without infringing upon the perceived rights of the end consumer. And subsequent laws protecting content owners have done little to prevent the production of systems that bypass CSS and other DRM tactics.

Until fairly recently, each industry has taken its own initiative in defining DRM and most industries –as exemplified by CSS – have implemented their own solutions. In response to the needs of all members of the media value chain and the need to standardize DRM, the Open Mobile Alliance (OMA) (created by a consortium of some 350 mobile device manufacturers and formerly known as the WAP forum), as a part of its umbrella effort to openly standardize interoperability of mobile communication worldwide, began in 2001 to create a DRM standard specification, which is now in its second version [Rourke 2004]. The OMA DRM standard applies to media content sent to and from mobile devices via any content-based communication channel.

2. OMA DRM 1.0

The OMA's first version of DRM –the OMA DRM 1.0 Enabler release – was issued in November of 2002 and is now supported on more than 50 mobile handsets [XML Cover Pages 2004] and provides basic protection functions for limited content value such as simple pictures, videos, animations, and sounds/ringtones [Rourke 2004]. In OMA DRM 1.0 Enabler Release, the OMA specifies three classes of DRM for mobile devices: Forward-Lock, Combined Delivery, and Separate Delivery.

Forward Lock is the first generation and most basic form of DRM for Mobile Devices. Essentially, when a protected file becomes resident on a device, usage of the file is unlimited but the file cannot be forwarded by any communication channel or sent to external storage devices under any circumstance [Openwave Development Network 2004]

Combined Delivery is a more sophisticated form of DRM for mobile devices. In Combined Delivery, the file is packaged with a Rights Object (RO) (ROs are delivered in Rights Expression Language (REL) which is in XML). When the file is resident on the phone, the RO dictates not only that the file cannot be forwarded via any channel [Openwave Development Network 2004] but it also dictates the number of times the file may be consumed OR the length of time the file is valid. Upon expiration of the file, the user is prompted to renew the license for the file [Openwave Development Network 2004].

Separate delivery is the third and most versatile of the 3 mobile DRM types. Separate Delivery separates the file from the rights required to use the file. This allows content to be consumed on a handset according to the rights provided to the user in the RO. The content can then be forwarded but the RO remains on the handset. Consequently, the recipient(s) to whom the content has been forwarded will be required to purchase the rights from the rights holder in order to consume the content. In this way, the content developer can safely distribute protected content, insured that when the content is consumed by the end user, royalties have been paid; in addition, the developer has potential additional revenue streams from the viral superdistribution of the file [Openwave Development Network 2004].

3. OMA DRM 2.0

In February of 2004, the OMA announced the release of the second version of its DRM standard specification (OMA DRM 2.0 Enabler Release) and has been releasing the specific requirements documents for the new version throughout the first half of the year. Most prominent mobile communication device producers and many related companies have already agreed to the OMA DRM 2.0 standards.

DRM 2.0 uses OMA DRM 1.0 as a foundation and in 2.0, many of the ideas are conceptually the same in that the content and Rights object will be in "DRM Content Format" in which the "permissions and constraints" for use of the content will be assigned. If the user agrees to the permissions and constraints, the consumer will receive the rights to the object and can begin use. The content may also be forwarded. [Buhse 2004].

Unlike 1.0, DRM 2.0 can handle premium content and makes use of newer more powerful handset and network capabilities. High bandwidth mobile networks are emerging in which carriers and content providers can benefit from the "release rich audio/video content and applications [XML Cover Pages 2004]. Concurrently, more and more mobile devices feature removable high-capacity storage cards and large, high resolution color screens which facilitate 'downloading and streaming [of] rich media' content [XML Cover Pages 2004].

3.1 Improved Security

DRM 2.0 dictates that valuable content can be entrusted only to a secure device on which the integrity of both content and rights objects can be insured. This is accomplished by a mutual authentication process between device and rights issuer (server) in which the server checks devices for trustworthiness. Since, "DRM Content is consumed according to the specified rights...., the value is in the rights and not in the Content itself." In DRM 2.0, Rights Objects will only work on devices authorized by the rights provider" [XML Cover Pages 2004] In addition, security will be increased to insure that content will not "leak out" during distribution. To do this, the DRM second release will make use of a "public-key encryption for protecting the symmetric keys used to encrypt content—a feature that is common in DRM technologies for PCs." It is also possible that DRM 2.0 could make use of 'digital certificates or cryptographic digests...for insuring the integrity of the content itself' [Rosenblatt 2004]. A third security method is to be determined.

3.2 Creation of New Business Models

With DRM 2.0, the end user can place protected content on multiple devices such as storage devices, music players, and other phones. This allows "new business models" [Rourke 2004]. According to Bill Rosenblatt [2004], the "the problematic concepts like 'device ownership' and 'backup' are left to implementers" and 'content owners are free to grant such rights or not, as they choose. Like DRM 1.0, usage limitation will be limited by "meter" which keeps track of the number of times a file has been used or the amount of time the file has been in use. Unlike, 1.0, users will be able to purchase "subscription rights for bundles of content." In addition, user will be able to buy content as gifts for others and have content sent with a pre-purchased RO. Another step forward will be the ability to support sharing of files via P2P. In DRM 2.0, the revocation status of protected content can be monitored by the right issuer; likewise, the "device can identify Rights Issuer revocation status" [Buhse 2004].

3.3 Advantages to End Consumer

Consumer have grown to expect positive experiences when dealing with downloaded files, and DRM will have to be flexible enough to allow the consumer to use the content in the ways in which the consumer has become accustomed, including placing the "content on any device they own" [RSA Security 2004]. According to Rick Welch, with RSA Security, "mobile operators, content providers and device manufacturers want to provide the most desirable experience for consumer through the creation of a network of trusted devices that make digital content more portable." Before buying protected content, the consumer may view files and determine their rights for consumption and "superdistribution" prior to purchase. After purchase, DRM 2.0 will allow the end user to move protected content along with the right object to (trusted) multiple external DRM devices (PC, PDA, other phones, etc) and later return them to the original device. The files may also be shared within a particular user domain, such as a circle of friends [Buhse 2004].

As in DRM 1.0, Separate Delivery, a consumer who has received the rights protected file may also send the file to friends who in turn will send it to friends, and so on. Each recipient can buy the rights to consume the file before forwarding or the file may be forwarded without purchasing rights. The caveat is that the recipient must have a trusted OMA DRM enabled handset in order to receive or preview/consume the file. In this way of insuring the ability for consumers to virally distribute the file, "content and service providers make money through "word of mouth" advertising [Buhse 2004].

Parallel to development of DRM 2.0, the Content Management License Administrator (CMLA), "a consortium whose members span device makers, software vendors and content providers [(the entire value chain)], announced its intention to build a licensing authority...and legal trust foundation for OMA DRM 2.0 in time to build devices...for the 2004 year-end holiday season" [Rosenblatt 2004].

4. MICROSOFT & MOBILE DRM

Microsoft, the most powerful player in the computing industry, has also made some attempts to integrate DRM into its

subsystems (such as Windows Media Player) and is stepping up efforts with the release of Windows Media DRM 10 and the future release of Next-Generation Secure Computing Base (NGSCB) (formerly known as Palladium) [Wikipedia 2004].

Microsoft also has a new DRM solution for mobile devices in an attempt to make mobile phone usage more similar to file consumption of a PC. File usage scenarios include services such as music purchase, music subscription, and movie rental. This solution is a “porting kit” that provides portable devices with the ability to play content that was previously protected on the Windows Media Rights Management Software Kit (SDK). The play of files is facilitated through a basic playback application (media player) that is loaded on a qualifying handset (i.e., with a secure clock, appropriate memory, etc) and communicates with rights-issuing Microsoft server. The mobile playback application is limited in capability. It can monitor number of plays and validity periods and maintain security updates. The mobile playback application can only play files and will not allow more advanced capability such as “license creation or content encryption” [Microsoft Corporation 2004].

5. FUTURE CHALLENGES FOR DRM

Mobile DRM has evolved at lightning speed. This is due in the most part to the fact that as content-rendering devices, mobile devices are young, relatively simple, and more malleable and shapeable than older more complex and more entrenched technologies such as PCs [Rosenblatt 2004].

Despite the relative youth of Mobile DRM and flexibility of DRM and mobile technology, critics say that DRM models for mobile applications are not flexible enough at this stage and should offer a broader range of features (such as “pay-per-use” features) and capabilities and should be mapped to desktop application DRM. With the exception of Microsoft’s attempts to sync mobile and desktop rights management, current desktop application rights management does not readily translate into the mobile environment. In addition, bridging the gap between fixed and mobile environments is difficult because in the current mobile environment, handsets are not always connected to the Internet as costs to do so are prohibitive [Dahlem et al. 2004].

Another impending problem DRM faces is the fact that there are several Rights Expressions languages (REL) currently available. RELs are vital since they are language in which Rights Objects are written and “describe licenses governing the access to digital content.” [Delgado et al. 2004]. Current RELs include Creative Commons, METSRights, MPEG -21, and ODRL (Open Digital Rights Language). MPEG-21 and ODRL are the most common in RELs in use and ODRL has been officially sanctioned by the OMA. Even though both of these primary RELs are complex and powerful enough, a problem arises simply because there are two primary RELs while ideally what is necessary is one REL language. One REL would insure interoperability and effective execution of future business models. One point of view is that the “use of different RELs could divide the network commerce in two separate factions” [Delgado et al. 2004]. To solve this problem, some experts are working on methods to make the two languages interoperable. [Delgado et al. 2004]. Finding a way for different RELs to be interoperable appears to be the most realistic options since—according to Karen Coyle [2004]—the “prevailing wisdom is that there is not (and probably never will be) a universal REL, any more than there will be a universal data format. This is especially the case because a rights expression

language exists in the context of a larger system and the nature of that larger system and its requirements determine the features needed in the REL” [Coyle 2004].

6. CONCLUSION

Mobile technology has provided the mobile communicator with the ability to readily copy, distribute, and share all types of digital media without recognizing or compensating the creator or owner of the media. And DRM is the primary means by which members of the digital media value chain are insuring just compensation for digitized intellectual property in the future.

DRM is relatively young and will continue to evolve based on improvements in technology and the desire to protect digital assets. It will be driven by the interplay between the content industry and mobile device producers and the OMA and other organizations such as ODRL and, more recently, a consortium of mobile operators including Vodafone, Orange, and NTT DoCoMo who have organized to better drive the development of mobile handsets by producers to insure service rich—and consequently, higher revenue producing—handsets accessed with very user friendly interfaces for acquiring digital assets.

7. REFERENCES

- BUHSE, WILLIAM, PhD. 2004. *OMA Secure Content Delivery for the Mobile World*. Open Mobile Alliance Presentation, Beverly Hills, CA. <http://odrl.net/workshop2004/prez9>
- CHIARGLIONE, LEONARDO. 2003. *The Digital Media Manifesto*. <http://manifesto.chiariglione.org>
- COYLE, KAREN. 2004. *Rights Expression Languages: A Report from the Library of Congress*. http://www.loc.gov/standards/Coyleresport_final1single.pdf
- DAHLEM, D., DUSPARIC I., and DOWLING, J. 2004. A Pervasive Application Rights Management Architecture (PARMA) based on ODRL. In *Proceedings of the First ODRL International Workshop*, 45-63. <http://odrl.net/workshop2004/paper/odrl-dusparic-paper.pdf>
- DELADO J., POLO J., and PRADOS J. 2004. Interoperability between ODRL and MPEG-21 REL. In *Proceedings of the First International ODRL Workshop*. <http://odrl.net/workshop2004/paper/odrl-polo-paper.pdf>.
- KAPICA, JACK. 2004. CD Levies Could Double, Group Warns. *Breaking News Technology Globe and Mail Update*. <http://www.globetechnology.com/servlet/story/RTGAM.20040421.gtccfda0421/BNStory/Technology/>
- MICROSOFT CORPORATION. 2004. A General Overview of Two New Technologies for Playing Protected Content on Portable or Networked Devices. <http://www.microsoft.com/windows/windowsmedia/drm/faq.aspx>
- OPENWAVE DEVELOPMENT NETWORK. 2004. Openwave White Papers: OMA-DRM. http://developer.openwave.com/dvl/support/documentation/white_papers/wp_oma_drm.htm
- ROSENBLATT, BILL. 2004. DRM Watch. <http://xml.coverpages.org/ni2004-5-31-a.html>
- OURKE, VANESSA. 2004. Open Mobile Alliance Takes Critical Next Step in Delivering Premium Content to Consumers Via Wireless Media

Devices OMA Issues Version 2.0 of its Digital Rights Management Enabler Release. DRM Press Release, Ketchup Public Relations. <http://www.openmobilealliance.org/docs/DRMPressReleaseFinal020104.doc>

RSA SECURITY. RSA Security Supports Open Mobile Alliance DRM 2.0 for Delivery of Secure Content. Press Release. RSA Security, Inc. http://www.rsasecurity.com/press_release.asp?doc_id=3337

WEBOPEDIA.COM. 2004. Definition: "DRM." Jupitermedia Corporation <http://sbc.webopedia.com/TERM/D/DRM.html>

WIKIPEDIA. Online Encyclopedia. 2004. Definition: "Digital Rights Management." Wikimedia Foundation. http://en.wikipedia.org/wiki/Digital_Rights_Management

WIKIPEDIA. Online Encyclopedia. 2004. Definition: "Palladium Operating System." Wikimedia Foundation. http://en.wikipedia.org/wiki/Palladium_operating_system

XML Cover Pages. 2004. "Open Mobile Alliance Releases Working Drafts for OMA DRM Version 2.0" <http://xml.coverpages.org/ni2004-05-31-a.html#highlightsVersion2>.

Bandwidth Optimization by Reliable-Path Determination in Mobile Ad-Hoc Networks

Sriram Raghavan[†], Srikanth Akkiraju*, Santhosh Sridhar

Department of Electronics,

MIT Campus, Anna University

*sriram1raghavan@msn.com, *srikanth_akkiraju@yahoo.co.uk

Abstract: A Mobile Ad Hoc Network is typically characterized by two sets of nodes, one set being more stationary than the other. It is observed that identifying this set of relatively stationary nodes critically facilitates the dual purpose of reducing the routing overhead and providing larger data bandwidth. It is therefore both interesting and expedient to study the behavior of a typical Mobile Ad Hoc Network and identify the sub-graph which can impact these improvements where the application demands are high. It is particularly enlightening when scaled to a larger network, as one of the substantial factors affecting cost in an Ad Hoc network is the routing overhead. In this paper, a method is proposed to determine, dynamically, the most reliable path for routing in a Mobile Ad Hoc scenario and ensure that packets are routed along that path to guarantee enhanced data traffic routed per unit time owing to savings from route-discovery and to decrease normalized routing overhead. The Mobile Ad Hoc Network is modeled as a graph with the mobile units denoting the nodes of the graph and the set of reachable units as its neighbors and identifying the weakly connected sub-dominion of that graph, which is the most reliable path for routing. The proposed algorithm recognizes those set of weakly connected nodes which are relatively more stable when compared to the other nodes hence requiring less routing and thus resulting in more number of data packets and ensuring better bandwidth utilization. Our algorithm proposes and confirms the dual betterments namely improvement in the utilization of bandwidth and reduction in the normalized routing overhead in any given network scenario.

1. Introduction

Mobile Ad Hoc networks are autonomous distributed systems that comprise a number of mobile nodes connected by wireless links forming arbitrary time-varying wireless network topologies. Mobile nodes function as both hosts and routers.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MUM 2004, October 27-29, 2004 College Park, Maryland,
USA. Copyright 2004 ACM 1-58113-981-0 /04/10... \$5.00

As hosts, they represent source and destination nodes in the network, while as routers, they represent intermediate nodes between a source and destination providing store-and-forward services to neighboring nodes. Nodes that constitute the wireless network infrastructure are free to move randomly and re-organize themselves in arbitrary fashions. Therefore the wireless topology that interconnects mobile hosts / routers can change rapidly in unpredictable ways or remain relatively static over long periods of time. These bandwidth constrained multi-hop networks typically support best effort voice and data communications where the achieved "good put" is often lower than the maximum radio transmission rate after encountering the effects such as multiple access, fading, noise, and interference. In addition to being bandwidth constrained, Mobile Ad Hoc networks are power-constrained because network nodes rely on battery power for energy. Providing suitable quality of service (QoS) support for the delivery of real-time audio, video and data in Mobile Ad Hoc networks presents a significant technical challenge.

Mobile Ad Hoc networks may be very large which makes the network control extremely difficult. The end-to-end communication abstraction between two communicating mobile hosts can be viewed as a complex "end-to-end channel" that may change routes over time. There may be a number of possible routes between two communicating hosts over which data can flow and each path may have different available capacity that may or may not meet the quality of service requirements of the desired service. Even if the selected path between a source-destination pair meets the user's needs at the session set-up time, the capacity and error characteristics observed along the path are likely to be time-varying due to the multiple dynamics that operate in the network. The fading effects resulting from host mobility cannot be always

masked by the link layer and typically result in discernible effects on the application's perceptible quality (e.g., assured delivery of audio / video may degrade rapidly). This results in topological dynamics that operate on slower time scales than channel fades and other such discontinuities. Reacting to these network capacity dynamics over the appropriate time scale requires fast, lightweight and responsive protocol operations. Flows must be established, maintained and removed in Mobile Ad Hoc networks over the course of a user-to-user session. Typically, "*connections*" (i.e., the establishment of "*state*" information at nodes along the path) need to be maintained and automatically renegotiated in response to the network topology dynamics and link quality of service changes. Since resources are scarce in these networks, any protocol signaling overhead needed to maintain connections limits the utilization of the network. Therefore, bandwidth required to support signaling systems must be kept to a minimum. This places emphasis on minimizing signaling required to establish, restore, maintain and tear-down network state associated with user sessions. In addition, due to the disconnected nature of maintaining state in Mobile Ad Hoc networks, explicit tear-down mechanisms (e.g., disconnect signaling) are impractical. This is due to the fact that it is infeasible to explicitly remove network state information (established during session set-up) in portions of the network that are out of radio contact of a signaling controller due to topology changes. There is a need for new Mobile Ad Hoc architectures, services and protocols to be developed in response to these challenges. New control systems need to be highly adaptive and responsive to changes in the available resources along the path between two communicating mobile hosts. Future protocols need to be capable of differentiating between the different service requirements of user sessions (e.g., continuous media flows, micro-flows, RPC, etc.). Packets associated with a flow traversing intermediate nodes between a source and destination may, for example, require special processing to meet end-to-end bandwidth and delay constraints. When building quality of service support into Mobile Ad Hoc networks the design of fast routing algorithms that can efficiently track network topology changes is important. Mobile Ad Hoc networking routing protocols need to work in union with efficient signaling, control and management mechanisms to achieve end-to-end service quality. These

mechanisms should consume minimal bandwidth in operation and react promptly to changes in the network-state (viewed in terms of changes in network topology) and flow-state (viewed in terms of changes in the observed end-to-end quality of service). Both the size of the network and the level of mobility of subscriber units affect the performance of a Mobile Ad Hoc network. We propose a graph based approach to abstract the network.

2. Motivation and Related Work

There are several standard techniques proposed to identify clusters in a graph. These can be broadly classified into identifier-based and connectivity-based cluster-head selection techniques. The popular techniques are the lowest-ID [17] and the maximum-connectivity [18]. In the former, if a low-ID node ever gets highly mobile, this could result in unacceptably large number of cluster head changes. In the maximum-connectivity as well as Lowest-Distance value cluster, the criterion for clustering changes from one mobile node to another depending on the activity and distance which can cause instability. In [10], the clustering is based on entropy of the mobile nodes observed over a long period of time. The entropy is a measure of how confined the movements are, and this is used to classify nodes as either conservative or exploratory.

In the proposed algorithm, a path is defined as the most reliable within the graph, through which delivery is guaranteed. In the ideal case, the path is the set of all cluster-heads identified using some suitable algorithm so that if any packet of data can be routed through one or more of these cluster-heads, they will then be delivered to the destination from the nearest cluster-head.

Researchers have proposed several techniques to identify reliable paths in Mobile Ad Hoc Networks. One desirable qualitative property of Mobile Ad Hoc routing is that it should be able to adapt to the change in traffic and network topology from time to time. Johnson and Maltz [6, 7] suggest that the conventional routing protocols are inefficient for ad-hoc networks as routing related traffic may waste a large portion of wireless bandwidth, especially if the protocols require periodic updating. Ko and Vaidya [8] suggest a Location-aware routing protocol, but this is stateless and wasteful in time as the route is discovered as and

when data has to be transmitted. Dommetry and Jain [9] briefly suggest location information in ad-hoc networks but do not elaborate on its usage. In this work, we consider a localized Mobile Ad Hoc Network where the route to destination is through a set of reliable mobile nodes identified having observed the network for suitable period of time. We thus incorporate state-information into each of the mobile nodes thereby ensuring bandwidth-enhanced routing with guaranteed delivery.

3. Dominating Set of a Graph

An efficient and reliable method of forming clusters is based on the idea of graph domination. A dominating set of a graph $G = (V, E)$ is a vertex subset S of V such that every vertex v that belongs to V is either in S or is adjacent to a vertex of S . A vertex of S is said to *dominate* itself over all adjacent vertices. A vertex of S can qualify to be a cluster head. A Connected Dominated Set (CDS) is a dominating set whose induced sub-graph is connected. In the context of Mobile Ad Hoc Networks, this implies that the connected dominating set can connect with every other mobile unit in the network and thus ensure that there is *at least* one path to each mobile unit. *Connected Dominating set ensures connectivity to all nodes in the remaining graph.* Delivery can thus be guaranteed if data can be routed from the sending unit to some mobile unit $v \in S$. Routing concentration hence shifts to routing from sending unit to any unit of S . In the end, one would wish to find a small number of elements leading to a small dominating set, in order to simplify the network structure as much as possible. Figure 1 gives an example of abstracting a mobile ad hoc network. The nodes in blue denote the mobile units and the edges denote the reachable mobile units from a given unit.

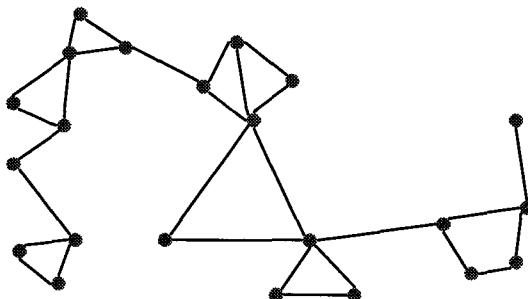


Figure 1: Network Abstraction as a Graph

In Figure 2, the network structure is indicated in blue. The red vertices in the graph represent cluster-heads and the black edges represent the virtual connections between clusters.

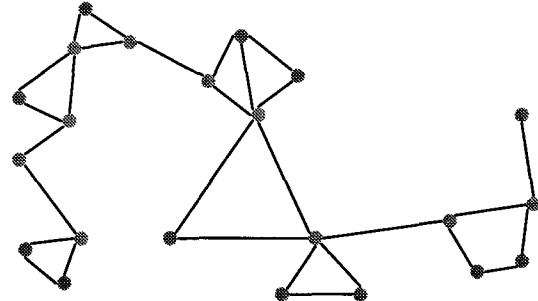


Figure 2: Dominating Set of the Graph

4. Proposed Methodology

1. Set the Area of Coverage as A metre² and the number of Nodes as N .
2. Set the Bandwidth as B and the Antenna model as Omni-directional Two-Ray ground Propagation model.
3. Configure Random positioning for nodes and mobility characteristics and prepare routing table.
4. Configure generation of data traffic between the nodes with delay-sensitive packets.
5. Monitor the traffic flow and congestion by tracing all network events.
6. Identify from trace, the set of most reliable nodes, using maximal occurrence of nodes in the routing path.
7. If there is unusual drop rate, reconfigure the network graph and routing table, goto step 3.
8. Route data traffic through the identified set of reliable nodes.
9. Compare Performance in Bandwidth Efficiency, and Routing Overhead.

Figure 3: Algorithm for Proposed Bandwidth Optimization

5. Network Model

The network model considered for verification of the proposed algorithm covers a total area of 670m x 670m and each node in the network is self-powered and can transmit to a maximum distance of 250m. The wireless channel bandwidth is 6 Mbps and all nodes in this network contend for access to a single channel. Mobility has been incorporated in all the nodes. The speed distribution is random in the range 0-10 meters per second. The MAC uses CSMA/CA for media access. AODV was identified as the routing protocol. Statistical real-time traffic has been generated to verify the

algorithm. The network is saturated, i.e. a node always has a packet to send, and designed to be a data centric network with delay-sensitive data. This assumption is supported by the fact that in future, mobile communications will be ubiquitous and will deal with real-time data more frequently. It is also assumed that each node has at least two reachable neighbors in the initial scenario. This assumption is required to ensure that no node gets completely disconnected from the network, in which case, further routing to and from that node becomes impossible. Each packet has a payload of 512 bytes. The packet arrival is modeled as a Poisson process while the packet holding time is exponentially distributed. Each node attempts to re-transmit a particular packet seven times, failing which, the packet is dropped. An individual packet may get dropped if it exceeds certain maximum number of hops before reaching the destination. Different network loads were created by varying traffic generated to and from each node and total amount of traffic generated in the network as a whole.

6. Results and Inferences

In this section, we present the results for the experiments conducted to verify the algorithm. The algorithm was verified on a grid of 20 mobile nodes moving randomly within the grid with a velocity range of 0-10 m/s. This velocity range was chosen on the basis that the proposed algorithm may be incorporated into GSM Base-stations providing both voice and data services. The Two-Ray Ground radio propagation model was implemented in the wireless scenario with an omni-directional antenna. Drop-Tail Queue was implemented at the routers which will drop packets when buffer size was exceeded. The results indicated that the algorithm significant enhancement and the path identified remains valid until the network indicates any singular nodes, by means of large burst losses. Under such circumstances, the routing table is re-configured, the algorithm is re-computed and the reliable path is re-determined. A total of 40000 packets were exchanged. The average data traffic at a transmitter was 1232 packets and subsequently after the modification it was increased to 2805, an enhancement of 227.67%, as seen from Figure 4. In Figure 5, it is seen that the averaged data received at any destination increases from 1220 to 2635 packets, again an increase by a factor greater than two.

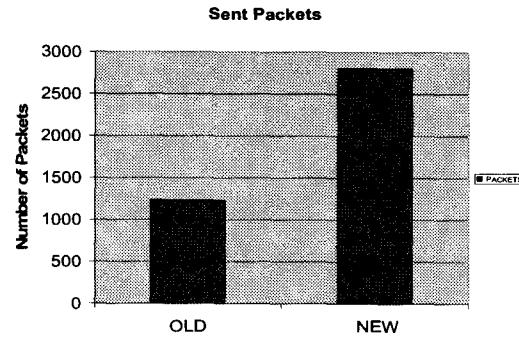


Figure 4: Graph depicting the Enhancement of Traffic at one of the senders due to Reliable Path Routing

This enhancement in data throughput can be attributed to the existence of the reliable-path which ensures that route-discovery mechanisms need not be employed which could otherwise cease all network activity in case of bursty losses to reconfigure the routing table and drop all outstanding packets at all routers. The cost associated with this enhancement in throughput is the need for additional hardware configured to enforce the routing according to the reliability of the links and the mobility of the nodes.

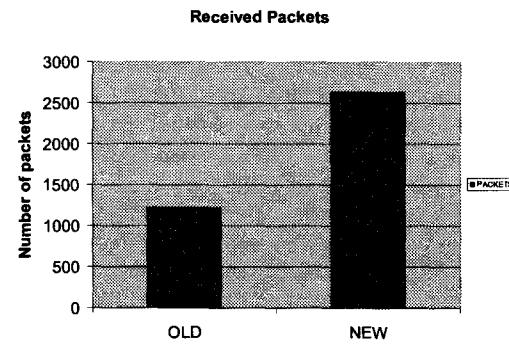


Figure 5: Graph depicting the Enhancement of Traffic at one of the receivers due to Reliable Path Routing

Another significant component is the normalized routing load, NRL, defined as the number of routing packets per data packet delivered; the results indicate an increase from 10.47 to 21.88 NRL. From the above mentioned results, the algorithm clearly shows that normalized routing load is decreased by a margin of 18.78% at a sender and 7.08% at a receiver. As normalized routing load has a substantial influence on the cost incurred by a network, this improvement will certainly help reduce the costs significantly. There is also a marginal drop in the efficiency of the network which can be attributed to the congestion incurred ascribable to the increase in traffic flow within the same time period. In conclusion, the positives of our

algorithm, the improvement in cost factor as well as the better utilization of the channel outweigh the marginal drop in the efficiency by far.

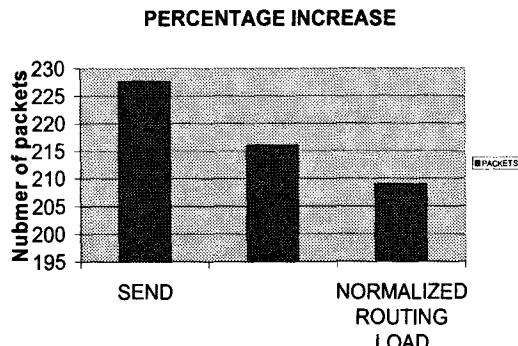


Figure 6: Graph comparing the data traffic enhancement at the two ends to the Normalized routing load

In spite of the drop, increasing the total number of packets delivered has ameliorated by more than two times.

7. Future Work and Conclusion

Number of packets delivered along with the normalized routing load is one of the most important factors in an Ad Hoc network, especially one with mobility incorporated. Normalized routing load has a significant influence on the cost incurred by a network; our algorithm positively scales down the costs. Particularly in large networks such improvement is substantial along with the number of packets delivered, the more number of packets delivered per unit time the better. From the results attained using our algorithm, it is evident that the aim of achieving the better utilization of the channel in conjunction with the reduction of routing overhead is accomplished. Future road map in this field can lead to further crystallizing improvement of the efficiency of the network along with the increase in number of packets delivered while still keeping an eye on the cost parameters.

REFERENCES:

1. S. Banerjee and S. Khuller, *A clustering scheme for hierarchical routing in Wireless Networks*, Technical Report CS-TR-4103, UMD, College Park 2001
2. Geng Chen and Ivan Stojmenovic, *Clustering and Routing in Wireless Networks*, Technical Report TR-99-05, SITE 2001
3. I. Chlamtac and A. Farago, *A new approach to the design and analysis of peer-to-peer mobile networks*, Vol. 5, Issue 3, ACM Trans. Wireless Networks, May 1999
4. Bevan Das and Vadvir Bharghavan, *Routing in ad-hoc networks using minimum connected dominating sets*, IEEE International Conference Communications ICC 1997
5. Prosenjit Bose, Pat Morin, Ivan Stojmenovic and Jorge Urrutia, *Routing with Guaranteed Delivery in Wireless Networks*, TR Carleton University, CANADA, 2002
6. D.B. Johnson and D.A. Maltz, *Dynamic Source Routing in Ad Hoc Wireless Networks*, Kluwer Academic 1996
7. D.B. Johnson and D.A. Maltz, *The Dynamic Source Routing for Mobile Ad Hoc Networks*, IETF 1998
8. Nitin H. Vaidya and Young Bae Ko, *Location-Aided Routing in Mobile Ad hoc Networks*, ACM Trans. Wireless Networks 2000
9. G. Dommetty and R. Jain, *Potential Networking Applications of GPS*, Technical Report, TR-24, Ohio State University 1996
10. Phillips Koshy and S.V. Raghavan, *Information Theoretic Approach to Clustering in Mobile Ad Hoc Networks*, NSL IITM 2003
11. Teresa W. Hayes, Stephen T. Hedetniemi and Peter J. Slater, *Fundamentals of Domination in Graphs*, ISBN # 0-8247-0033-3, 1998
12. M. Chatterjee , P. Krishna, D. K. Pradhan, and N. H. Vaidya, *A cluster-based approach for routing in Dynamic networks*, Computer Communication Review, Mobile Computing 1997

13. Prosenjit Bose, Pat Morin, Ivan Stojmenovic and Jorge Urrutia, *Routing with Guaranteed Delivery in Mobile Ad-Hoc Networks*, Kluwer Publications 2001
14. Susanta Dutta, Ivan Stojmenovic and Jie Wu, *Internal Node and Shortcut-based Routing with Guaranteed Delivery*, Wireless Networks, Kluwer Publications 2001
15. Peter Cheeseman, John Stutz and Robin Hanson, *Bayesian Classification Theory*, TR FIA-90-12-7-01, Artificial Intelligence Research Branch, NASA, 2001
16. Peter Cheeseman and John Stutz, *Bayesian Classification: Theory and Results*, Artificial Intelligence Research Branch, NASA 1996
17. M Jiang, J. Li and Y.C. Tay, *Cluster Based Routing Protocol (CBRP) Function Specifications*, IETF Draft, 1999
18. T.C. Hou and T.J. Tsai, *An Access Based Clustering Protocol for Multi hop Wireless Ad Hoc Networks*, IEEE Trans. Communication vol. 19, 2001

atMOS: Self Expression Movie Generating System for 3G Mobile Communication

Satoru Tokuhisa

5322 , Endo , Fujisawa ,
Kanagawa , 252-8520 , Japan
+81 466 49 3545

dk.sfc.keio.ac.jp

Taku Kotabe

5322 , Endo , Fujisawa ,
Kanagawa , 252-8520 , Japan
+81 466 49 3545

taku.sfc.keio.ac.jp

Masa Inakage

5322 , Endo , Fujisawa ,
Kanagawa , 252-8520 , Japan
+81 466 49 3545

inakage.sfc.keio.ac.jp

Abstract

This research focuses on movies as a new communication tool, and proposes a self expression movie generating system "atMOS". The purpose of this research is to encourage mobile movie communication. Therefore, this system adopts the concept of exchangeability, expressivity and reproducibility for mobile movie contents.

In "atMOS", based on the sub-concept of "self packaging movie", users can make an original promotion movie of themselves which can be viewed on 3G cellular phones. Users can enjoy the original movie not only by themselves but also with other people by exchanging these movies as a communication tool, using the cellular phone.

Categories and Subject Descriptors

C.3. [Special Purpose and Application-Based Systems];
J.5. [Arts and Humanities].

General Terms

Performance, Design, Experimentation, Human Factors.

Keywords

Media Design, Mobile, Movie, Cellular Phone, 3GPP, Performance, Media Communication, Sensing, Max/MSP.

"Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee."

MUM 2004, October 27-29, 2004 College Park, Maryland, USA. Copyright 2004 ACM 1-58113-981-0 /04/10.. \$5.00"

1. Introduction

Presently, the development of specs, services and contents for cellular phones are amazing.

As for the specs, sophisticated features are inevitable, and camera functions have become the standard for cellular phones [1][2][3][4]. Services are also enriched at present. For example, a debit service using digital money through the cellular phones [5], real-time route searching service using GPS and an acceleration sensor [6] is already implemented. As for the contents, automatic distribution of service of movies is in the works [7]. Some of these services can be used not only in Japan, but also in other countries. World leaders in cellular communications such as Sony and Nokia are pushing for 3G smart phones. For example, NOKIA, Finland's cellular phone manufacturer, also makes cellular phones with advanced design and function.

As stated above, technical development and deployment in services and contents for cellular phones are high-flying. From a research point of view, many papers have been presented in academic conferences like CHI, Mobile HCI, Ubicomp and Pervasive Computing, on cellular phones as a mobile information terminal. These researches can be classified broadly into 3 categories. First is the research of the interface of cellular phones. Research comparing and reviewing the menu designs [8], and research of command input method of text by 12 buttons aimed at beginners [9] can be stated as an example of this category. Second is the research of cellular phones as a information input device for a certain content service. In the MIVE system [10] a user uses a cellular phone to express their feelings and reviews of a movie on a large screen while they are watching it as one example. A video mixing system [11] which can be operated by multiple users through cellular phones is another example of this category. Third is the Context-aware content service which uses the positional information and the network. For example, Noriyuki Ueda's GIS with cellular

phone +WebGis [12] create a virtual city by collaborating the GPS positional information and images taken by cellular phones. The Re: living Map [13] is another content which uses the GPS and camera function of cellular phones, but enables to re-experience and share a place where the user has walked through.

On the other hand, there are also many researches done from the sociological point of view. "Perpetual Contact" [14] has verified theoretically and practically the comprehensive survey and social interaction of cellular phones in countries all over the world, for example USA, France, Holland, Finland and Italy, based on the reports and researches of each country's researchers.

In terms of cellular phones as a communication media, the direct communication method using the present cellular phones can be classified into voice, text, still images and video. Therefore, direct communication by voice, communication by mail, communication by sending still images that has been taken and processed by users or communication by sending motion pictures taken by the users can be raised as these methods. Real-time motion image communication by cellular phones with picture phone functions can be categorized as a motion image communication.

Before stating the characteristics of mobile movies as a communication media, the main proposal of this research, the characteristics of image communication must be mentioned. Print Club [15] is a representative of analog image communication. Print Club is a system that creates an original photo sticker, combining the frames, stamps and scribbles by the user. The main purpose of this sticker is to express oneself, and also to exchange with other people. This stands as a media communication. An enjoyment of expressing oneself as intended can be attained and the sticker has a factor that will remind the situation just as photo picture does.

This analog image communication Print Club, which holds expressivity, exchangeability and reproducibility, was extended by the appearance of cellular phones with camera functions. A service "@Sha-mail" [16] that started in 2001 by J-Phone (currently Vodafone) enabled people not only to collect images they took with their camera functions on the cellular phones but also to send these images by the mail functions of the cellular phones. The convenience of being able to send and receive images, which was taken by the users needs, attached to the mail, made the exchangeability of Print Club to an intense expansion. Functions for picture frames and paints have appeared in some cellular phones, and expressivity is advancing. NOKIA has put Medallion [17], a device

which the user can wear one's favorite image, on the market. This can receive infrared data which enables users to wear others images, and presents another aspect to image communication.

Therefore, the basis of image communication in cellular phones lies on the concept of expressivity, exchangeability and reproducibility. Likewise, contents with expressivity, exchangeability and reproducibility are adequate as a communication media for movie communication in 3G cellular phones. At the same time, an expansion of the attributes associated with contents itself must be done as the media expands from still images to motion images. As it stands now, mobile movie communication is still at an early stage of development, The reason for the lack of positive contents is considered to be from the lack of these aspects. Consequently, we developed a self expression movie generating system which promotes mobile movie communication based on the structured concept model of expressivity, exchangeability and reproducibility.

2. Concept

The concept of the self expression movie generating system, which encourages mobile movie communication, can be summarized in the following 3 points:

- exchangeability: contents premised to be exchanged.
- expressivity: contents full of expressiveness
- reproducibility: contents that revive the users' feeling

For the exchangeability, in order to exchange, there is a prerequisite that other people wants the object. This prerequisite is affected by the rarity and originality. Limited versions of trading cards and figures have a rarity value. By giving rarity and originality in the contents, the exchange of the object can be encouraged.

For the expressivity, the expressiveness of movies is important. Making use of the new attribution centered on time, and to be more expressive than that of a still image is inevitable.

For the reproducibility (see Figure 1), in this term, it can be defined as the reproduction of the users' feelings and senses by using this content. The users' feelings and senses can be defined as the reminding of the intention and the memory which is the reproduction of body sensation, a new attribution in addition to the attribution held by the image itself. In order to reproduce the user's senses, the movement of the user must be strictly sensed (systematic reproduction of body sensation), directly connected to the expression and must be an enjoyable expression, just

as the user intended (reproduction of intension). Based on this process, the system's aim is to reproduce not only the users' memory but also the body sensation attained by this content when the movie is seen.

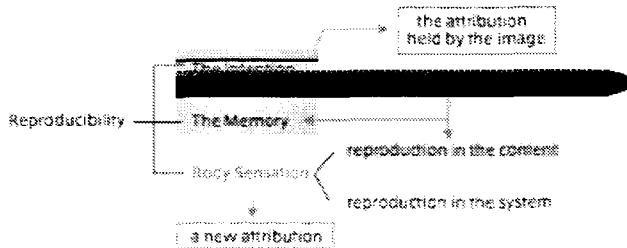


Figure 1. The Model of Reproducibility.

Naturally, these three concepts refer to each other. In order to meet the rarity and originality of exchangeability, the reproducibility and expressivity is highly concerned. The cycle can be described that by adopting the reproducibility and the expressivity in the system, the exchangeability of the contents rises.

The self expression movie generating system, which adopts these three concepts to encourage mobile movie communication, is the proposal of this research *atMOS*. *atMOS* aims to encourage the communication through a cycle of creation, acquisition and exchange, as seen in Figure 1. The main function of *atMOS* as a movie generating system is represented in Creation Self Expression in Figure 2.

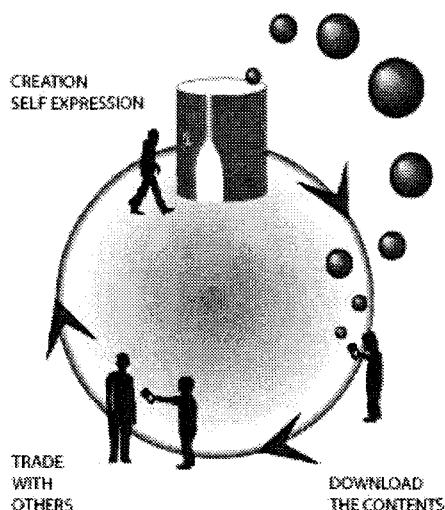


Figure 2. atMOS Cycle.

3. atMOS

"*atMOS*" (Figure 3) is a system that creates a promotion movie of the user, able to be viewed on 3G cellular phones, by simple actions and is based on a sub-concept of "self packaging movie".

This chapter describes in detail the system of *atMOS* from the expressivity, reproducibility and exchangeability point of view, together with the sub-concept "self packaging movie".

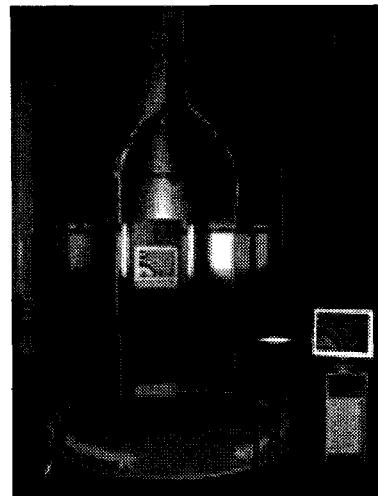


Figure 3. *atMOS*.

3.1 Self Packaging Movie

atMOS uses a sensing system to sense the motive actions of the user. Sound and image processing is done based on the digitalized data of the user's motive action. To process images, the image of the user, taken by the embedded digital video camera inside the chassis, is used as the main material. Users can download the result movie into their cellular phones. By saving the movie in their cellular phones, users can enjoy it not only by themselves but also by exchanging them with their friends. In terms of making sound and image processing based on the movement of oneself, and making oneself a material for a movie, it is packaging oneself as a content. Therefore, it is a "self packaging movie".

3.2 System Design

The system flow of *atMOS* is shown in Figure 4. It can be broadly categorized into 4 sections, as below.

- (a) Performance Section
- (b) Real-Time Effect Section
- (c) Rendering Section
- (d) Communication Section

Sensing of the users' performance is done in (a), and the data is digitalized and sent to (b). (b) allocates the data to the image and sound effects, and sends it to the monitor, speakers and (c) after processing. (c) gathers the movie data and encodes it to a format that can be viewed on cellular phones. (d) uploads the encoded movie to the server and sends to the user a mail showing the URL to download this movie.

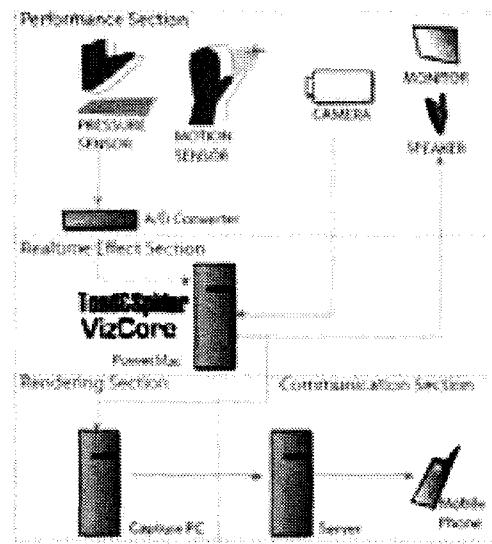


Figure 4. atMOS System Flow.

In the next place is how the three concepts, exchangeability, expressivity and reproducibility are implemented in this system of atMOS, made by this flow.

As for the development, Max/MSP and Jitter by Cycling(74) is used. The specs for the processing computers are PowerMac G4 (CPU Dual1.25GHz) when exhibiting for the first time at SIGGRAPH2003 [18], and PowerMacG5(CPU Dual2.0GHz) when exhibiting for the second time at Japan Media Art Festival [19]. These computers were used for (a) and (b), while ThinkPad31 was used for the capturing and rendering, (c) and (d).

3. 2.1 Performance Section

The performance section senses the users' performance and makes the users' action into data. Here, the objective was to realize a high reproducibility of the users' motive action. This was done by sampling the movement of the user in a condition close to free hand, in order to reduce the factor that obstructs the movement.

atMOS uses a digital video camera and LED pointers to sense the body movements of the user. The movie, taken by the digital camera inside the chassis, is processed into a reversal image and screened on the monitor outside the chassis. (Figure 5)

The result of the image processing appears real-time on the monitor. Users can see which effect is linked to which movement, so they are able to play while directly checking the movie. This enhances the reproducibility of intention.

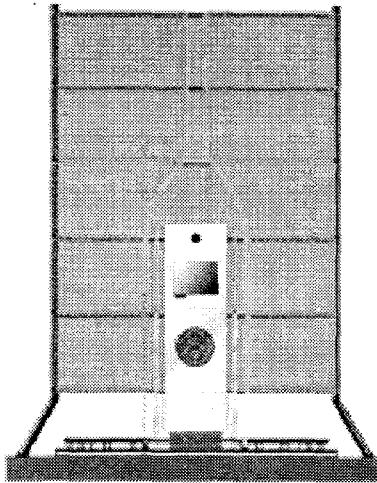


Figure 5. atMOS Design.

The position of LED pointer, which the user has, is recognized as coordinates (x,y) in XY plane (however, $-100 < x < 100$, $-100 < y < 100$) in which the central point of this monitor is set to (0,0). The quantity of the movement is detected from subtraction between the present point (x_1, y_1) of LED pointer and the point (x_0, y_0), which is the point before 1 phase (200msec) at the sampling period, in this virtual absolute coordinate. And, this quantity of the movement is reflected on sound and image processing. The sampling period is 400msec as one unit, detects the quantity of the movement with the first half 200msec, and applies the value to each parameter at the latter half 200msec. The looping of detection and application is continually repeated because the latter half overlaps the first half of the next sampling period. This enhances the reproducibility of body movements. In addition, when the LED pointer of two or more is used, an even XY is returned and the quantity of the movement is calculated with the value. Thus, not only a calculation of multiple LED pointers but also a play at a multiple number of people is implemented.

LED pointers that emits a certain amount of light, which makes the image analyzing detect an accurate data, plays a very important role in the reproducibility of body sensation At the same time, LED pointers can intensively add colors to the movie source taken by the digital camera This enhances the expressivity.

The digitalized data of the user's action is sent to the real-time effect section and is given to a parameter for the sounds and images to be processed.

There are also 8 pressure sensors (Figure 6) arranged in a circular pattern in the floor of the chassis. Each one of these sensors is allocated

to a set of image effect and sound effect. Each sensor has a different shaped sticker, so that the user can remember which place to step on when he/she wants to use the effect. This enhances not only the reproducibility of the intention but also the reproducibility of body sensation in this content.

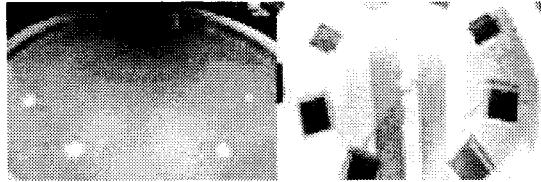


Figure 6. Pressure Sensor.

3.2.2 Real-Time Effect Section

The real-time effect section processes the sound and image using the data of the amount of movement as formed in the performance section.

In the real-time effect section, the objective was to create a system with high expressivity which flows together with the music, so that users can choreograph himself/herself radically by the users' action and by selecting the movie effects and imaginary instruments.

Figure 7 is a sample of a movie made by atMOS. In this passage, the movie effects will be described. The movie processing for atMOS is done by a movie processing sub-system "Viscore". The movie material of the user taken by the digital video camera facing the user inside the chassis is processed in real-time.

8 separate effects (see Figure 8) are prepared for each pressure sensors in the floor of the chassis. A complex effect can be processed by a combination of these 8 effects, and this enhances the expressivity. As the users are able to experience a arbitrary enjoyment, the expressivity of intention is enhanced.

In addition to this, the amount of movement obtained in the performance section is connected with the intensity of the effect, so the reproducibility of intention is enhanced by this collaboration. The intensity of the effect will rise when the user moves intensely.

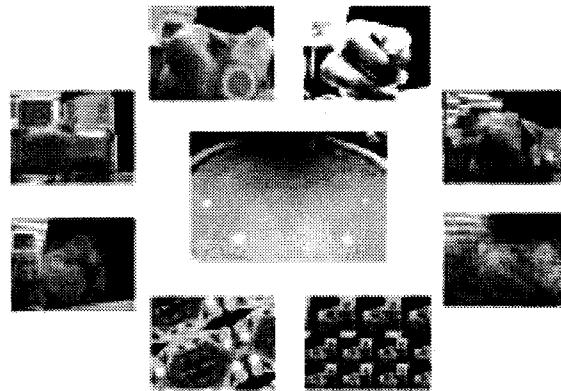


Figure 8. Effects Samples.

The 8 movie effects have a common meaning for each row to enhance the reproducibility of body sensation. By making the effects have a common meaning for each row, the user can re-experience the same situation by perceiving which sensor has the intended effect, and this can be checked on the monitor at real-time. The front row sensors are effects to change the hue of colors. The sensors in the second row are those to process multiplication effects. This can be identified clearly on the monitor while the user is playing at a short distance from the monitor. The sensors in the third row are effects with afterimages. This effect can be enjoyed by viewing the monitor from the rear of the chassis. The sensors in the fourth row are effects to change the movie by dividing it into a number of pieces. The user stands at the rear end of the chassis and will be able to enjoy the shapely change of image in the monitor.

In this passage, the processing method for sounds will be mentioned. The sound processing for atMOS is done by a sound processing sub-system "Toad&Spider". This system is an automatic music creating engine which automatically processes score creation, real-time synthesis and real-time effects. This tries to automatically complete all the process needed to compose trance/techno music. Toad&Spider is starts simultaneously with the initialization of atMOS. The expressivity is enhanced as a different music is created each time.



Figure 7. Vizcore Image Sample.

Considering the completeness of the musical piece, the interactivity was designed simple and to the minimum. The user controls 8 interactive imaginary instruments. These are allocated on each of the pressure sensors in the floor. As the effects are allocated on the same sensors as the movie effects, so these effects are grouped in a similar concept. This enables the high reproducibility of intention with the imaginary instruments by keeping the speed, chords at a certain level and by changing the scale of the music.

Also, atMOS has an imaginary audience system. The cheering sound is given as a effect to the users' movement. More the user intensively moves in the chassis, more the imaginary audience applauds.

3.2.3 Rendering Section

The rendering section encodes the movie data sent from the performance section to 3GPP, a standard format for 3G cellular phones. 3GPP is adopted to enhance the exchangeability, because this enables the movie to be seen not only with cellular phones but also with PC and other digital devices. Figure 9 shows an example of how the movie, created by atMOS, is played on a cellular phone.

3.2.4 Communication Section

The communication section uploads the encoded movie to the server and sends a mail with the URL for download to the users' cellular phone. Users are able to enjoy the movie by themselves and with others by transferring it by email (see Figure 10). Likewise, users can enjoy other users' movie created by atMOS, by downloading it to one's cellular phone. atMOS enhances the exchangeability by evolving the playing, download, viewing and sending in a seamless flow, just as image communication evolves seamlessly from shooting to sending.



Figure 10. Movie on 3G Cellular Phone.

3.3 Verification

This chapter verifies the effectiveness of atMOS as a self expression movie generating system and an encouragement to mobile movie communication through the data obtained by the two exhibits.

3.3.1 SIGGRAPH 2003

During the days of exhibition at SIGGRAPH, 300 people played atMOS. The ratio of men to women who played atMOS was 1:1.

As this exhibition itself was based on new technologies, people who came to this exhibition did not seem to feel reluctant to move around in the chassis. Without explaining the details for atMOS, people who actively moved their bodies while playing noticed the movie to have a strong difference. This proves that the change of effects encourages the user in a sense of reproducibility of intention.

A large number of sample data was obtained from this exhibition. There was a great difference between each movies of the users. This supports that the realization of expressivity and reproducibility was achieved in each section. This realization makes clear that there was originality and rarity in each of the user's content.

As this exhibition was in USA, the movies could not be sent to the user's cellular phone by network. Instead, the data was copied to a 3G cellular phone, which the users viewed, through a memory card. Despite this fact, the movie was very popular and many users asked for the movie to be sent to their PC. Also, some people who had already played atMOS appeared again and brought their friends to the booth. As the system was assumed to be played by a number of people at the same time, by being able to process multiple LED pointers movement, there were many cases where users played this with their friends. This verifies the effectiveness of encouraging direct communication.

Figure 11 is an example of the movie with especially high originality. Owing that people did not feel reluctant to dancing, there were many movies that had highly original effects synchronized to user's original movement.



Figure 9. Communication Image.

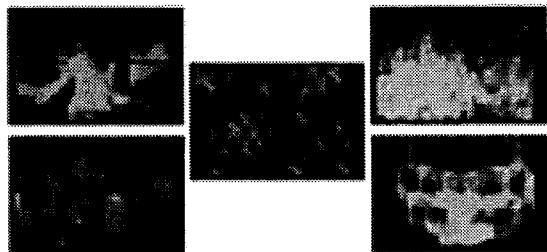


Figure 11. Self Packaging Movie Sample.

3.3.2 Japan Media Art Festival2004

In this exhibition, the complete system was presented as the system was fully equipped with network and cellular phones. Given this factor, we did a questionnaire survey to verify the effectiveness of mobile communication media. Figure 12 is a graph from the result of the survey. The users' intent of whether the user wants the movie, if the user wants to show the movie to others, and if the user wants to see other people's movie. These questions were answered in 4 levels (Absolutely Yes, Yes, No, Absolutely No) and 132 people of ages 13 to 40 answered this question.

72% of the users answered that they want their movies. 56% of the users answered that they want to show this movie to others. And 79% of the users answered that they want to see other peoples movies. As a result, a high number of users wants their own or other peoples' movies but only about half the users wants to show it to others. Based on this result, the effectiveness of this system as a mobile movie communication can be verified that though only half of the users wish to exchange their movies, they feel a satisfaction to the mobile movie created by this system, and if the movie resulted as a cool movie the users feel that they would exchange it with others. Based on this standpoint, the aim of this research was accomplished by the product which is created by this system.

On the other hand, the attribution of the source of this questionnaire survey must be taken into account. Although this exhibition was an art festival, and most people who came there were interested in digital contents, there was a different aspect from the exhibition in USA. That is to say, there were only a few people who would play this system actively as seen at the exhibition in USA. Due to the difference in national character, there is a need to verify the effectiveness by researching in other countries of Asia, Europe and USA as a comparative study.

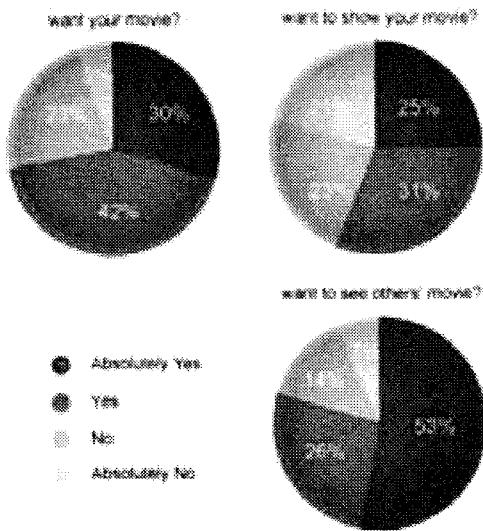


Figure 12. Graph for User's Intent.

3.4 Observation

Due to the attainment of reproducibility, expressivity in each section, atMOS enables to make a mobile movie full of originality. The rarity and originality of exchangeability was realized due to the high expressivity based on the reproduction of the users' experience on atMOS.

The system, on the other hand, still needs improvement. In order to enhance the exchangeability, an enhancement of reproducibility and expressivity, which forms the originality and rarity, is necessary. For example, in the present system, the reproduction of motive action is remained only to reproduce the motion of the hand holding the LED pointer, but there is a need to approach from what the movement implies. For instance, the movie undulate a wave pattern when a user moves one's arm in a wave like motion.

Also the exchange should not be concluded by an analog method of mail, and there is a need to set a place of exchange outside the system. For example, a device that enables to obtain other peoples' movies by accessing to a certain community site can be considered. It is essential that there is a need to structure a new method of communication that involves many people, which was not possible when the tool for communication was by mail that was sent mainly to acquaintances.

4. Conclusion

This research focused on movies as a new communication tool, and proposed a self expression movie generating system "atMOS". The purpose of this research was to encourage mobile movie communication and adopted the concept of exchangeability,

expressivity and reproducibility for mobile movie contents in the system. This successfully demonstrated the potency of mobile movie communication.

5. Reference

- [1] Casio Japan. <http://www.casio.co.jp/k-tai/a5406ca/detail1.html> [referenced 10 July 2004]
- [2] DoCoMo net.
http://www.nttdocomo.co.jp/english/p_s/products/index.html [referenced 10 July 2004]
- [3] DoCoMo FOMA VideoPhone.
http://foma.nttdocomo.co.jp/english/tv/list/tv_tv_tv.html [referenced 10 July 2004]
- [4] Vodafone V601N.
<http://www.vodafone.jp/english/products/kisyu/v601n/index.html> [referenced 10 July 2004]
- [5] Mobile Wallet Service Useable with New i-mode Smart-Card Handsets.
[http://www.nttdocomo.com/presscenter/pressreleases/press/press_release.html?param\[no\]=457](http://www.nttdocomo.com/presscenter/pressreleases/press/press_release.html?param[no]=457) [referenced 10 July 2004]
- [6] EZ Navi Walk, a full-scale navigation service for pedestrians.
http://www.kddi.com/english/corporate/news_release/2003/1006/attachment.html [referenced 10 July 2004]
- [7] KDDI EZmovie
http://www.au.kddi.com/ezweb/au_dakara/ez_movie/index.html [referenced 10 July 2004]
- [8] Robert St. Amant, Thomas E. Horton, and Frank E. Ritter. 2004. Model-Based Evaluation of Cell Phone Menu Interaction. *Proceedings of the 2004 conference on Human factors in computing systems*, 343 – 350.
- [9] Andriy Pavlovych, and Wolfgang Stuerzlinger. 2004. Model for non-Expert Text Entry Speed on 12-Button, *Proceedings of the 2004 conference on Human factors in computing systems*, 351-358.
- [10] Takeshi Kubo. 2003. "MIVE": A Proposal of LIVE Cinema Watching System using the Internet. *Keio University*.
- [11] Haruhiko Katayose, Tsuyoshi Miyamichi, and Nakamura Mitsuda. 2003. Collaborative Visual Jockey using Mobile Phone. *HCI International 2003*.
- [12] Ueda, N., Nakanishi, Y., Manabe, R., Motoe, M., and Matsukawa, S. 2003. GIS with cellular phone + WebGIS - Construction of WebGIS using the GPS camera cellular phone. *The Institute of Electronics, Information and Communication Engineers*.
- [13] Yoshimasa Niwa, Takafumi Iwai, Yuichiro Haraguchi, and Masa Inakage. 2004. The Re: living Map - an effective experience with GPS tracking and photographs. *Pervasive2004*.
- [14] James E. Katz, and Mark A. Aakhus. 2002. Perpetual Contact: Mobile Communication, Private Talk, Public Performance. *Cambridge Univ Pr*.
- [15] ATLUS Amusement. <http://www.atlus.co.jp/am/english.html> [referenced 10 July 2004]
- [16] Vodafone @SHA-mail.
http://www.vodafone.jp/english/live/shamail_s/index.html [referenced 10 July 2004]
- [17] NOKIA Medallion.http://www.nokia.com/nokia/0_43613.00.html [referenced 10 July 2004]
- [18] Siggraph2003 Emerging Technologies.
<http://www.siggraph.org/s2003/conference/etech/atmos.html> [referenced 10 July 2004]
- [19] JapanMediaArtFestival2004.
http://plaza.bunka.go.jp/english/festival/sakuhin_backnumber/15/atmoshtml [referenced 10 July 2004]

The Challenges of Wireless and Mobile Technologies

-The RFID Encourages the Mobile Phone Development -

Taro Hori

Graduate School of Waseda University

Global Information and Telecommunication Studies
1011 Okuboyama Nishi-Tomida Honjo-Shi

Saitama Japan
+81-495-24-6098

horitaro@asagi.waseda.jp

Mitsugi Matsumoto

Graduate School of Waseda University

Global Information and Telecommunication Studies
1011 Okuboyama Nishi-Tomida Honjo-Shi

Saitama Japan
+81-495-24-6098

mmatsumoto@waseda.jp

ABSTRACT

The progress of ICT has brought us numerous benefits, and the concept of ICT has continuously been extended from "network" to "global network," then to "ubiquitous network." The infrastructure of broadband environment such as ADSL or FTTH in Japan has been effectively developed. Likewise, mobile phones have become extremely popular and widely used. Ninety percent of all the mobile phones in Japan are browser phones, and they are contributing to forming a high-mobility society. In this paper, we would like to present possibilities of mobile terminals, especially mobile phones utilizing the RFID (Radio Frequency Identification) technology. The RFID technology has increased the capacities in memory, widened reading ranges, and accelerated processing speeds. Based on the premise of the RFID technology, we would like to discuss the challenges in operating wireless devices from the customers' perspectives.

Categories and Subject Descriptors

D.4.4 [Communication Management]: Network communication
- Control of the mobile terminal which used RFID.

General Terms

Performance

Keywords

RFID, Mobile Network, Cellar Phone, Terminal,

1. INTRODUCTION

A Ubiquitous Society realized by the development of information and communications technologies (ICT) can also be called a knowledge-based society. Knowledge is no longer equivalent to establishing policies and building machines as it used to be in the

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MUM2004, October 27-29, 2004, University of Maryland College Park, Maryland, USA.

Copyright 2004 ACM 1-58113-000-0/00/0004...\$5.00.

traditional industrial society of the 20th Century that solely pursued efficiency. Rather, today's knowledge is considered ecological and organic in a way that allows enough flexibility to swiftly sense environmental shifts. Furthermore, the main characteristic of knowledge in the networked society of the 21st Century is that it is connected through networks for the purpose of achieving higher goals based on innovations and new creations.

Developments in mobile terminals such as mobile phones have significantly stimulated the building of a ubiquitous networked society. The Radio Frequency Identification (RFID) technology has drawn much attention for not only its efficiency in inventory and production management, but also its significant potential for improving our daily lives. What is required is to widely and rapidly diffuse the RFID technology by finding better ways to embed it into any kinds of devices as default, and build platforms so that this technology shall become freely available and utilized. The Internet has been accessible almost exclusively through computers. However, in a ubiquitous society, anyone can connect to the internet through a variety of devices such as televisions and mobile phones. This paper examines and discusses the current capabilities and shortcomings of mobile phones and RFID built into mobile terminals which create a network of not only people but also of all the objects. In addition, we identify and design three mobile system models for mobile terminals. The purpose of this paper is to propose a new business model to facilitate consumers' convenience by combining mobile phones and the RFID technology.

2. The prosperity and problems of the mobile phone market

The number of mobile phone users in Japan has reached 82.33 million (May 2004) and the diffusion rate of the mobile phone is over 60%. There are two main factors for this rapid spread of mobile phones. One factor is the introduction of the terminal buyback system in 1999 which has substantially reduced the initial cost. The other factor is the introduction of NTT DoCoMo's "i-mode" in 1999, which has astonishingly accelerated the diffusion. Today, browser phones such as i-mode account for nearly 90% of all mobile phones. Furthermore, along with the development of broadband and the introduction of the 3rd

Generation (3G) technology, mobile phones have dramatically increased their convenience. However, while mobile phones have become personal media for communication, social problems related to the use of mobile phones have emerged. For instance, discourteous use of mobile phones in cars, trains, and buses are often argued as social problems as they affect airplane's control systems and pacemakers that can threaten people's lives. Although much effort has been made to raise awareness of courteous use of mobile phones in public transportation facilities, it has not yielded enough positive outcomes.

In this paper, the potentials of the RFID technology are discussed, which we believe is the essential technology for realizing a ubiquitous society of the 21st Century. We also believe that the RFID technology can contribute to solving these social problems. We would like to suggest that by building RFID into mobile phones, people other than the owner of the phone may gain the authority to control the use of the phone within the specific location.

3. Hybrid RFID/Mobile phone and Technological Limitations

By looking at the recent wireless devices and technologies in Japan such as Infrared Ray, Bluetooth, QR code/reader, Wireless LAN, they all seem to possess a variety of communication and transfer functions. It is essential to utilize these interfaces more effectively into mobile phones in order to produce and introduce more advanced ICT services. As a part of this attempt, RFID has been built into mobile phones. RFID is an ID system employing radio waves or electromagnetic waves, and is often called "Auto-ID system," "IC tag," or simply "tags." The RFID system has three components: an antenna, a transceiver and decoder, and a transponder (RF tag) which is built into the object stated above. It is possible to add data to the wide range of materials like plastics, clothes, skins, papers and so on of various shapes and sizes. The main characteristic of utilizing RFID is that we can assign a unique ID to each object in the real world and conduct a certain operation according to the object. In order to build RFID into mobile phones, it is required to set the standard on its size and form it into chip shape.

The RFID system packaged with the transceiver and decoder emits radio waves or electromagnetic waves. It detects the reader's activation signal when passing through the wave's emission zone. Recent handheld terminals contain the RFID tags, which enable a device to read data stored in the chips at a distance without any physical contact. The RDIF technology being built into or mounted onto handheld terminals such as personal computers or mobile phones continues to develop. For instance, NTT DoCoMo is starting a new service employing the RFID technology called FeliCa into mobile phones beginning in July of 2004. FeliCa is a noncontact IC card developed by SONY. With

this service, users can use their mobile phones as IC cards interchangeably.

Mobile phones employing FeliCa provide various benefits with its large capacity, and satisfy multi- purposes such as electronic money for shopping, entry and/or exit pass at the ticket wicket, employee ID, PC login, and any online charges. Furthermore, FeliCa technology is expected to bring a dramatic shift to retail industry as it has already attracted major enterprises. KDDI and VODAFONE have shown interests in utilizing the technology into their businesses. While the size of data that FeliCa can transfer is increasing, there are limitations in its applications. That is, FeliCa's service is application-based and it is being built into limited terminals, therefore, it is not universal across different applications with most RFID tags. In addition, the circuit of terminals and IC chips are built independently into the object such as handheld PCs and mobile phones. Thus, it is impossible to design an interactive transmission of radio waves between RFID and CPU terminal. This indicates that although mobile phones are able to capture information embedded into the barcode or QR code, it is still impossible to add data onto the communication media themselves.

4. Possibility of Mode Change Technology of Mobile Phones utilizing RFID

The most significant advantage of utilizing the hybrid RFID/mobile phones is that they can function regardless of the specification of the object. As stated previously, tags can be read through various methods such as traditional bar codes or other reading technologies. Therefore, not only systems such as physical distribution tracking or personal certification, it is also possible to build systems that are truer to the real world with higher interactivity.

By combining the hybrid RFID technology and mobile phone technology, we may be able to build a highly sophisticated model that allows two way communications between the CPU in mobile phones and the hybrid RFID, which would maximize the benefits of business and private sectors. This system does not depend on the applications to use the RFID inside the mobile phones. Instead, this model connects the CPU to RFID, and controls RFID based on the information the CPU has, or controls the CPU based on the information that RFID and the reader/writer exchange. This model is expected to solve the disadvantage of the FeliCa technology because it is only utilizable under limited condition. Furthermore, although it is still impossible to control and operate the CPU in the terminals with short-distance communication systems such as Bluetooth and Infrared, this model can be seen as a system that dynamically incorporates the benefits of both RFID and mobile phones.

Taking into consideration the mass diffusion of broadband use in Japan with the world's lowest fee, the RFID model with TCP/IP protocol is quite appropriate and ideal for mobile phones. In

addition, by utilizing the internet into this RFID model, mobile phones will gain further applicability and universality. In order to describe the efficiency of the hybrid RFID/mobile phone, we would like to introduce three models of the RFID-enabled mobile phone:

Case 1) The Terminal Control Model: adopting RFID at the gate of hospitals and airplanes

Case 2) The Missed Call Alerts

Case 3) The Mobile RFID and Extensive Network

Case 1) The Terminal Control Model: adopting RFID at the gate of hospitals and airplanes

As there is a possibility of mobile phones exerting an adverse effect inside of hospitals and airplanes, turning off one's mobile phone has become a societal norm. However, whether being conscience or not, one can forget about turning off the phone. In Japan, incidents caused by mobile phones such as pacemaker failure, or aircraft control system failure have not yet occurred. The possibilities of these incidents, however, can not be entirely denied.

In order to prevent such incidents, we would like to introduce a new method that automatically and forcibly turns off mobile phones using RFID at the gate of hospitals and airplanes. Applying the same technology, we may also be able to build a new manner mode system (built-in silent application). This manner mode system automatically switches the mobile phones into the silent mode according to the location such as inside public transportation facilities or lecture halls.

Figure 1 shows the mechanism of this model. The RFID reader/writer built into the gate sends a signal to turn off the power of the mobile phone or switch into the manner mode to the RFID installed in the terminal. When the RFID in the terminal receives this signal, it controls the CPU and executes the command. Under the present technological conditions, however, the RFID terminal and CPU function independently, thus it is impossible to realize this interactivity. We can also envision more sophisticated uses of this model. By developing RFID that identifies the moving direction of the terminals, an appropriate signal can be sent out to "turn off" the mobile phones that are entering the hospital area and "turn on" the phones leaving the hospital area. Furthermore, it is also possible to develop new modes such as one that receives but does not send out radio waves, or one that is incapable of sending/receiving radio waves while the power is on and stays in the "no service" mode.

Case 2) The Missed Call Alerts

Figure 2 shows a collaborative system between the long distance communication system and the short range communication

system. The long distance communication system only works between the mobile base station and the terminal, and is used mainly for voice transmission and Internet. The short range communication system is a communication standard that reads infrared ray and QR code, and works only within a short distance such as between the terminal and the object. This system also functions when the RFID built into the terminal communicates with the reader/writer terminal.

The data received with the short range communication system are transmitted to the main computer which controls the reader/writer function. Through LAN connection, the data are utilized and the feedback is provided during the next short range communication.

When the power is off, mobile phones cannot recognize incoming calls because the mobile base station cannot receive the radio wave. However, by adopting this system, missed call alerts can be sent out even when the power is off. When the terminal passes the exit of the hospital, the system sends out the missed call alerts as the computer controlling the reader/writer at the gate of the hospital transmits the acquired terminal ID to the mobile base station. Introducing such system would be beneficial to users as it reduces the disadvantages of turning off the mobile phones.

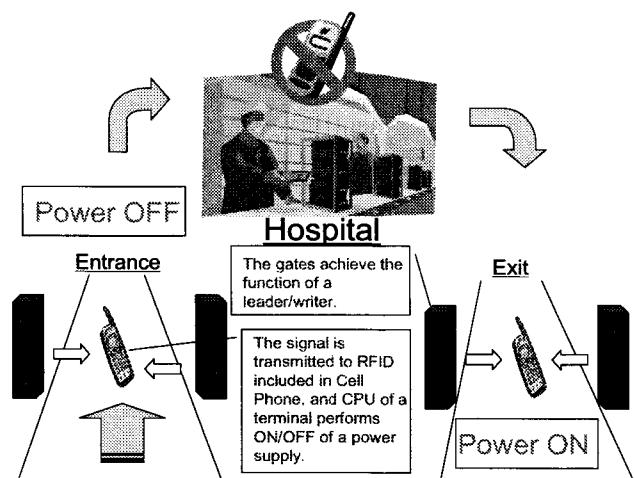


Fig1. At the gate of the hospital: The terminal control model with built-in RFID

Although as a similar service, voice mail system has become affordable, it requires registration and is not free of charge. Therefore, the power and mode controlling system we introduce would be significantly functional and add further value to mobile phones.

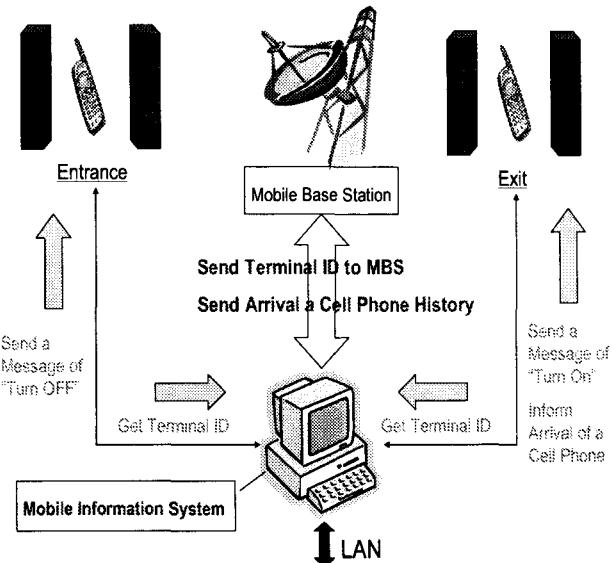


Fig.2 The Missed Call Alert System

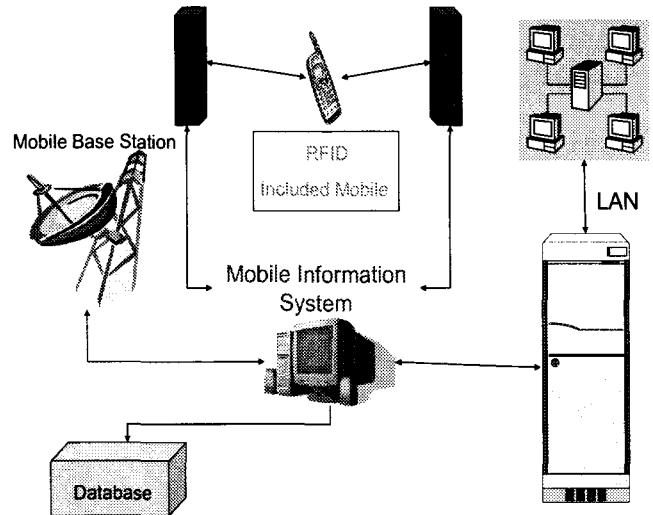


Fig.3 Mobile RFID and Extensive Network

Case 3) The Mobile RFID and Extensive Network

Figure 3 illustrates a model utilizing a database. In this system, the ability of retrieving mobile terminals is increased by storing the data acquired by the RFID reader/writer in the database. Incorporating a database would not only reduce the load on the main computers. As we expect that RFID will be built into a variety of objects including mobile phones in the near future, incorporating a database is plausible and it would significantly enhance the performance. For instance, in case an object is being lost, it can easily be detected and retrieved with mobile phones by making an inquiry in the database with the assigned ID in RFID. Furthermore, as the network expands with utilizing databases, we believe mounting RFID onto mobile phones will become universal. This also suggests the potentials of mounting a CPU onto mobile phones in gaining control over the RFID send/receive system.

5. Conclusion and Discussion

In this paper, we described how RFID would broaden its functionality in the next generation ICT environment from sending/receiving information and allocating unique IDs to controlling of individual objects and incorporating network. However, there are several issues to be solved: 1) limitation of carrier frequency bands for RFID systems, 2) radio wave collision with mobile phones in the limited carrier frequency bands, and 3) standardization of IC chips built into mobile phones.

The RFID tags are either passive or active. Passive tags operate without an internal battery and power is applied through the reader/writer. Active tags, on the other hand, are powered by an internal battery.

Besides the above stated characteristics, tags are classified according to their transmission type: electromagnetic coupling type, electromagnetic induction type, a microwave type, and optical type. Each type is different in terms of the size of its transmission area. Although it is possible to adopt the anti-collision function which enables the reader/writer to read more than two tags simultaneously, it may also place limitations on the shape of the tags.

For a future study, we would like to conduct a demonstration experiment to find out which RFID standard is most suitable for the models we proposed in this paper. We also would like to investigate the necessary software for intercommunication between RFID and CPU, and scrutinize how to incorporate existing communication systems. FeliCa could establish a security system that prevents those without authority from accessing the services by layering applications and providing a firewall to each

layer's application. Establishing a perfect security system is imperative and essential for the further diffusion of RFID

While we mentioned the possibility of mobile phones' electric waves exerting an adverse effect, RFID may pose similar threats because of its use of electric waves and electromagnetic waves. Regarding this issue, the Japan Automatic Identification Systems Association states that, "theoretically, it is safe for antennas to be as close as 10 cm to the pacemakers." Nevertheless, it also states that, "there are approximately two hundred kinds of pacemakers and there hasn't been any comprehensive research conducted on their safety with RFID, hence further investigation on this issue is required." Therefore, diffusing RFID inevitably requires collaboration with pacemaker manufacturers.

As challenges for the future, we would like to further investigate solutions for technical problems concerning data transmission which was conventionally utilized for long distance communication. By applying RFID to mobile devices, data transmission may also become available for shorter distances, and

we believe it will ultimately bring mobile communications much closer to customers.

6. REFERENCES

- [1] Klaus Finkenzeller, *RFID HANDBOOk Secound Edition*, Wiley, 2003
- [2] Japan Automatic Recognition System Association, *Intelligible RFID* , Ohmsha ,2003
- [3] Junichi Sakata, *The Research of Spreadized Policy of RFID on SCM*, Matster thesis of GITS Waseda University, 2001
- [4] Jun Hosoya, *Proposal of Anti Collision Procedure at the Time of RFID Data Transmission Which Made 13.56MHz Belts Subcarrier*, Master thesis of GITS Waseda University, 2001
- [5] Mitsuji Matsumoto, *Mobile Computing Text Book*, Askey, 1999

Context-Aware Middleware for Mobile Multimedia Applications

Oleg Davidyuk, Jukka Riekki, Ville-Mikko Rautio and Junzhao Sun

Department of Electrical and Information Engineering

P.O. BOX 4500, University of Oulu

Finland, 90014

358 8 553 2544

{oleg.davyuk, jukka.riekki, ville-mikko.rautio, junzhao.sun}@ee.oulu.fi

ABSTRACT

We present a context-aware middleware for mobile multimedia applications. The middleware offers functionality for service discovery, asynchronous messaging, publish/subscribe event management, storing and management of context information, building the user interface, and handling the local and network resources. It supports a wide range of context information including location, time, and user's preferences. Further, it allows controlling the connectivity of the device; the middleware is capable of switching traffic from one network connection to another. It can locate the services and software components as well. It facilitates development of multimedia applications by handling such functions as capture and rendering, storing, retrieving and adapting of media content to various mobile devices. The middleware offers facilities for media alerts, which are multimedia messages that are generated when predefined context is recognized. With these capabilities, it enables development of complex context-aware multimedia applications for mobile devices.

Keywords

Multimedia content adaptation, context-aware and mobile computing

1. INTRODUCTION

Personal digital assistants (PDAs) and mobile phones nowadays offer a versatile set of facilities, which enables developing various new applications. Context-aware mobile multimedia is an application area that has been attracting the attention of researchers in recent years. This application area offers a big business potential, as the users of mobile devices want to access multimedia content and context-aware services from their devices.

However, mobile devices have limited bandwidth, small screen size and low computational power. Hence, the multimedia content has to be adapted to fit the constraints of the device.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MUM 2004, October 27-29, 2004, College Park, Maryland, USA.
Copyright 2004 ACM 1-58113-981-0/04/10...\$5.00.

Mobility and context-awareness introduce further challenges. The applications have to adapt themselves to a changing environment, i.e. the recognized context.

An intermediate software layer, which performs the tasks related to mobility and context-awareness and provides multimedia support could solve the problems raised. It helps to avoid the increasing complexity of the applications and lets the developers concentrate on the application-specific tasks.

Several middleware solutions have been developed in the area of distributed computing, such as CORBA [1] and J2EE [2]. However, these technologies are intended to be used in networks with fixed infrastructure and require the devices to have a lot of available resources. Hence, they are not suitable for mobile computing. Furthermore, they have no support for multimedia.

The Service-oriented Context-Aware Middleware (SCAM) project [3] provides an architecture for context-aware mobile services. It concentrates on context-related facilities and has few services to support mobility of the applications. It does not offer functionality for multimedia applications, though. The CAMPUS project [4] presents a middleware for context-aware applications. Its novelty is in the context model, which is based on the fuzzy sets theory. However, the middleware is suitable only for fixed devices, because it has no support for mobile and multimedia applications.

The CORTEX architecture [5] is a middleware for context-aware distributed applications. It has functionality for messaging, service discovery and resource management. The system is not capable of controlling the network resources and provides no adaptable UI for applications. Further, it does not support multimedia applications. The DISCWORLD system [6] concentrates on mobile robot systems. Its middleware includes several components providing functionality for communication. The aim of the DISCWORLD middleware is to solve the network challenges of connecting a set of robots and controlling PDAs into one network, hence the middleware focuses on messaging facilities only. Furthermore, it has no support for multimedia applications.

The middleware service for distributed multimedia applications, offered by Lohse et al. [7] provides functionality for rendering audio and video in distributed environment. It also supports adaptation of multimedia content for restricted resources of mobile devices. However, the middleware solution does not offer functionality for context-aware applications.

The QoS DREAM framework [8] is focused on supporting the development of context-aware multimedia applications. It has event messaging component, data storage and distributed multimedia service. The framework supports streaming media, but

the context is presented by location information only. The framework does not support mobility of the applications (e.g. the applications cannot migrate from one host to another). Besides, it assumes that the network connection is fixed. The mobile devices are used as location sensors only.

Lau and Lum [9] developed a decision engine for multimedia content adaptation. They offer a real-time processing system for rich multimedia elements, which allows accessing the content using PDAs. Lemlouma and Layaida [10] offer a framework for adaptation of content delivery to mobile devices. However, these solutions do not offer functionality for the development of mobile context-aware applications, even though the projects are focused on the delivery of multimedia to mobile devices with constrained resources.

From this short survey, it can be noticed that there is a gap between context-aware solutions for mobile applications and solutions for distributed multimedia applications. The first ones handle contextual information and support mobile applications by hiding heterogeneity of the hardware. They offer solutions that are light enough to be executed in a mobile environment where the resources are limited. The solutions for distributed multimedia applications assume that mobile devices have restricted resources and, hence, offer functionality for multimedia content adaptation. Besides, they support mobile applications by providing network transparency, making them able to operate in a distributed environment. These middleware solutions are not able to perform context-related tasks, however. The only framework that supports context-aware, distributed and multimedia applications is QoS DREAM. However, it has a number of constraints, which were described above.

Furthermore, neither context-aware middleware solutions for mobile applications nor solutions for distributed multimedia applications are capable of controlling network resources at the middleware level.

In this work, we present a middleware solution for development of context-aware and mobile multimedia applications. The middleware is able to perform most of the tasks dealing with context-awareness and mobility, and it focuses on multimedia support. It locates the services and components, provides asynchronous messaging between the applications, stores and processes the context data collected from various sources, manages local and network resources, etc. The support for multimedia includes functionality for capturing, rendering, processing, adaptation and storing the media content. The middleware introduces a new type of events, called media alerts, and facilitates building media messaging applications. Media alerts are multimedia messages that are generated when a predefined context is recognized.

This paper is organized as follows. Section two specifies our context model and context-aware middleware. We describe the functionality and architecture of the CAPNET middleware in the third section. The implementation and the prototype are presented in the fourth section. The fifth section contains a discussion and summary of the work.

2. ARCHITECTURE REQUIREMENTS

Context awareness of ubiquitous mobile applications has recently been attracting the attention of many researchers as an interesting and perspective topic for research. Context awareness is an essential feature of mobile systems, because almost all the

ubiquitous applications utilize context information in their operation. We adopt Dey's definition of context as "any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and application themselves" [11]. An example of context information can be a user's location, time, user's profile, local resources of the mobile device, available services, etc.

Context awareness characterizes a system to use context information when it performs its tasks. In the present paper, we use the task-oriented definition of context awareness: "a context-aware system uses context to provide relevant information and/or services to the user, where relevancy depends on the user's task." [11]. That is, the main stress is put to the context that is relevant to the task.

Context awareness involves performing data acquisition from sensors, context recognition and other tasks necessary to complete before the context can actually be used. Delegating the data acquisition and context processing tasks to applications makes them almost impossible to reuse. One solution to such a problem is to decouple the tasks from applications and move desired functionality to the lower layers. Such layers, which serve the needs of applications, usually form a special layer called middleware. We keep to the definition of middleware given by Couluoris as a "software layer that provides programming abstraction as well as masking heterogeneity of the underlying networks, hardware, operating systems and programming languages" [12]. This middleware layer hides the heterogeneity and distributed nature of devices measuring the context information. A context-aware middleware serves the context needs of applications. The functionality of such a layer is discussed in detail in the next section.

A context-aware middleware has to provide the applications with the following context-oriented functionality [13]:

- support of a variety of sensor devices,
- support of the distributed nature of context information, because the data comes from different sources,
- providing for transparent interpretation of applications and abstraction of context data,
- maintenance of context storage, and
- control of the context data flow.

Mobility introduces a number of constraints to the middleware[14]:

- the bandwidth is low and hosts can be unreachable due to network partitions or poor coverage,
- the local resources like memory capacity and CPU power of the device are very limited,
- the communication between the system components is asynchronous, and
- the execution environment is dynamic.

Dynamic execution environment refers to constantly changing distributed mobile systems. Hosts are not statically connected to the network and the configuration of such systems is hard to predict. Therefore, the middleware must support discovery of hosts and service in such a continuously changing system.

Furthermore, multimedia applications set requirements for communication and computation. Multimedia algorithms need a

lot of bandwidth and processing power. Hence, the middleware has to be capable of:

- Using the available bandwidth: the middleware has to use all the different, available connections and switch to the connection that best fulfills the requirements at the given moment;
- Controlling the place of computation: to cope with limited resources the middleware needs to decide where the computations should be performed based on the situation at hand.

Furthermore, the middleware must support various multimedia devices such as video cameras, microphones, etc. Finally, there are requirements that the middleware has to fulfill to make the system adaptable [15]:

- Triggering of adaptation on a system-wide level,
- Support for system-wide adaptation policies, and
- Providing a common interface between devices and middleware.

3. CAPNET MIDDLEWARE

CAPNET middleware has been developed to fulfill the requirements set to the architecture by context-awareness, mobility of the software components, multimedia applications and adaptation.

The CAPNET middleware is located below the application layer and on top of the layer of existing technologies. All the functionality provided by CAPNET middleware is divided into a set of components. Each component offers a particular service to other components and applications. The CAPNET middleware includes such components as connectivity management, service discovery, messaging, component management, user interface, context, media and context-based storage. The structure of the CAPNET middleware is shown in Figure 1.

CAPNET middleware fulfills all the requirements set by context-awareness. It supports different devices to collect the location data and record stream video. The data comes from distributed software components, located physically at different networked devices. Context abstraction is performed transparently through intermediate components, so applications receive already abstracted data, instead of rough measurements. The middleware offers a full set of context storage facilities, like storing any context-related information, even including files.

The mobility of the CAPNET system is supported by light component framework. The components are executed at the device that has enough resources. The connectivity management component controls the bandwidth and overall traffic. The messaging component, service discovery and component management provide the necessary level of transparency for applications making them able to operate in mobile environment.

The requirements set by multimedia applications are also fulfilled with the component framework and the basic services that it offers.

The adaptation is performed at middleware level, which facilitates easy building of multimedia applications. Every application defines its resource and network requirements, which have to be taken into account during the execution of the application. Besides, applications define network policies that constrain connections and traffic. Finally, the middleware

components have common interfaces to the layers located beneath. It allows easily extending the set of the hardware supported by CAPNET middleware.

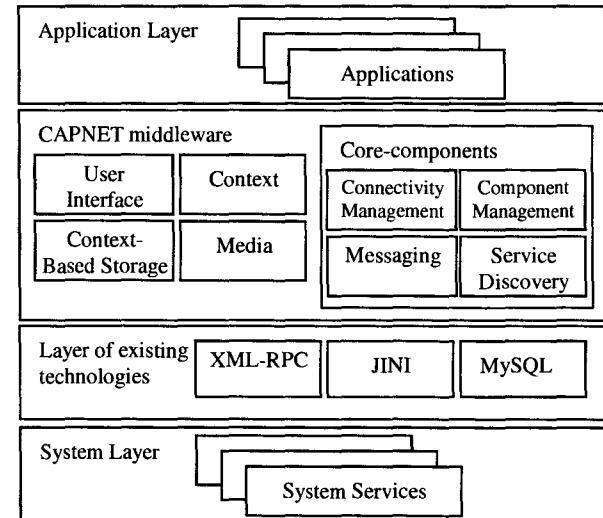


Figure 1. The architecture of CAPNET middleware.

3.1 CAPNET Components

The CAPNET middleware components are atomic software entities with explicitly defined interfaces. The internal structure and complexity of the implementation are hidden behind the component's interface. All the components are represented with proxies or stub components. This proxy approach helps to achieve the distribution transparency in the environment. Hence, the components communicate with each other in the same fashion regardless of their real location – local or remote. The components can be searched and started at system startup or on the fly, while applications are running. This facilitates the adaptation of the middleware to the environment.

The CAPNET core contains component management, messaging and service discovery components. They provide a full set of services required to support the other components and applications in a distributed environment.

Component management performs the controlling of the components and their stubs. It starts the components at startup, initializes and creates proxies, processes the requests to gain access of the components to each other. When a handle to a component is requested, the component management returns the reference to the proxy component, instead of reference to the real component. If the required component is not available at the local device, the component management loads it from the Internet from a specified location. The component management is able to make decisions and move components to the remote host if the device does not have enough local resources (e.g. memory or CPU power is limited) to execute the component. This is required when the media components need to perform processing or adaptation of the multimedia content.

The communication in CAPNET middleware is the responsibility of the messaging component. It creates channels and offers the functionality for asynchronous communication,

channel-related operations and supports remote procedure calls in the environment. A channel is a logical link between different physical applications, which are located at networked devices. The messaging component provides the location transparency together with component management.

The event-notification mechanism is based on a publish-and-subscribe model. The components and applications (event consumers) can subscribe events of a specific type from event producer components.

In the dynamic distributed environment, the availability of services and hosts changes in continuous fashion due to the network disconnections and poor coverage in some locations. Hence, service lookup becomes an important issue, because there is no centralized server and the default server does not exist. Location transparency is achieved with service discovery facilities. This component locates services and available components. Clients request service discovery to find services provided by other components. Service discovery looks up and returns the list of the matches to the client, which selects the desired component from the list and requests component management to get reference to the selected component.

The connectivity management and media components are described in more detail in the following subsections. The rest of the middleware components are optional to be running in the device, but necessary to support context-aware mobile applications. Among service-enabling components are context-based storage, context, and user interface components.

The storage facilities in CAPNET middleware are provided by context-based storage. It mainly stores and retrieves data by request. The component deals with synchronization of the data and alerts the clients when the content of the database is changed. The context-based storage is responsible for defining the privacy and access rights to the user's data.

The responsibility of the context component is providing the context information: context delivery, processing and service. The context component plays the role of a wrapper for context sensors. Furthermore, it is used as a server component, when the context data comes from distributed sources. The context component along with context-based storage offers context history maintenance facilities.

The user interface (UI) component acts as a UI front-end to a mobile device and supports the design and implementation of application UIs using three different techniques: abstract UIs (XML), plug-in UIs (downloadable Java code) and Web-based UIs (HTML). The UI software developer may choose the technique that best suits her needs, but the UI has to be separated from the application logic. For abstract UIs, the component presents a simple interface for building and modifying the UI implementation as well as exchanging messages and events between the application and the UI implementation. In case a Java plug-in UI is used, only message exchange is supported. Messages are delivered using the CAPNET messaging mechanism. Web-based UIs are launched in the Web browser of the target device and communication between the application and its UI has to be built separately using the Web techniques (Web server, CGI-scripts, Java Server Pages, etc.).

3.2 Connectivity

The connectivity management component controls and monitors the connections of the mobile device. The component hides the

details of the actual connections behind channels. Channels provide network transparency for the application layer and allow dynamic adaptation of the network facilities without interfering with ongoing communication. That is, components use channels and they do not see which connection is used to transfer the data. The connection can even be switched in the middle of communication without the components noticing it. Currently, connectivity management supports the TCP/UDP, HTTP, Multicast and TCP listening channels.

The clients can request real-time information about the status of the channel, e.g. bandwidth and availability of the connection. Besides direct requests to the component, the clients are able to subscribe this information and get notifications when the status of a channel changes (for example a connection is terminated or becomes available).

The connectivity management component controls the channels by switching the connections on the fly (e.g. from WLAN to GPRS) according to the defined policies. The channel policies can be changed through the public interface on the fly by the user or an application. We are not aware of any middleware solutions that can control connectivity at the channel level, limit the overall traffic or bandwidth and dynamically switch connections while clients are using them.

The connectivity management has three subcomponents: connection controller, policy manager and connection monitor. The architecture of the connectivity management component is shown in Figure 2.

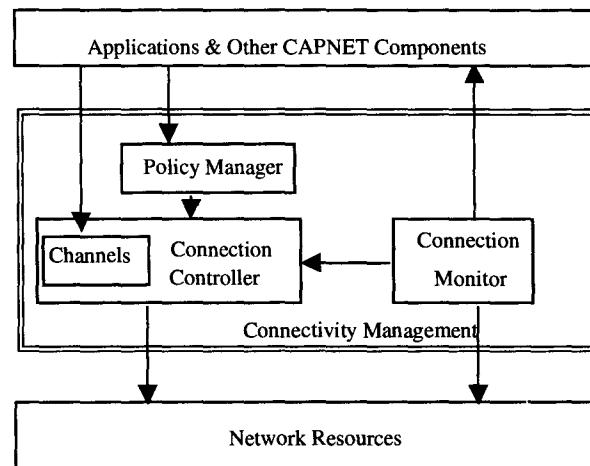


Figure 2. The architecture of the connectivity management component.

The connection monitor senses the channels' context information. It collects both QoS parameters of channels (e.g., round trip time, RTT) and network characteristics like IP addresses of the local and remote hosts. The policy manager is responsible for defining criteria and rules for channels. The policy manager allows binding the rules to the particular channel or device. The connection controller is capable of maintaining both the policy manager and connection monitor subcomponents. Besides that, the connection controller manages the channels as the client desires.

3.3 Media Components

The media components facilitate portability and scalability of native media capabilities across the various devices. The goal is to provide developers of media applications with functionality to capture images, audio and video. Media can be processed and stored either locally or remotely. Media content can be adapted for various devices. The adaptation can be performed both at a local or a remote device.

Media components are divided into core and optional components. The core media component is static and it is always present when media functionality is used. The optional, dynamic components are loaded by request, depending on the needs of applications and available resources at the device. So, if the device does not have enough resources, the dynamic media components can be started remotely, at the server side. The dynamic components can be constrained to run only at the server side. The architecture of media components is shown in Figure 3.

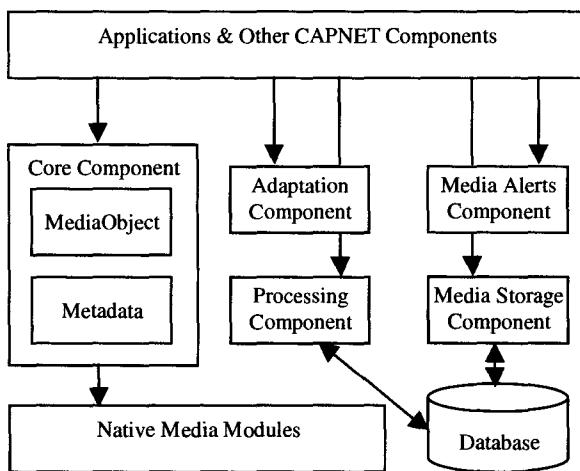


Figure 3. The architecture of media components.

The core component creates and performs serialization of media objects and handles related metadata. The media object contains the data of all media sub-elements, relations of media elements and metadata of each sub-element. The component supports three types of relations between media elements: spatial, temporal and navigational relations. Spatial relations define how the elements are spatially related to each other. The temporal relations define how the media elements are temporally related, e.g. element B has to be played before element A. The navigational relations define the links of elements to each other, e.g. element A can contain a link, which requires the media playback to play element B. The spatial, temporal and navigational relations are presented according to the SMIL 2.0 [16] standard.

The rest of the components are dynamic and they offer functionality to store and retrieve media elements and process them if necessary. The components are also capable of capturing and rendering media objects. Furthermore, a media alert component offers an interface, which supports a push-type delivery mechanism of media objects. Media storage component is an interface between the database and media components. It

uses metadata to create indexes and offers a flexible way to query the database for media objects.

Applications require adaptation of media objects to make them suitable for playback on different devices. This functionality is supported by media components as well. The clients pass the media objects and information of the terminal device to the media components. The dynamic components retrieve the facilities of the device from the database and adapt the media object according to these constraints. The adaptation is performed either locally or remotely (e.g. at server side).

Capture and rendering facilities are supported by media components to provide an easy way to create media objects and play them. Before the actual capture or rendering starts, the media components check if the target device has enough resources to play or capture media. Instead of CAPNET media components, it is possible to use external, third party components to perform capture or rendering of media objects.

Media alerts are events associated with media content. They are generated when some media-related context occurs, e.g. a camera detects movement or a microphone detects speech. Components and applications can subscribe to receiving the alerts that they are interested in from the media alert component. Besides, the media alert component delivers media alerts between other components and applications by request.

The media components support also invocation of the media alerts by applications. So every application can invoke its own kind of alerts. This allows creating a flexible infrastructure for delivering media messages. The media alerts are delivered in an asynchronous way. If the client is not available to receive a media alert, it will be queued and delivered later, when the client is available.

4. PROTOTYPE AND EXPERIMENTS

Two prototypes have been implemented to verify the CAPNET architecture. The prototypes were tested with Compaq iPAQ PDA equipped with a WLAN network card. The programming environment of the device includes Insignia Jeode Java virtual machine. The devices are located by Ekahau WLAN positioning engine. It locates the devices according to the measured strength of the WLAN signal. The prototype utilizes a MySQL database, which is running at the server side. All the communication between components is performed by using open-source Marquee XML-RPC library. The service discovery is based on Jini.

The functionality of the prototype is implemented as described below. The components can be searched, downloaded and started dynamically. The requests of the clients to get references to other components are processed by component management. It is capable to start components both at local and remote hosts. If a component is not available at the required host, the component management is able to download it from the Internet. The capabilities and requirements of the components are presented in the form of XML descriptions. Such descriptions include both optional and mandatory characteristics of the components, like names of the component's interfaces, unique ID numbers, URLs of the component's code and so on. The descriptions are used by component management and service discovery components. The messaging component is able to perform remote procedure calls between components regardless of their location – be it at the same or different hosts. The functionality implemented in the messaging component includes

sending of asynchronous messages. All the message-based communication in our prototype is based on XML-RPC protocol. We selected XML-RPC because it is a lightweight protocol and its learning curve is shorter than CORBA's IIOP or SOAP. The disadvantages of XML-RPC are that it does not allow sending objects as pass parameters of functions and it dictates to use unnamed, positioned structures in XML messages [17]. The mobility limits the network resources in our prototype; therefore we are not able to utilize the whole features offered by object-oriented XML-RPC.

We implemented our own component approach, because CORBA and J2EE are suitable only for fixed distributed solutions, not for mobile web applications. However, the CORBA implementation works faster than XML-RPC based client [18]. At the same time, both CORBA and J2EE require more time for learning and more bandwidth for normal performance.

The connectivity management component is capable of changing the connections from one type to another without interference to the work of applications. This channel adaptation is transparent and does not require any decision-making process from applications. Applications and clients just need to define the channel policies, so the connectivity management changes the connection accordingly to these policies.

The service discovery component is capable of locating services and other components using Jini and an internal database. When needed, the clients ask service discovery to look a service or component up. The service discovery provides the client with the list of matches. Then, the client selects one entry from the list and requests the component management to get a handle to the desired component. The service discovery performs location-aware discovery of the services as well. For example, the clients can constrain service discovery to request a list of services available near the location of the client. The performance of the dynamic service discovery was tested as well.

The CAPNET environment supports different kind of events. The context-related events are produced and collected in a push model. Whenever the event occurs it is forwarded to the context component, which sends it to the component responsible for that event. The events are delivered in a publish-and-subscribe fashion. The event suppliers have to register their interest in some event type from a certain component event producer. The middleware supports internal events, which are delivered directly from one component to another. The internal events are the responsibility of the messaging component.

The media component is capable of capturing the video and taking images. The synchronization of audio and video streams has not yet been completed, so the video is recorded and played separately from the sound. The media component is capable of creating media objects and handling them related to the objects' metadata. The rendering functionality is implemented, as well as media alerts and storage facilities. The adaptation of media objects is at an early stage of implementation. The other components, like context, UI and context-based storage are implemented as it was described in the previous section.

We experimented with prototypes for routine learning and business meetings. The first prototype is capable of learning user's habits and able to support location-based reminders. It consists of four applications: profile manager, reminder, calendar and personal assistant. The calendar application is used to store appointments and meetings. The context component gets events from the calendar via the reminder application. Finally, the profile

manager detects the routines from the context component. The personal assistant application is used to browse the applications at the mobile device.

According to the prototype scenario, the user buys his phone and starts using it. It learns that the user has a habit of switching the phone on the silent mode during weekly meetings. After it has detected a routine, the system offers to switch the phone on silent mode automatically if the user has a meeting. Furthermore, the prototype supports location-based reminders. So, the user attaches a reminder to a location and then the system recognizes the place and notifies the user by popping up a message.

The prototype for business meetings is based on a scenario as follows. The user has to make a presentation. He remotely checks the available services in the meeting room. After the presentation is created, he adds the presentation file to the event in his calendar application. In the morning the user gets a reminder about the meeting. When it is his turn, the user searches for available services in the meeting room, and then selects the slide projector. The list of files associated with the meeting is shown as well. The user browses the required file and starts the slideshow by using the mobile device as a remote control unit. During a meeting the user's wife tries to reach him by phone. The wife gets a notification that her husband is at the meeting. She decides to leave a video message. When the meeting ends, the user gets notification about the video message and watches the message.

The prototype for business meetings needs phonebook, video messaging, projector, and service browser and file browser applications. The video messaging application is capable of recording and showing the video messages. The projector application supports controlling the actual projector via mobile device. The service browser is the application that helps the user to request service discovery for available services. The file browser application is an interface to the database. The scenario for the media component is presented in Figure 4.

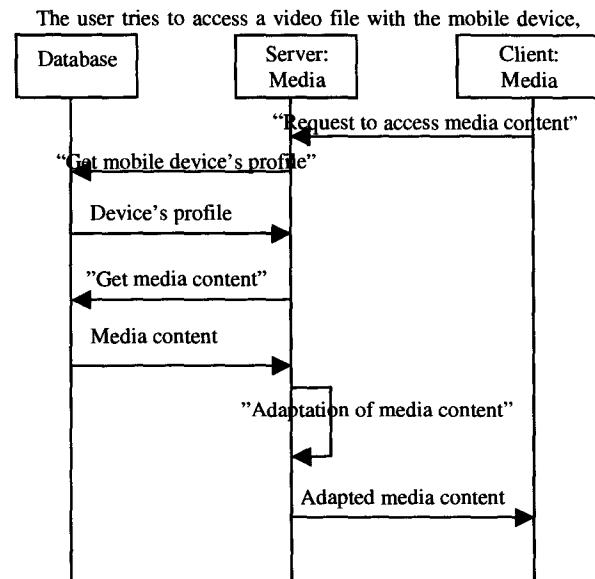


Figure 4. Media components scenario.

but it is not capable of playing the file, because of the small

screen. The media component, located at the mobile device, requests the server media component, located at the server, to get the video file. The server component, in its turn, queries the device profile from the database and then retrieves the original video. After the adaptation is completed, the server sends the file to the mobile device, where the user can play it. In the experiments, both the prototypes worked as expected. The media scenario was performed as explained. The media components processed requests to provide access to the video clip. They retrieved the device profile from the database and delivered the video clip to the mobile device. The clip was rendered at the device. The adaptation mechanism of the media content is still at the implementation stage.

5. DISCUSSIONS

We have presented context-aware middleware for mobile multimedia applications. The middleware provides key features for both context-aware multimedia applications and their distributed environment. The middleware is decomposed into a number of components, offering certain functionality for applications. The middleware is developed to work in the mobile environment. Hence, the resources of the terminal devices are very limited.

The middleware is oriented to provide functionality for context-aware applications. It has a common interface for context sensors, so it is easy to embed a new sensor to the system. The sensors can reside on physically distributed devices. Hence, the context information is forwarded to a centralized context component that performs further context delivery and processing. Besides, the context component is responsible for context recognition. These context-related operations are performed at middleware layer, which makes context processing and recognition transparent for applications. The context-based storage is the component that provides data storing facilities. It is responsible for data synchronization as well. If the content of the database was changed, the component notifies the clients. They can subscribe their interest to receive such alerts from the context-based component. Furthermore, it together with the context component controls the context data flows in the environment.

The middleware supports component mobility; it can decide whether to start a component on the client side (mobile device) or server side (fixed PC) depending on the resource requirements of the starting application or component. The functionality of moving components during the execution has to be implemented in the next prototype. The distributed environment is supported by messaging, service discovery and component management components. The messaging component offers functionality for asynchronous communication and remote procedure calls. The service discovery component is capable of locating services, components and devices. The component management handles the components during their lifecycle. Furthermore, mobility of the hosts can cause disconnections and network partitions and thus makes prediction of disconnections an extremely difficult matter. The next prototype will use DB2 Everyplace engine as the database, which has embedded synchronization and backup mechanisms to prevent data loss in case of disconnections.

The middleware provides functionality for creating and handling multimedia content. It is capable of storage, retrieval, capture and rendering of media. The middleware offers a possibility for creation and invocation of media alerts that

facilitates development of multimedia messaging applications. Besides, it is capable of controlling the network resources and switching the channels according to the policies, which can be defined by multimedia applications. So the middleware is able to adapt to the situation.

A prototype of the CAPNET system has been implemented with the presented functionality. The application developers can use the services of the middleware and focus the design on application-specific issues. The CAPNET middleware is fully developed with Java, so it makes the prototype adaptable to mobile devices. Furthermore, the architecture of the middleware is built to increase the portability of the system to different platforms. The Java approach facilitates the interoperability of the implemented prototype, which is an important issue in the mobile environment. However, implementation of the system in the native language promises to be more effective than implementation in Java, because Java cannot access all the resources of the mobile device. The implemented media services enable rapid development of multimedia messaging applications by supporting a varied set of functions for multimedia creation, delivery, adaptation and processing.

Future work includes improving the architecture to get more reusable components. The middleware is decomposed into several sub-layers, and higher layers of the middleware can be easily ported to different OS. The messaging component in the next prototype is based on Session Initiation Protocol, which helps to solve the network partitioning problems. The component management takes more responsibility from the application layer and is able to move the components from one device to another during their execution with minimal interference to their work. The service discovery component is based on the RDF model. It offers an easy way to combine complex sequences of events and conditions to describe action-triggering rules. The context-based storage uses DB2 Everyplace as the database that definitely gives benefits when considering a dynamic mobile environment. It is capable of automatic data update and synchronization. The media component will be implemented to provide adaptation of media content as well as capture and rendering of video messages with sound.

Finally, we are implementing the prototype on Symbian OS. The basic functionality of core components like messaging and component management will be implemented on Symbian OS during 2004.

6. REFERENCES

- [1] Object Management Group: Common Object Request Broker Architecture: Core Specification, v. 3.0.3, 2004.
- [2] Sun Microsystems, Inc: Java 2 Platform Enterprise Edition Specification 1.4, 2004.
- [3] Gu, T., et al.: A Middleware for Building Context-Aware Mobile Services, *In Proceedings of IEEE Vehicular Technology Conference (VTC-Spring 2004)*, Milan, Italy, 2004.
- [4] Hisazumi, K., et al.: CAMPUS: A Context-Aware Middleware, *The 2nd CREST Workshop on Advanced Computing and Communicating Techniques for Wearable Information Playing*, Nara Institute of Science Technology, Nara, Japan, 2003.

- [5] Duran-Limon, H., et al.: Context-Aware Middleware for Pervasive and Ad Hoc Environments, Computing Department, Lancaster University, Bailrigg, Lancaster, UK, 2000.
- [6] Hawick, K.A., James, H.A.: Middleware for Context Sensitive Mobile Applications. In *Proceedings of workshop on Wearable, Invisible, Context-Aware, Ambient, Pervasive and Ubiquitous Computing*, Adelaide, Australia, 2003, pp. 133-141.
- [7] Lohse M., Repplinger M., Slusallek P.: Session Sharing as Middleware Service for Distributed Applications, *Interactive Multimedia on Next Generation Networks, Proceedings of First International Workshop on Multimedia Interactive Protocols and Systems (MIPS 2003)*, Naples, Italy, 2003.
- [8] Naguib, H., Coulouris, G., Mitchell, S.: Middleware support for context-aware multimedia applications, In *Proceedings of the 3rd IFIP WG 6.1. International Working Conference on Distributed Applications and Interoperable Systems (DAIS 2001)*, Krakow, Poland, 2001.
- [9] Lau, F., Lum, F.: A Context-Aware Decision Engine for Content Adaptation, in *Proceedings of the 8th annual international conference on Mobile computing and networking*, Atlanta, Georgia, USA, 2002.
- [10] Lemlouma, T., Layaida, N.: Adapted Content Delivery for Different Contexts, In Proc. Of 2003 Symposium on Applications and the Internet, IEEE, 2003.
- [11] Dey, A.: Providing Architectural Support for Building Context-Aware Applications, PhD thesis, Georgia Institute of Technology, 2000.
- [12] Coulouris; G., Dollimore, J., Kindberg, T.: *Distributed Systems Concepts and Design*, Addison-Wesley, 2001.
- [13] Dey, A., et al: The Context Toolkit: Aiding the Development of Context-Enabled Applications, in *Proceedings of ACM SIGCHI Conference on Human Factors in Computing Systems (CHI-99)*, Pittsburgh, Pennsylvania, USA, 1999.
- [14] Carpa L. et al.: Middleware for Mobile Computing, In *Proceedings of the 8th Workshop on Hot Topics in Operating Systems*, Elmau, Germany, 2001.
- [15] Efstratiou, C., et al.: Architectural Requirements for the Effective Support of Adaptive Mobile Applications, in *Proceedings of IFIP/ACM International Conference on Distributed Systems Platforms and Open Distributed Processing (Middleware-2000)*, New York, USA, 2000.
- [16] W3C: Synchronized Multimedia Integration Language, <http://www.w3.org/TR/smil20/>, 2004.
- [17] Gabhart, K., Gordon, J.: Wireless Web Services with J2ME, *WebServices Journal*, Volume 2, Issue 2, 2002.
- [18] Gryazin, E., Burmakin, E., Tuominen, O.: Comparison of CORBA and Web Services middleware operation in wireless environments, in *Proceedings of International Workshop on Mobile Computing, (IMC 2003)*, Rostock, Germany, 2003.

Middleware Support for Implementing Context-Aware Multimodal User Interfaces

Pertti Repo

INTERACT Research Group, Infotech Oulu
University of Oulu, Dept. of Inf. Processing Sciences
P.O.Box 3000, FI-90014 OULU, Finland
+ 358 – (0)8 – 553 1897
Pertti.Repo@oulu.fi

Jukka Riekki

Intelligent Systems Group, Infotech Oulu
University of Oulu, Dept. of Electrical and Inf. Engineering
P.O.Box 4500, FI-90014 OULU, Finland
+ 358 – (0)8 – 553 2798
Jukka.Riekki@ee.oulu.fi

ABSTRACT

In context-aware pervasive networking (Capnet) environments the user interface of a service application has to be adapted to whatever type of a device the user is using, taking into account the situation at hand, i.e. the user's current context, as well as user's personal preferences. In addition, new and more natural interaction styles have to be provided for the user. When designing and implementing user interfaces, or generating them automatically at runtime, some *a priori* knowledge of the capabilities of user's device and her personal preferences are required. Furthermore, runtime adaptation calls for mechanisms for recognizing context and delivering it. This paper presents the support that could be provided in middleware level for implementing context-aware multimodal user interfaces and the Capnet middleware user interface architecture model for providing that support. Adaptation algorithms are not discussed. The Capnet middleware backbone provides support for context-awareness via context recognition and automatic delivery of context data through a notification mechanism. For multimodal input and output third party components are required. The proposed architecture model is partly implemented and tested with software prototypes.

Categories and Subject Descriptors

D.2.11 [Software Architectures]: Patterns

General Terms

Design, Experimentation

Keywords

component-based development, middleware, pervasive networking, context-aware applications, multimodal user interfaces

1. INTRODUCTION

The ubiquitous computing or pervasive networking future will change radically the way services will be provided to the user. In

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MUM 2004, October 27-29, 2004 College Park, Maryland, USA. Copyright 2004 ACM 1-58113- 981-0 /04/10... \$5.00

such an environment computers and mobile devices as well as applications are expected to interoperate in an *ad hoc* manner to provide the user a seamless access to surrounding services while taking into account the situation at hand, i.e. the user's current context. The seamless access to services does not only mean transparent communication between devices and applications, but also new and more natural interaction styles in the user interface (UI), e.g. touching or pointing, or using voice commands and computers responding with sound synthesis. In addition, these styles will be used in combination with each other and with the more traditional interaction styles, provided by the graphical UI (GUI), to create so-called multimodal UIs.

The above vision sets new and stringent requirements for the development of middleware as well as application and UI software. The heterogeneity of the mobile devices, that user's want to use to access the services, with different UI capabilities and computing platforms, makes the development even more complex. The application logic has to be separated from the application UI to enable remote access, and the UI has to be adapted to the capabilities of the user's device. When the UI is designed and implemented, some sort of *a priori* knowledge about the device types and their properties has to be known. Even, if there is a smart algorithm capable of adapting the UI at runtime, the algorithm most probably requires such information. In addition, user's personal preferences may affect the way the UI is created.

To alleviate the implementation of remote UIs, the UI software developer has to be provided with easy means for getting information about user's device and its properties as well as components that are available for generating context-aware multimodal UIs in user's device. Since the service application UI will be generated dynamically, i.e. on demand, we believe that the middleware should provide the above mentioned support for the UI software developer. We do not address any actual adaptation algorithms in this paper.

In Capnet (Context-Aware Pervasive NETworking) project [1] we are studying the technologies and building prototypes that will enable the realization of context-aware pervasive networking environments. The goal is to build a middleware and an application framework to support the development of context-aware applications. Our research method is iterative prototyping. A version of the architecture is specified and prototypes are built according to it; the architecture is then evaluated and improved according to lessons learned in prototyping, and so on. We are about to specify

the third version of our architecture and to build new software prototypes for testing the ideas.

We present in this paper the support that could be provided in middleware level for implementing context-aware multimodal UIs. In addition, we present the Capnet middleware UI architecture model for providing the support. The proposed architecture model is partly implemented and tested with software prototypes, but remains partly as a hypothesis to be implemented and tested in the future. The central component in the architecture is the Capnet User Interface component, which acts as a UI front-end to mobile devices to alleviate the development of context-aware multimodal UIs in context-aware pervasive networking environments.

This paper is organized as follows. After this introductory part we look into related work. The third chapter presents how a middleware could support the implementation of context-aware multimodal UIs. The fourth chapter gives an overview of the Capnet middleware and more detailed description of the Capnet User Interface component, which implements the middleware support for implementing UIs. The presentation ends with a conclusion and a look into the directions of our future research.

2. RELATED WORK

Support for implementing UIs is not a common feature in middleware systems. More often application frameworks provide some value adding services for implementing UIs, but in a particular application domain [2]. The problems associated with UI software development in general have not traditionally been the topic of middleware research. Bauer et al. [3] present an exception; a component-based software framework that contains a UI Engine resembling Capnet's UI component. The UI Engine creates user interfaces from XML-based descriptions provided by applications. The difference is that a description is sent to the engine at application's bootstrapping stage only, whereas in Capnet the description can be sent during operation as well, if the application decides to change the UI as a response to a new situation. Finally, Capnet offers a wide set of services for building pervasive applications for mobile users, whereas the framework presented by Bauer et al. is targeted to building augmented reality applications.

Gaia and BEACH are examples of middlewares that support implementing and managing UIs, but have a different emphasis than Capnet. Gaia [4] offers support for building pervasive applications, which are required to implement a modified Model-View-Controller (MVC) architecture model. The emphasis in Gaia is describing the requirements that an application sets to runtime environments using a query language, and the Gaia system taking care of the application adaptation at runtime according to those requirements. The Capnet system does not presume any particular application architecture, except that the UI has to be a separate entity. Abstract UI is one of the key elements in our solution and the main difference when compared to Gaia. In BEACH [5], the emphasis is on a software infrastructure supporting the usage of several user interfaces at the same time by a co-operating group of people.

Several research groups have presented systems that would enable the use of mobile devices as personal or universal remote controllers for other devices in *ad hoc* networking environments. The UI solutions have based on downloadable (mobile) code, e.g. in [6] and [7], or HTML pages and a Web browser, e.g. in [8], or XML

describing a UI abstraction, e.g. in [9] and [10]. We have also been experimenting with XML-based UI techniques and found them very interesting, because of their promise of platform independence. There are a number of XML languages for describing UIs and the number seems to be growing [11], but we recognize that the other techniques have their advantages, too.

While mobile code makes the UI platform dependent, it may provide easier UI implementation for a certain device type or provide a UI that cannot be expressed in an abstract way with HTML or XML. Mobile code might be good for providing e.g. map-based UIs or rich multimedia content. The Java language provides for easy implementation of mobile code and, if the target device contains a Java virtual machine, why not use it.

The Web browser and server have become ubiquitous and their functionality can be extended both on the client and the server side. There exists legacy business solutions built using the Web techniques and should be provided with easy introduction to pervasive networking environments. The UI software developer can rely on a standard language (HTML) and can get more detailed information about the Web browser from the User-Agent field in the requests HTTP header, but is at the same time limited with the capabilities of the Web infrastructure. Some extensions to the User-Agent field have been proposed in [12] so that it could contain information about screen size, colors, input methods, etc. But adding new interaction modalities, such as voice, to the Web-based UIs is, at least currently, difficult, if not impossible.

3. MIDDLEWARE SUPPORT FOR USER INTERFACE IMPLEMENTATION

3.1 Requirements

The UI adaptation, whether it occurs at UI design time or runtime, is a difficult research problem and has been addressed in numerous papers. We have first decided to look what can be done in the middleware level, independent of any application domain, to alleviate the problems associated with the UI adaptation and the creation of context-aware multimodal UIs. By trying to discover what is required from the middleware, we are taking our first steps towards finding our own solutions to those problems.

Whichever technique is chosen for generating the service application UI in a user's mobile device, some information of the target device and its properties are needed either at design time, or at runtime by some sort of adaptation algorithm that automatically generates the UI implementation. To provide better user experience, one could consider taking into account user's personal preferences, too. What actually are valid UI capabilities or user preferences from an adaptation algorithm's point of view, we have not yet specified, and even the algorithm itself could be a matter of choice. But we do include in those capabilities the software components in the target device that are available for generating a UI for an application. The middleware should make it easy to obtain the UI capabilities of a user's device as well as user's personal preferences.

As we recognize that there are many techniques for generating remote UIs, those techniques could be used in combination, too; e.g. to provide on-line help as HTML pages to a UI built from downloaded code. So we find the different techniques comple-

mentary rather than exclusive. Therefore we believe that it is important that the middleware supports different UI techniques.

To provide multimodal UIs, third party software components are needed for e.g. voice recognition and sound synthesis. Also gesture-based input methods, such as pointing or touching, will need a dedicated software component. It is not feasible that every software developer should implement her own components for providing multimodal UIs. Software development in general is more and more component-oriented. Mobile device manufacturers tend to provide application programming interfaces (API) of useful components for software developers. Thus the middleware should also promote component reuse by providing easy means for launching an external software component.

In context-aware environment everything that bears relevance to the user is context data. It is not possible to list that data in detail, nor is it possible to give a comprehensive definition that would include or exclude a thing as a context data. From the service application point of view even the users own actions are context data. The middleware should support the gathering of user actions as context data.

3.2 Supported features

To summarize, we have concluded that to alleviate the implementation of context-aware multimodal UIs, the middleware should provide:

- Information about the user's mobile device: make and model, screen size, colors, etc., i.e. the device's UI capabilities. That information should also contain the software components that are available for generating or rendering the UI, such as a Web or WAP browser, voice recognition and sound synthesis component, XML UI script interpreter, etc.
- Information about user's personal preferences, e.g. that voice commands and sound output are preferred over GUI or vice versa, or that they should be combined in a certain way.
- Information not only about user's current context, but also about her actions, e.g. which application she is currently using.
- An API for launching external software components, such as a Web browser, or for using components, which are able to generate multimodal UIs, e.g. voice recognition and sound synthesizer components.
- An API for enabling the use of mobile code as the remote application UI.
- An API for communication between the service application component and its remote UI, if e.g. XML UI description language is used to define the actual UI.
- An API for modifying the UI according to changes in the user's context or personal preferences.

The above list is what we have discovered thus far in our research. It is neither comprehensive nor exclusive and by no means without problems. What actually constitutes information about the UI capabilities of a device and what are user preferences and how to express that kind of information, are some of those questions

we have not yet thoroughly studied. There are proposals for standards, e.g. the Composite Capabilities / Personal Preferences (CC/PP) model [13] from the World Wide Web consortium that could be used, but would need to be extended to our purposes. The same problems are associated in expressing user's current context. It is obvious that some sort of syntax and semantics have to be agreed on these issues to make the above mentioned middleware support useful for the UI software developer.

4. CAPNET MIDDLEWARE

4.1 Overview

The Capnet architecture divides functionality into components. A simple application consists of a single application logic component and some other components providing services for the application logic, all in the mobile device. At the other end of the continuum, the components in the mobile device and the network form a hierarchy, with the application requesting services from some components, which in turn request services from some other components.

The functionality offered to the applications is divided into domains. Each component (that is not an application component) provides services to a given domain. The domains form a context-aware and pervasive middleware layer that offers a set of generic services for applications. In line with Bernstein's [2] specification of middleware, the Capnet middleware masks the complexity of networks and distributed systems and thereby allows developers to focus on application-specific issues. Furthermore, it factors out commonly used functions into independent components, so that they can be shared across platforms and software environments.

The component management domain controls all components and processes the requests of component access. The connectivity management domain controls communication channels in a dynamic and transparent fashion. The focus is on adapting the wireless connection of the mobile device. The service discovery domain locates resources and services, including other Capnet devices and components. A device hosting Capnet components contains the core component of each of these three domains. Components from the rest of the domains are optional, from which currently implemented are the media, context-based storage, context and UI domains.

The media domain provides applications with a uniform interface for local and distributed multimedia capabilities. Context-based storage acts as ubiquitous data storage. The context domain offers operations for obtaining context information, both synchronously and asynchronously, as context events. It manages the future context as well, e.g. it can produce an event informing the user that a meeting context is anticipated in 30 minutes. The UI domain, which is the topic of this paper, provides services for applications requiring a UI either in the local memory space or in a remote device.

4.2 User Interface Component

4.2.1 Features

Key component in the UI domain is the Capnet User Interface component, which acts as a UI front-end to mobile devices and software components available in the device for generating context-aware multimodal UIs. The conceptual component architec-

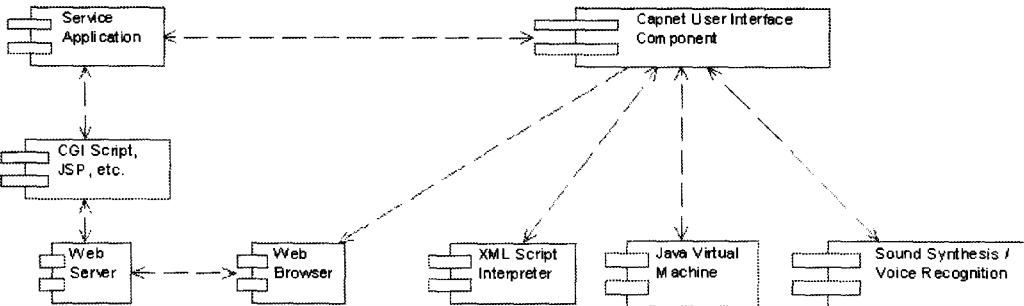


Figure 1. The conceptual user interface architecture of the Capnet middleware.

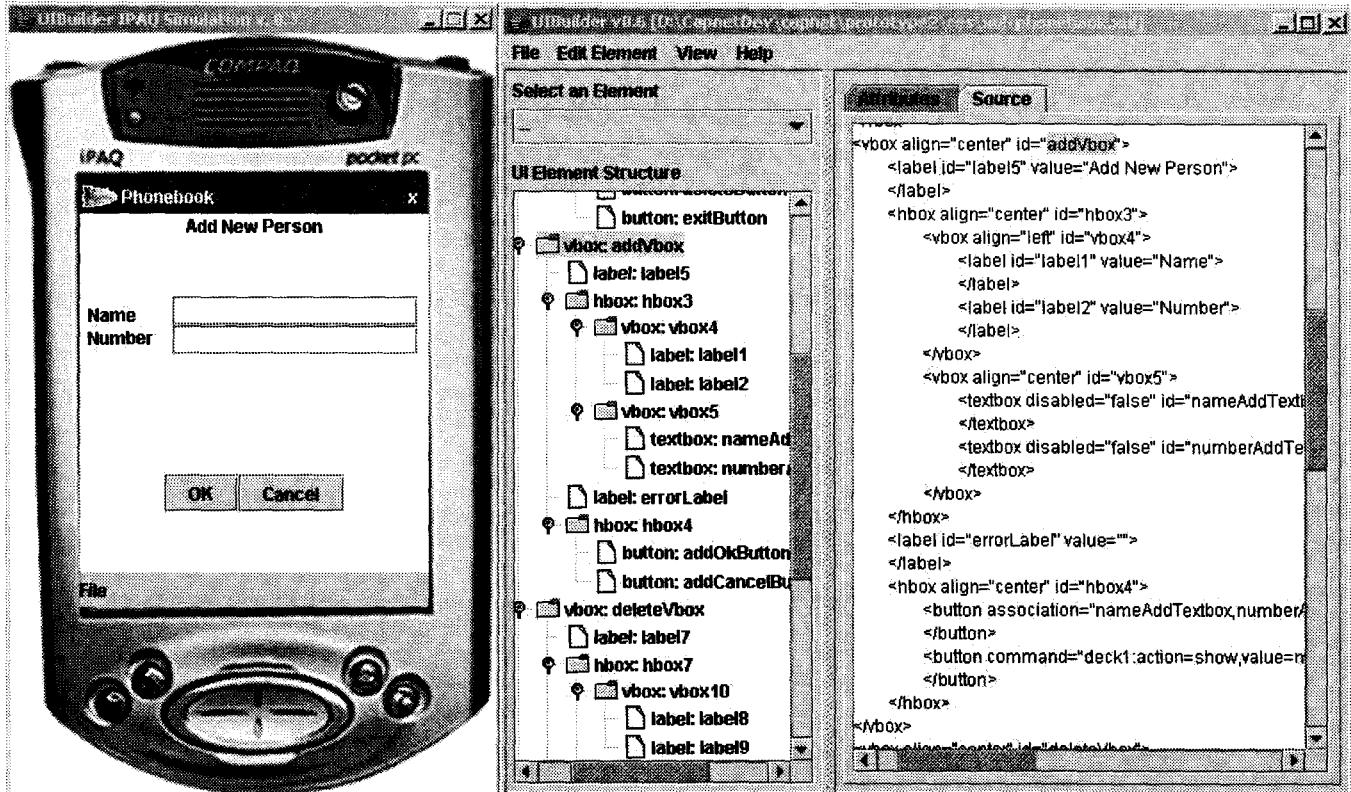


Figure 2. An example of an XML user interface script (modified XUL) is shown in Source panel on the right. Corresponding user interface rendered by the Capnet User Interface component is shown in IPAQ simulation window on the left.

ture is depicted in Figure 1. There will be only one Capnet User Interface component in each device, which is able to host a UI, and its main responsibility is to implement the middleware support for implementing context-aware multimodal UIs. This is the design choice we have found useful in our research and prototype development and intend to develop it further. But the current implementation does not provide all the features described in chapter 3.2, because the UI adaptation to different devices and platform has not yet been the major focus in our prototype development, but it will be in the next iteration round. Our prototypes have thus far been built with the Java language and on PocketPC platform in Compaq iPAQ personal digital assistant (PDA), so the UI capabilities of a device have not yet been dealt with. We have concentrated on the UI adaptation to context changes, which has led to the emphasis of the development of abstract UIs using an XML based UI script language. The user preference management and the UI as a producer of context data have not yet been imple-

mented in our prototypes. Our goal is to enhance the middleware architecture based on the ideas presented in this paper. Multimodal input and output are also not yet implemented in our prototypes. We plan to provide them using third party components, such as ViaVoice from IBM [14] for voice recognition and sound synthesis.

Current version supports design and implementation of application UIs using three different techniques: abstract UIs (XML), plug-in UIs (mobile Java code) and Web-based UIs (HTML). The abstract UIs do not depend on any programming language, device type, operating system or UI toolkit. This is achieved by separating the application from its UI implementation using an XML based UI script language to describe abstract UI elements and their properties. The UI component renders the actual UI implementation on the target device according to the script. An example UI script is presented in Figure 2.

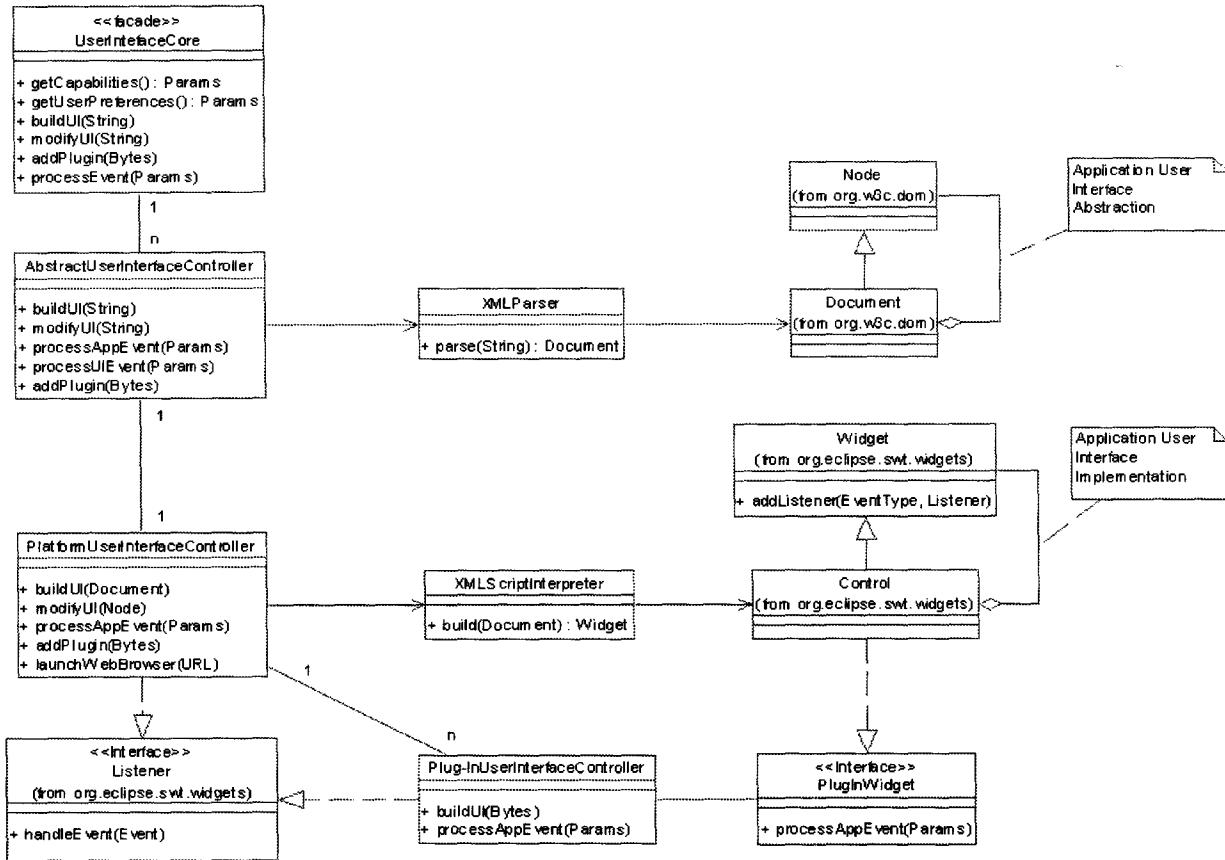


Figure 3. A conceptual class diagram of the Capnet User Interface component.

The application and the UI component share the same UI abstraction, i.e. the document object model (DOM) of the UI script. The component presents a simple interface for building and modifying the UI implementation as well as exchanging messages and events between the application and the UI implementation. Messages are delivered using the Capnet messaging mechanism.

If the designer prefers to implement an application UI, or part of it, using a Java UI toolkit, a UI plug-in can be used. The UI code has to implement specified UI plug-in interface for the UI component to be able to manage the plug-in. The Capnet middleware takes care of transferring the plug-in code from the application component to the UI component, which initializes and shows it. Messages between the application and plug-in are mediated by the component. The designer is responsible for implementing multimodality and adaptivity features of the UI plug-in.

Another option is to use Web-based (HTML) UI. A URL tag in the UI script causes the UI component to launch a Web browser of the target device to open the page the URL points to. After that communication between the application and the UI embedded in the Web browser occurs through the Web techniques (Web servers, CGI-scripts, etc.). In this option too, the designer is responsible for implementing the multimodality and adaptivity features.

4.2.2 Architecture

There are three major layers in the Capnet User Interface component architecture. Figure 3 depicts the conceptual class diagram. On top is the core component layer, composed of one class, which

acts as a façade to application components and other Capnet components. That means that the core component class provides a unified interface for application components, which communicate only through this interface with their UI implementations, if abstract (XML) or mobile code UI is built. Thus the core component layer hides the UI component's internal implementation.

The second (middle) layer is the abstract UI layer for managing the application UI abstraction. The main class in this layer is the AbstractUserInterfaceController class, which manages the creation and modification of the application UI DOM independently of the platform specific UI implementation. The third (bottom) layer is the platform specific layer, which is tightly coupled with the target device. The main class in this layer is the PlatformUserInterfaceController, which is responsible for managing the creation and modification of the application UI implementation and launching the Web browser of the target device. In addition, it manages the UI plug-ins and, in the future implementations, also the input and output modalities provided by the third party components that are available in the device for generating multimodal UIs. The PlatformUserInterfaceController class converts concrete UI events to their abstract counterparts and vice versa, when the application UI is built from an XML script.

The idea behind the component architecture is that the top two layers could remain the same from device to device, and only the platform specific layer should need to be changed. As we have so far used only the Java language, Java AWT and Swing as well as the Standard Widget Toolkit (SWT), from the Eclipse project

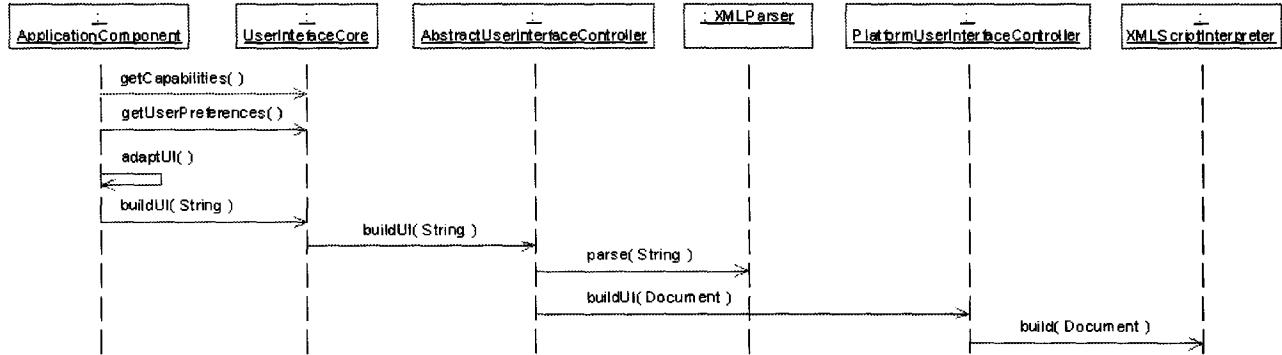


Figure 4. Building an abstract user interface scenario.

[15], have been used as the implementation UI toolkits in the platform specific layer.

4.2.3 Building an Abstract User Interface

The Capnet application framework requires an application component to implement a simple interface for the Component Management component to be able to initialize, start and stop it. The interface is also required for communication between other Capnet and application components. Figure 4 depicts as a conceptual sequence diagram how the application component collaborates with the Capnet User Interface component to build an abstract UI.

The UI generation starts with a user requiring a service. The user is carrying a PDA and wants to use e.g. a projector in a meeting room to show pictures to other meeting participants. The PDA and the projector host the Capnet middleware components. The user initiates the projector service by using a Service Browser application to select it from a list of available services. Many things occur inside the PDA's and the projector's Capnet components, concerning e.g. connectivity management, service discovery and component management, which are not described here. Anyway, the service application component is found and started and the communication between the PDA and the projector is established using the Capnet middleware.

To build the service UI, the service application component first requests from the Component Management component a handle to the Capnet User Interface component inside the user's mobile device. The Component Management component returns a handle to the UserInterfaceCore object. By calling getCapabilities and getUserPreferences methods of the UserInterfaceCore object, the application component may request the device capabilities and user's personal preferences, respectively. An adaptation algorithm might use this information to generate an XML description of an application UI, adapted to the user's device and preferences. According to our hypothesis, the adaptation algorithm might also select to use the Web techniques or mobile Java code for the UI, but this is not depicted in Figure 4. After the UI script is generated the application component calls the buildUI method of the UserInterfaceCore object with the generated XML string as a parameter. In our prototypes the buildUI method is synchronous, so that it returns only when the UI is fully built or throws an exception in case of failure.

Inside the Capnet User Interface component, an AbstractUserInterfaceController object is created for each application requesting a UI. The UserInterfaceCore object forwards the buildUI call

to the AbstractUserInterfaceController object, which uses an XMLParser object to parse the XML string and create the UI Document object. The AbstractUserInterfaceController object in turn creates a PlatformUserInterfaceController object and calls its buildUI method with the newly created UI Document object as a parameter. The PlatformUserInterfaceController object uses an XMLScriptInterpreter object to generate the actual UI widget structure with the UI toolkit available for that platform.

After the application UI is built, the UI events from the UI widget structure flow through the PlatformUserInterfaceController and AbstractUserInterfaceController objects to the UserInterfaceCore object, which sends them to the application component using its processEvent method with the event parameters as key-value pairs. When the application component wants to send an application event to its UI implementation, it calls the UserInterfaceCore objects processEvent method in the same way.

If the application component modifies the UI DOM, it notifies the change by calling the UserInterfaceCore objects modifyUI method with the changes in an XML string as the parameter. The modifyUI method call is forwarded down to PlatformUserInterfaceController object, which takes care of that the UI implementation will be changed accordingly. What causes the need for modifying the UI at runtime, are the changes in user's context that the Capnet middleware delivers automatically.

4.2.4 Tool Support

We have been experimenting with an elementary prototype tool for designing abstract UIs using XML, which is generated by the tool. The tool is called UIBuilder and is depicted in Figure 2. The Capnet User Interface component is embedded in the tool allowing the UI designer to immediately see the results of her design either as a PC window or as a simulated iPAQ window. We plan to extend the tool's simulation capabilities to mobile phones. In addition, we want to add voice command and sound synthesis capabilities to the simulation in the future versions.

The tool support for creating context-aware multimodal UIs for pervasive networking environments is a big research issue of its own and cannot be addressed in more detail in this paper. Our middleware design allows at least some sort of design time support for the UI designer to see the possible results of abstract UI implementations. We have found this to be beneficial, because a tool like UIBuilder enables us to study from the UI designer's point of view the requirements that can be set to middleware support for implementing such UIs; e.g. what has to be designed explicitly and what can be left for the system to generate auto-

matically. Currently our system supports semiautomatic layout management, i.e. the designer builds the UI DOM tree deciding only about whether the elements in a container will reside on top of each other or beside each other, and whether they will be aligned left, center, top, bottom or right. We anticipate that there are limits how far a UI can be abstracted and with the UIBuilder tool we are trying to find out about those limits in our future research.

5. CONCLUSION

We have presented in this paper the support that a middleware system could provide for implementing context-aware multimodal user interfaces in pervasive networking environments. In addition, we have presented the Capnet user interface middleware architecture model that partly implements that support, and partly presents a hypothesis to be implemented and tested in the future. The ideas are based on iterative prototyping. There is e.g. an instant-messaging application that is able to adapt the UI according to changes in user's context using the abstract UI techniques provided by the Capnet User Interface component [16].

The implementation of context-aware multimodal user interfaces requires some *a priori* knowledge about the capabilities of the user's device as well as about her personal preferences, either at design time, or at runtime, if there is an adaptation algorithm that is able to generate the user interface automatically. We include in the device's capabilities also the software components that are available for generating user interfaces, such as the Web browser or a sound synthesizer component. There are many valid techniques for implementing remote user interfaces, which should be supported by the middleware, in addition to using third party components that are able to generate multimodal user interfaces.

The key component in the Capnet user interface architecture model is the Capnet User Interface component, which acts as a user interface front-end to mobile devices. Its main responsibility is to provide the middleware support for implementing context-aware multimodal user interfaces. One of the benefits that our user interface model provides is that the Capnet User Interface component, while managing user interfaces for several applications, can provide contextual information about user's actions, e.g. what application she is currently using, etc. This information is then available as context data for other applications through the Capnet middleware context delivery system.

There are a number of questions that we have not yet addressed. What are actually the user interface capabilities of a device and how to describe them? What are the user's preferences and how to describe them? Are there any types of contexts or user's preferences that do not affect the application logic and could be handled by the User Interface component automatically? If the user interface could be automatically generated, what kind of an adaptation algorithm is required? We are studying these questions while developing the third version of our middleware architecture and building new prototypes for testing the ideas.

6. ACKNOWLEDGMENTS

Many thanks to fellow Capnet researchers for their invaluable comments and improvement ideas using the Capnet user interface architecture in several prototypes. Special thanks to researcher Mikko Perttunen for the initial idea of the Capnet User Interface

component being not only a consumer but also a producer of context data.

The Capnet project is part of "NETS – Networks of the Future" technology program [17] and funded by the Finnish National Technology Agency (TEKES) and following companies: Hantro, IBM, Nokia Mobile Phones, Serv-IT and Telia Sonera Finland.

7. REFERENCES

- [1] <http://www.mediateam.oulu.fi/projects/capnet/?lang=en>
- [2] Bernstein, P.A.: Middleware: a model for distributed system services. Communications of the ACM. Vol. 39, Num. 2. ACM Press (1996) 86-98.
- [3] Bauer, M., Bruegge, B., Klinker, G., MacWilliams, A., Recher, T., Riss, S. & San, M. (2001): Design of a component-based augmented reality framework. Int Symp on Augmented Reality 2001, New York, USA, 45-54.
- [4] Roman M, Hess CK, Cerqueira R, Panganat A, Campbell RH & Nahrstedt K (2002): A middleware infrastructure for active spaces. IEEE Pervasive Computing 1:74-82.
- [5] Tandler, P. (2001): Software infrastructure for ubiquitous computing environments: Supporting synchronous collaboration with heterogenous devices. Proceedings of Ubicomp 2001, Atlanta, Georgia, 96-115.
- [6] Beard M. & Korn P.: What I Need is What I Get: Downloadable User Interfaces via Jini and Java. CHI (2001) Extended Abstracts, 15-16.
- [7] Newman M.W., Izadi S., Edwards W.K., Sedivy J.Z. & Smith T.F.: User Interfaces When and Where They are Needed: An Infrastructure for Recombinant Computing. Proceedings of UIST'02 (2002), Paris, France, 171-180.
- [8] Kindberg T. & Barton J.: A Web-based Nomadic Computing System. Computer Networks. Vol. 35, Num. 4. Elsevier (2001) 443-456.
- [9] Vandervelpen C., Luyten K. and Conix K.: Location-Transparent User Interaction for Heterogeneous Environments. Proc. HCI International, Crete, Greece. Vol. 2 (2003), 313 - 317.
- [10] Vanderheiden G., Zimmermann G. & Trewn S.: A Standard for Controlling Ubiquitous Computing and Environmental Resources from Any Personal Device. Proc. HCI International Crete, Greece. Vol. 2 (2003), 499-503.
- [11] <http://xml.coverpages.org/userInterfaceXML.html>
- [12] <http://www.w3.org/TR/NOTE-agent-attributes-971230.html>
- [13] <http://www.w3.org/2001/di/>
- [14] <http://www-306.ibm.com/software/voice/viavoice/>
- [15] <http://www.eclipse.org/swt/>
- [16] Perttunen, M. & Riekki, J.: Inferring Presence in a Context-Aware Instant Messaging System. Accepted to the 2004 IFIP Int. Conference on Intelligence in Communication Systems (INTELLICOM 04), Bangkok, Thailand, 23-26 Nov. 2004.
- [17] <http://akseli.tekes.fi/Resource.phx/tivi/nets/en/index.htm>

Efficient Method for Multiple Compressed Audio Streams Spatialization

Abdellatif Benjelloun Touimi, Marc Emerit
Sound and Speech Technology and Processing
Laboratory

France Telecom R&D Division
2 avenue Pierre Marzin - 22307 Lannion, France

Phone: +33 2 96 05 36 77

email: {abdellatif.benjellountouimi,
marc.emerit}@francetelecom.com

ABSTRACT

This paper deals with the spatialization of multiple compressed audio streams. A new approach is proposed based on the combination of subband-domain filtering methods with linear decomposition methods of HRTFs filters, and more generally the filters used by sound spatialization techniques (binaural, transaural, ambisonic, etc). Such a combination allows both computation complexity reduction and memory size saving for sound spatialization systems of multiple compressed audio signals. These advantages are very useful for any implementation on limited capabilities terminals (computation power and memory size) which is the case of mobile and portable devices.

KEYWORDS

3D sound, streaming, compressed domain processing, subband filtering, HRTFs, linear decomposition.

1. INTRODUCTION

Nowadays, the existence of efficient compression audio formats allows the transmission of complex and structured sound scenes including multiple sound sources. In general, these sound sources are processed in such a way to generate the illusion of natural sound environment in terms of sources positions and room effects (reverberation). For instance, MPEG-4 standard defines transmission methods of complex sound scenes including compressed audio with their spatialization parameters (position, room effect). Such transmission is made over constrained network and sound rendering depends on the user terminal. The compression of sound sources uses mainly transform and subband coding (MPEG-1/2 Layer I-III, MPEG-2/4 AAC, Dolby AC-3...) which is based on transforming the time-domain represented signal to subband-domain and quantizing the

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MUM 2004, October 27-29, 2004 College Park, Maryland, USA.
Copyright 2004 ACM 1-58113-981-0/04/10... \$5.00

Jean-Marie Pernaux
ASSYSTEM Services
Paris, France

email: jeanmarie.pernaux@wanadoo.fr

resulting parameters for transmission.

With the new generation of multimedia mobile terminals, new enhanced audio applications and services are possible. 3D sound streaming is one of such applications. The auditor can enjoy spatialized sound by streaming multiple audio streams and listening to them using simple stereophonic headphone or other rendering system (like in car) connected to his mobile terminal or PDA. Teleconferencing on mobile terminals is another application where audio spatialization can bring a service enhancement. Such sound spatialization could be static or dynamic and interactive. In this last case, the auditor can modify the position of the sound sources interactively. The processing can be done either at the terminal level or at the server or in a combined way.

At the terminal level, the main operations before rendering the received audio streams consist in decompression operation and spatialization processing. One of the characteristics of mobile terminals and portable devices consist in their limited capabilities in terms of computation power and memory size. In such a context, the implementation of spatialization techniques is not an easy task. The integration and the factorization between the operations of compression/de-compression of audio streams and their spatialization become very interesting. Such an approach has already been proposed using subband-domain filtering techniques to reduce the computational complexity [2, 3, 7, 8]. However, the problem of memory size is still persisting. On the other hand, the linear decomposition technique of HRTFs, and generally spatialization filters, has been proposed [5]. One of its main advantages is that the filtering operations, and hence the computation complexity, depend much less on the total number of sources to be positioned in the space. We propose in this paper the combination of subband-domain filtering with linear decomposition techniques of spatialization filters. Such a new method allows both computation power minimization and memory size saving.

This paper is organized as follow. Section 2 gives an overview of sound spatialization and especially binaural synthesis. In section 3, we describe the principle of spatialization methods based on linear decomposition. Section 4 presents a summary of subband-domain filtering methods. Section 5 explains the proposed combined method and describes the corresponding

architectures. Finally, section 6 gives different applications of the proposed method and its benefits.

2. SOUND SPATIALISATION

Sound spatialization techniques aim at reproducing 3D sound field from monophonic sources. It consists in creating the illusion to the auditor that the perceived sounds have precise position in the space and adding particular acoustical properties of the real space (reverberation ...). One of the efficient sound sources positioning techniques is the binaural synthesis. In this technique the monophonic signal is filtered by the left and right Head Related Transfer Functions (HRTFs) that model the signal transformation due to the propagation channel from the sound source to the ear canals. The HRTFs functions depend on the frequency, f , and on the position, given by the azimuth and the elevation, $[\theta, \varphi]$. The HRTFs can be modeled and implemented by finite or infinite impulse response (FIR or IIR) filters. An HRTFs-database corresponding to all space positions is built and stored.

For a sound source to be positioned in the space position $[\theta, \varphi]$, the source signal is filtered by the left and rights HRTFs filters corresponding to such a position. In the case of multiple sources (S_n , $n = 1 \dots N$) positioning, the left and right signals corresponding to each of the sources are first computed and then all the left (respectively right) signals are summed to produce the left and right signals to be listened (figure 1).

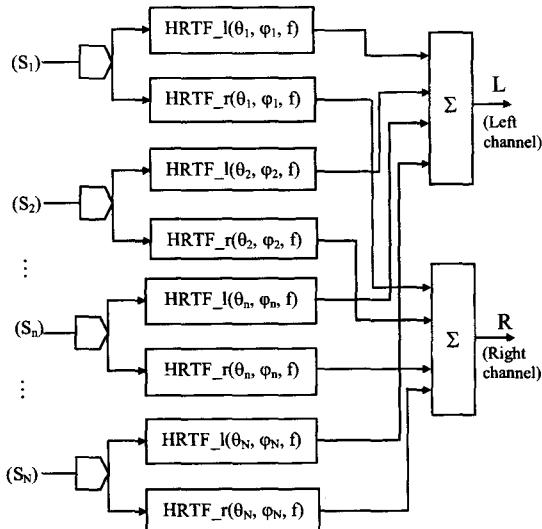


Figure 1 : Static binaural synthesis technique.

Dynamic spatialization consists in time varying the positions of the sound sources. In such a case the left and right HRTFs filters should be modified continually. The conventional used solution deploys in parallel two binaurals filters corresponding to the initial position, $[\theta_1, \varphi_1]$, and final position $[\theta_2, \varphi_2]$. The signal giving the illusion of mobile source between the two positions is obtained by dynamic linear combination of the signals resulting from the two filtering processes.

If N is the number of sources to be spatialized, $2N$ filtering operations are needed in the static case and $4N$ in the dynamic case. The computation complexity is hence proportional to the number of sound sources.

Other spatialization techniques exist like binaural transaural or ambisonic. The proposed method can be extended to such techniques as will be explained further.

3. SPATIALISATION TECHNIQUES BASED LINEAR DECOMPOSITION

3.1 Binaural Synthesis

For binaural synthesis spatialization, each HRTF is first decomposed to obtain a minimum phase filter characterized by its module and a pure delay, τ_n (ITD: Interaural Time Delay) [6]. The all-pass component of the filter is neglected. The spatial and frequency dependencies of these HRTFs modules are then separated using a linear decomposition over a basis of K reconstruction filters, $[L_k(f)]_{0 \leq k \leq K-1}$, which are the same for all the space positions and depend only on the frequency, f . Any HRTF filter module is then expressed as a linear combination of such filters:

$$|HRTF(\theta, \varphi, f)| = \sum_{k=1}^K C_k(\theta, \varphi) L_k(f). \quad (1)$$

The weighting coefficients, $C_k(\theta, \varphi)$, $0 \leq k \leq K-1$, are spatial functions depending only on the position $[\theta, \varphi]$. The positioning of any sound source, S , is performed only by varying the coefficients, $C_k(\theta, \varphi)$, $0 \leq k \leq K-1$. The number K of filters that need to be stored is then reduced and fixed as it merely corresponds to the basis filters, $[L_k(f)]_{0 \leq k \leq K-1}$.

In the case of multiple sources positioning, each sound source signal, S_n , $n = 1, \dots, N$, is weighed by the coefficients, C_{kn} , $1 \leq k \leq K$, corresponding to its position $[\theta_n, \varphi_n]$, for the left (respectively right) ear. The N signals of the sound sources weighted by the directional coefficients C_{kn} , $1 \leq n \leq N$, are summed and then filtered by the reconstruction filter $L_k(f)$, for the left (respectively the right) ear. The resulting signals corresponding to each reconstruction filter are then summed to be listened by the right (respectively left) ear. Figure 2 illustrates such an implementation, where the notation C_{kn} corresponds to the k^{th} directional coefficients for sound source S_n for the left channel and D_{kn} for the right channel. The connection between the directional coefficients and the filters are represented only for the left channel. They are similar for the right channel.

In this method, the addition of any sound source is simply performed by introducing its corresponding positioning weighting coefficients in the weighted sums. This is the main advantage over the classical binaural synthesis where each new sound source needs the use of two additional filters. The K basis filters are shared by the entire present sources. The number of needed filters is $2K$ both for the static and dynamic

spatialization cases. It is independent from the number of sound sources to be spatialized.

Different methods have been proposed for the linear decomposition to determine the spatial functions and the reconstruction filters. The method described in [5] is based on Karhunen-Loeve decomposition. In [1], a Principal Component Analysis (PCA) is applied to the HRTFs for such decomposition. Its implementation is also given in [11].

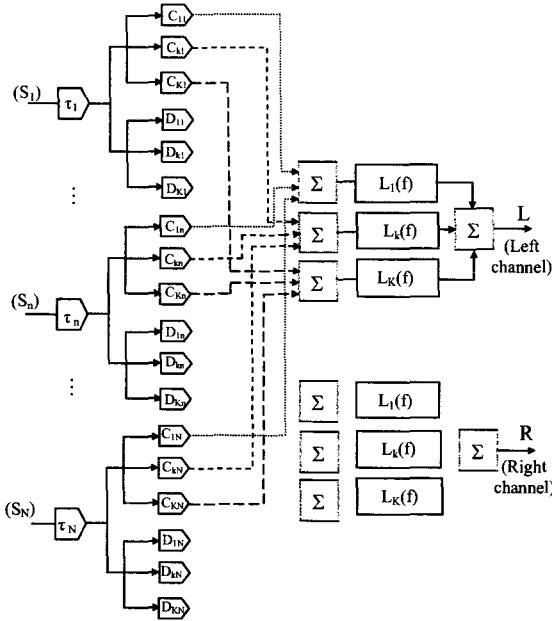


Figure 2: Binaural synthesis implementation based on HRTFs linear decomposition.

3.2 Generalisation: Spatial Sound Synthesis

The principle described above could be generalized to other sound spatialization systems like the ambisonic. Indeed, any sound rendering system could be modeled as follow:

- A real or virtual sound picking system called sound field *encoding*. It consists of recording N sound signals from the sound scene or simulating such signals (virtual encoding) and then encoding them to obtain K compact signals.
- Rendering sound system which consists of decoding the issued signals to adapt them to the sound rendering transducers (e.g. stereophonic headphone). The K signals are transformed to P signal inputs of the P sound transducers.

In the example of binaural synthesis, the encoding phase could be realized by recording real sounds using microphones introduced in the ears of an artificial head or using simulation by convolving a monophonic sound with the HRTFs corresponding to the desired direction. From one or several sound sources we obtain two signals (left and right ear) corresponding to the binaural encoding stage. The decoding stage in this case consists in applying the encoded signals to a stereophonic headphone.

Encoding and decoding methods based on the decomposition of HRTFs transfer functions over a filters basis are also possible. The method represented in figure 2 corresponds to the case where the encoding and decoding phases are combined. It is the case when 3D sound rendering is entirely realized in the terminal. The terminal receives N monophonic signals from N sources and realizes 3D sound rendering and mixing.

For the general case, the expression (1) could be generalized to any encoding type, for N sound sources and encoding format with K output signals, as follow:

$$E_k(f) = \sum_{n=1}^N C_{kn}(\theta, \varphi) S_n(f), \quad 1 \leq k \leq K. \quad (2)$$

The general relation for a decoding format containing K signals, $E_k(f)$, and sound rendering format with P signals is given by:

$$D_p(f) = \sum_{k=1}^K B_{pk}(f) E_k(f), \quad 1 \leq p \leq P. \quad (3)$$

For a given sound rendering system, the filters $B_{pk}(f)$ are fixed and depend only on its disposition.

In the case of binaural synthesis, $B_{pk}(f)$ corresponds to the reconstruction functions $L_k(f)$. The decoding filters matrix is hence diagonal. Other encoding/decoding couples are possible for such a case. One consists in generating K encoded signals, as in (2), which will be filtered lately by the reconstruction functions at the decoding phase, as in (3). Such a scenario corresponds to the case where the encoding is done on the server and the decoding at the terminal.

For the example of an order 1 2D-ambisonic system, we have an encoding format with $K = 3$ signals X, Y, W , for N sound sources, given as follow:

$$\begin{pmatrix} E_1 \\ E_2 \\ E_3 \end{pmatrix} = \begin{pmatrix} X \\ Y \\ W \end{pmatrix} = \begin{pmatrix} \sum_{n=1}^N S_n \\ \sum_{n=1}^N \cos(\theta_n) S_n \\ \sum_{n=1}^N \sin(\theta_n) S_n \end{pmatrix}. \quad (4)$$

The filters $B_{pk}(f)$ of the ambisonic decoding system depend on the considered rendering system.

4. SUBBAND-DOMAIN SPATIALISATION

4.1 Principe and architecture

Spatialization of audio signals represented in compressed domain needs fully decoding operation, since it applies the spatialization processing (binaural, transaural or ambisonic) in the time-domain, and if needed, re-encoding the resulting signals. Figure 3 illustrates such a system for the case of N audio sources spatialisation. An alternative to this heavy computational method consists in manipulating directly subbands signals after a partial decoding operation of the compressed audio streams.

For an implementation on a terminal, the spatialized subband signals are then synthesized using the synthesis filter banks (figure 4a). If they have to be transmitted, as in a server spatialization implementation, partial re-encoding operations are needed (figure 4b). The partial decoder consists mainly in dequantization; the synthesis filter bank is eliminated. As the synthesis filter bank takes the major part of the decoding process (e.g. 70% for MPEG-1 Layer II decoder) and their number in the new system is reduced, the computational complexity decrease essentially results from this elimination. The resulting computational gain is proportional to the number of audio streams to be spatialized.

Manipulation of subband signals for spatialization is not straightforward. The main problem lies in the transposition of the spatialization filters (e.g. HRTFs) used in the time-domain processing to subband-domain. The transposition should be designed in such way to have a transparent equivalence between the signals obtained by time-domain and subband-domain spatialization. Indeed, any processing of subbands signals introduces a modification of the aliasing components resulting from using non-perfect filters before the decimation in the analysis filter bank. If this modification is not adequately performed, aliasing cancellation will not be assured by the synthesis filter bank, hence causing audible artifacts in the reconstructed signal. Several solutions have been proposed to this subband domain filtering problem [10, 7]. A generic framework has been developed in [2]. It describes the transposition of any rational filter (FIR or IIR) from the time-domain to the subband-domain of any critically sampled perfect reconstruction filter bank. The principle of this framework is summarized in the next paragraph.

We notice that in figures 3 and 4 we represent, and without loss of generality, the spatialization system corresponding to a binaural synthesis. There are two outputs signals (left and right). Such a scheme can be replaced by any other spatialization system with K or P signals at output.

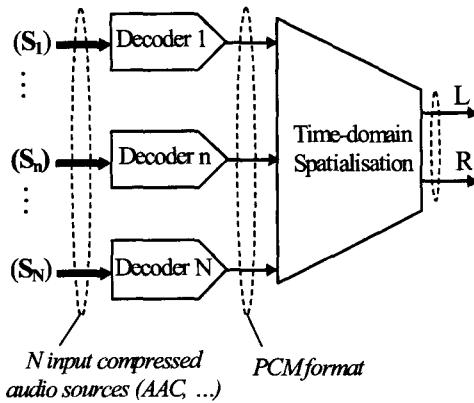
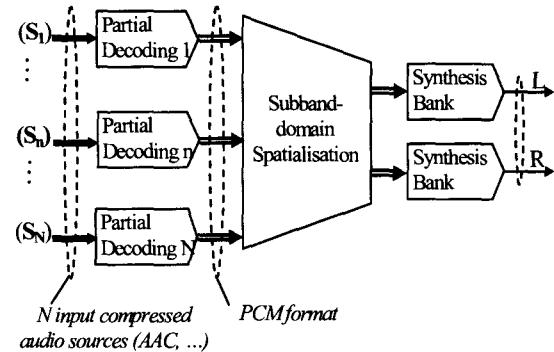
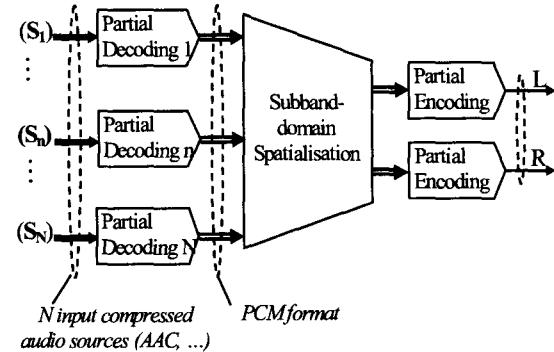


Figure 3: Spatialization of N coded audio sources at the level of the terminal.



(a) Implementation at the terminal.



(b) Implementation at the server.

Figure 4 : Compressed-domain spatialization (binaural synthesis) of N coded audio sources.

4.2 The Generic Subband-Domain Filtering Method

Let us consider an M -band critically sampled perfect reconstruction filter bank given by its analysis and synthesis filters, $H_k(z)$ and $F_k(z)$, $0 \leq k \leq M-1$, respectively. Consider also a rational filter given by its scalar transfer function, $Q(z)$. The transposition of filtering operation by such a filter in the subband-domain of the filter bank results in an $M \times M$ filtering matrix, $T(z)$, operation to be applied on the subband signals. Indeed, the vector, $V(z)$, of the filtered subband signals could be obtained from the input subband signals vector, $U(z)$, as follow:

$$V(z) = T(z)U(z), \quad (5)$$

where $U(z) = [U_0(z) \ U_1(z) \ \dots \ U_{M-1}(z)]^T$ and $V(z) = [V_0(z) \ V_1(z) \ \dots \ V_{M-1}(z)]^T$. The principle of the basic scheme of subband filtering is illustrated in figure 5.

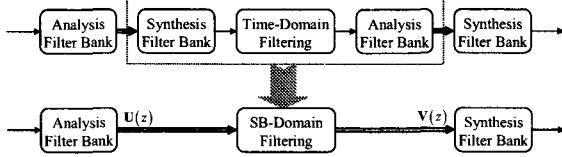


Figure 5: Subband domain filtering principle illustration.

The filters elements of the matrix, $T(z)$, are given by the following formula [2]:

$$[T(z)]_{n,k} = [z^{M-1} H_n(z) Q(z) F_k(z)]_{\downarrow M}, \quad (6)$$

for $0 \leq n, k \leq M-1$, where $\downarrow M$ stands for factor M decimation. This expression is equivalent to [12]:

$$[T(z)]_{n,k} = G_{nk}^{M-1}(z), \quad (7)$$

where $G_{nk}^{M-1}(z)$ denotes the $(M-1)^{th}$ polyphase component, resulting from an M -order decomposition, of the product filter $G_{nk}(z) = H_n(z)Q(z)F_k(z)$. This polyphase component could be obtained straightforwardly if $G_{nk}(z)$ is a FIR-filter. For the case of an IIR-filter, a method to derive it is given in [4].

In practice the matrix, $T(z)$, is not fully used, only the filters of the main diagonal and other some adjacent sub-diagonals have to be considered for the implementation. It is replaced by the sparse matrix, $\tilde{T}(z)$, whose lines are obtained as follow:

$$\begin{bmatrix} 0 & \cdots & 0 & T_{n[n-\delta]_M}(z) & \cdots & T_m(z) & \cdots & T_{n[n+\delta]_M}(z) & 0 & \cdots & 0 \end{bmatrix}, \quad (8)$$

$0 \leq n \leq M-1,$

where $[x]_M$ stands for x modulo M . δ denotes the number of bands which recover significantly in one side with the subband n . It depends on the considered filter bank. For example, in the case of an MDCT filter bank, δ could be taken equal to 2 or 3. For the Pseudo-QMF filter bank used in MPEG-1 Layer I and II coders, δ is equal to 1.

Despite such a simplification, the dimension of $\tilde{T}(z)$ remains high and depends on the number of subbands of the filter bank used by the coder (e.g. 512 or 1024 in the case of MPEG-4 AAC). It is one of the drawbacks of this method especially when it is necessary to store a high number of transposed filters.

Efficient methods could be used to implement the formula (5) based on $\tilde{T}(z)$ characteristics. For the case where $Q(z)$ is a FIR-filter, an Overlap and Add (OLA) method is indicated in [7]. For the case of an IIR-filter case an efficient structure is developed in [2].

5. COMBINED METHODS DESCRIPTION

5.1 Implementation architecture

As noticed in 4.2, the subband-domain spatialization requires a huge memory to store spatialisation filters matrix. Such a

problem becomes especially relevant if the whole matrix resulting from the transposition of all the HRTFs filters, in the example of binaural synthesis, representing all the needed space positions, must be stored. Even if the subband-domain audio spatialization is computationally efficient, it is less interesting for the applications with limited memory capabilities which are the case of mobile terminals and portable devices.

The proposed new method consists in combining subband-domain filtering with linear decomposition techniques of HRTFs filters, and more generally the filters used in spatialization sound techniques (binaural, transaural, ambisonic, etc). Besides the computation reduction due to compressed-domain filtering, the representation of all the spatialization filters on a common basis makes the number of filters independent of the number of sound sources. Such techniques enhance the computation complexity minimization in the case of multiple sound sources while reducing the memory size of spatialization filters.

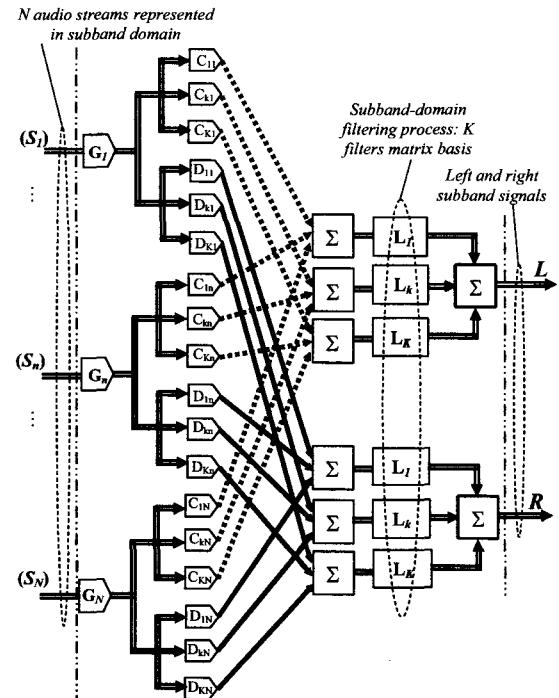


Figure 6: Subband domain spatialisation based on HRTFs linear decomposition.

The general architecture of the combined method is similar to that represented in figure 4a and 4b. The new subband-domain spatialization system corresponding to binaural synthesis based on HRTFs linear decomposition is represented in figure 6. It corresponds to the transposition of the system of figure 2. In this system:

- G_n , $1 \leq n \leq N$, are the filtering matrix resulting from the transposition of the ITD filters, τ_n , to subband domain.

- $\mathbf{L}_k(z)$, $1 \leq k \leq K$, are respectively the filtering basis matrix resulting from the transposition of the basis filters $[\mathbf{L}_k(f)]_{0 \leq k \leq K-1}$.
- The set of weighting coefficients C_{nk} , D_{nk} , $1 \leq n \leq N$, $1 \leq k \leq K$, are the same as in figure 2. They remain unchanged in the transposition to subband-domain as they are frequency independent.

Figure 7 presents a more general scheme corresponding to the transposition of sound sources spatial encoding and decoding equations, (2) and (3), in the subband-domain. The transposition of the encoding formula (2) is straightforward as the coefficients $C_{kn}(\theta, \phi)$ are frequency independent. They are used as combination coefficients of the subband samples. In figure 7b illustrating the spatial decoding in the subband domain, $\mathbf{B}_{pk}(z)$, $1 \leq p \leq P$, $1 \leq n \leq K$, are filters matrices resulting from the transposition of the basis filters $B_{pk}(z)$. We have to notice that all the time-domain scalar filters to subband domain matrix filters transposition operations are based on the technique described in section 4.2, mainly equations (7) and (8). The efficient methods indicated in this section are also used for the implementation.

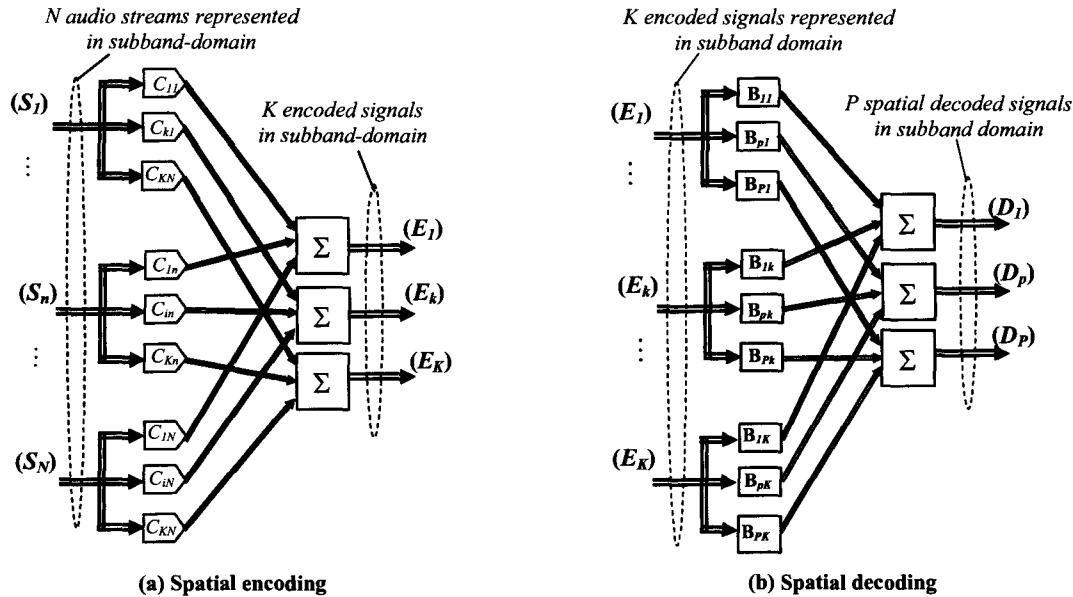


Figure 7. General scheme of spatialization in subband-domain based linear decomposition technique.

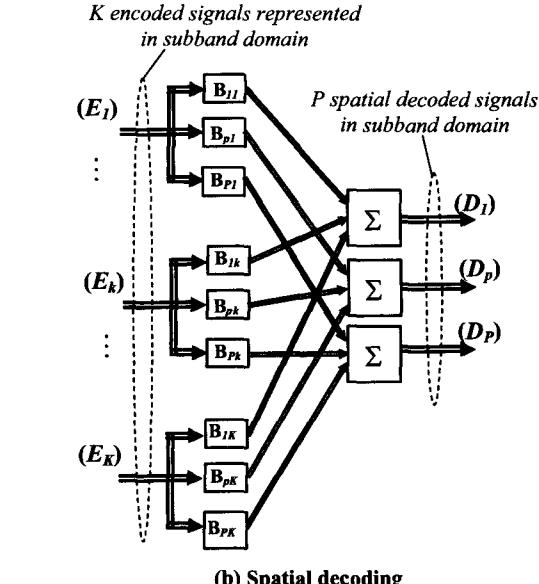
6. APPLICATIONS

As discussed earlier in this paper, sound spatialization of compressed audio streams can be implemented at different points of the transmission chain (server, network nodes or terminals). The considered application and the used communication architecture can guide the choice of the point of the implementation. In teleconferencing application, the spatialization processing will be done at the level of end terminals in the case of decentralized architecture and at the level of a multipoint control unit (MCU) in

The spatial encoding and decoding represented in figure 7 could be done separately or combined. They can be performed either in the server or at the terminal side. For example the encoding stage can be implemented at the server side and the decoding at the terminal side. In such a case, the compressed audio streams stored in the database are partially decoded, spatially encoded and then partially re-encoded to transmit K bitstreams. At the terminal level, a partial decoding is performed on the received bitstreams, spatially decoded, and finally synthesised by filters banks to obtain time-domain signal to be rendered on the transducers. The spatial decoding stage depends on the rendering system. Such degree of freedom is useful for mobile terminals. Indeed, the user can choose between stereophonic headphone in mobility or other enhanced rendering systems when in stable position (e.g. in a car).

Several decoding systems could also be cascaded as in figure 8:

- Binaural encoding and decoding at the server (stereophonic headphone) and transaural decoding at the terminal.
- Binaural encoding at the server and binaural followed by transaural decoding at the terminal.
- Ambisonic encoding at the server and ambisonic decoding followed by binaural at the terminal.



the case of a centralized architecture. For audio streaming applications, notably over mobile terminals, the spatialization could be done at the level of the terminals or at the server side, or off line during the content creation.

The context of MPEG-4 is particularly interesting to apply the combined method. In this context multiple audio streams included in structured sound scenes are streamed or downloaded. The scene description is given in special format called *AudioBIFS* (Binary Format for Scene description). At the terminal side, each audio stream

is decompressed (MPEG-4 Audio compression) and post-processing is performed to build the sound scene.

In addition to audio streaming, other 3D sound services on mobile terminals could be implemented based on the described method. 3D audio chat, spatialized sound messages broadcasting, games are few examples of such services.

7. CONCLUSION

The combination of subband domain filtering techniques and spatialization based linear decomposition is the main idea developed in this paper. A solution has been described both for binaural synthesis and general spatialization methods. However such a method deals only with the static spatialization case. One interesting and remaining problem is the case of dynamic spatialization dealing with mobile sources. The linear decomposition method of spatialization filters is much profitable for such a case as it has been discussed in section 3.1. However, this case implies a time-varying filtering in the subband-domain. For the moment no efficient method has been proposed to solve this problem. It is one of the issues to overcome in order to achieve optimal combined compression/spatialization systems in dynamic context.

8. REFERENCES

- [1] Abel, J. S. and Foster, S. H. *Method and apparatus for efficient presentation of high-quality three dimensional audio*. Patent WO94/10816, Aureal Semiconductor Inc., Fremont, CA, USA, 1997.
- [2] Benjelloun Touimi, A. A Generic Framework for Filtering in Subband Domain. In *Proceeding of IEEE 9th Workshop on Digital Signal Processing*, Hunt, Texas, USA, October 2000.
- [3] Benjelloun Touimi, A. *Compressed-domain Audio Signal Processing: Techniques and Applications* (in French). Ph.D. Thesis, Ecole Nationale Supérieure des Télécommunications de Paris, May 2001.
- [4] Bellanger, M. *Digital Processing of Signals*. Wiley, 1984.
- [5] Chen, J. Vercoe, B. P. and Hecox, K. E. *Method and apparatus for producing directional sound*. Patent WO96/13962, Wisconsin Alumni Research Foundation, Madison, WIS, USA, 1996.
- [6] Jot, J.-M. Larcher V. and Warusfel, O. Digital Signal Processing Issues in the Context of Binaural and Transaural Stereophony. *AES 98th Convention*, February 25-28, 1995, Paris.
- [7] Lanciani, C. A. and Schafer, R. W. Subband-Domain Filtering of MPEG Audio Signals. In *Proceeding of IEEE International Conference on Acoustic, Speech, Signal Processing*, 1999.
- [8] Lanciani, C. A. and Schafer, R. W. Application of Head-related Transfer Functions to MPEG Audio Signals. In *Proceeding of 31th Symposium on System Theory*, March 21-23, 1999, Auburn, AL.
- [9] Lanciani, C. A. *Compressed-Domain Processing of MPEG Audio Signals*. Ph.D. Thesis, Georgia Institute of Technology, 1999.
- [10] Levine, S. N. Effects Processing on Audio Subband Data. In *Proceeding of ICMC*, Hong Kong, 1996.
- [11] Martin, J. and Dudouet, E. *Dispositif de simulation sonore et procédure pour réaliser un tel dispositif*. French Patent FR-2782228, 1998.
- [12] Vaidyanathan, P. P. *Multirate Systems and Filter Banks*. Prentice Hall, Englewood Cliffs, NJ, 1993.

Digital Photo Similarity Analysis in Frequency Domain and Photo Album Compression

Yang Lu^{*}, Tien-Tsin Wong, and Pheng-Ann Heng
Computer Science and Engineering Department
The Chinese University of Hong Kong
Hong Kong
{ylu, ttwong, pheng}@cse.cuhk.edu.hk

ABSTRACT

With the increasing popularity of digital camera, organizing and managing the large collection of digital photos effectively are therefore required. In this paper, we study the techniques of photo album sorting, clustering and compression in DCT frequency domain without having to decompress JPEG photos into spatial domain firstly. We utilize the first several non-zero DCT coefficients to build our feature set and calculate the energy histograms in frequency domain directly. We then calculate the similarity distances of every two photos, and perform photo album sorting and adaptive clustering algorithms to group the most similar photos together. We further compress those clustered photos by a MPEG-like algorithm with variable IBP frames and adaptive search windows. Our methods provide a compact and reasonable format for people to store and transmit their large number of digital photos. Experiments prove that our algorithm is efficient and effective for digital photo processing.

Categories and Subject Descriptors

I.4 [Image Processing and Computer Vision]: Compression (Coding)—*Approximate methods*; I.5 [Pattern Recognition]: Clustering—*Algorithms, Similarity measures*

General Terms

Algorithms, Design, Measurement

Keywords

Similarity Analysis, Frequency Domain, Energy Histogram, Photo Album Sorting, Adaptive Clustering, Image Compression, JPEG, DCT, MPEG

*contact author

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
Copyright 200X ACM X-XXXXXX-XX-X/XX/XX ...\$5.00.

1. INTRODUCTION

With the wide use of digital camera and internet pictures, more and more people have built up their photo album of the daily life events and beautiful landscapes easily. Taking photographs with a digital camera is so convenient and low cost that it is easy for a user to generate thousands of photographs per year. 700 digital photos, each with the resolution of 2048×1532 , occupy over 1GB space on disk. This flood of photographs presents a storage challenge: how can a user find a compact and reasonable format to store or transmit his or her collection, and how to effectively organize and manage the large collections of digital photos?

Recently, digital photo album management has attracted much attention. Research has been carried out on clustering photographs reasonably and providing effective searching engine. Loui and Savakis [6] developed a automated event clustering system, which mainly focused on the metadata of digital photos, i.e. the date/time information, as well as the color histograms. Their algorithms are effective when data/time information is available and indicates the photo events margins. If people take photos of different subject at the same time, the photo content should be considered for image similarity analysis. Lim et al. [5] studied home photo content modeling for Personalized Event-Based Retrieval system and tried to address the semantic gap between feature-based indices and retrieval preferences. They focused on the event taxonomy for home photos and designed a system that utilize the low-level feature-based representations of digital photos to generate visual content of photos. Furthermore, Yeh and Kuo [12] suggested an iteration-free clustering (IFC) algorithm to modify the existing binary tree indexing structure for a nonstationary image database without reapplying the K-means algorithm to the database, where the database updating problem is modeled as a constrained optimization problem.

However, all of previous research work focus on the retrieval and indexing problems of digital photo album, but not sorting or compression. And most of them are available only for low resolution images. As digital cameras and other equipments supply much high resolution images, users prefer take photos no less than 800×600 in size, which provide high-definition color information to scan and develop the photographs. As we can see that people tend to take several photos in the same place with the relatively constant portraits, and generally the landscape and scenery also have high similarity between each other. The changes of objects and background in some photos are not considerable,

which provide good cross-image correlation for image similarity analysis and compression. We can utilize the high similarity information to sort and group the randomly placed disordered photos and remove the redundancy information among them.

In this paper, we explore the digital photo similarity analysis techniques in frequency domain, and then perform photo album sorting, clustering and compression algorithms. We aim to sort these miscellaneous photos according to their semantic similarities, and categorize into different clusters. Based on these sorted photo sequence, we further compress all the photographs by a MPEG-like algorithm with variable IBP frames and adaptive search windows for motion compensation. The overall system design is illustrated in Fig. 1.

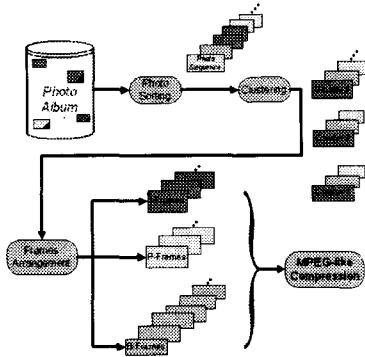


Figure 1: Photo Album Process Flowchart

2. FEATURE ANALYSIS IN FREQUENCY DOMAIN

Due to the limitations of space and time, most of the photos are stored and distributed in compressed format, among which JPEG is the most popular standard applied in digital camera and internet multimedia data. Previous digital photo album processing need to decompress photos into un-compressed domain before carrying on with other existing image processing and analysis techniques. This is not only time consuming, but also computationally expensive. Compared with the conventional image feature detection and texture analysis approaches on pixel domain, performing image analysis in frequency domain directly become more beneficial:

- 1) Exempt from the huge workload of decompressing every image.
- 2) Process is performed on less amount of data since most of the frequency coefficients tend to zero.
- 3) Utilize the image features contained in frequency domain directly, such as the mean and directional texture information that DCT (Discrete Cosine Transform) coefficients provide.

Therefore, a new research stream which is conducted to do image analysis and feature extraction directly in frequency

domain has drawn much attention, and some techniques explored for editing and analysis the compressed images with frequency coefficients become hot topics in recent years.

Smith and Rowe [9] first implemented several operations directly on JPEG data, such as scalar addition, scalar multiplication, pixel-wise addition, and pixel-wise multiplication of two images on RLE blocks. With these algorithms, one can execute the traditional image manipulation operations on compressed images directly, yielding performance 50 to 100 times faster than that of manipulating decompressed images. Based on image analysis and processing techniques in compressed domain, some applications of image indexing using DCT frequency coefficients have been developed. Chang [1] studied video indexing, image matching, and texture feature extraction with compressed images. Shneier and Abdel-Mottaleb [8] suggested a method of generating feature keys of JPEG images with the average value of DCT coefficients computed over a window. During retrieval, images with similar keys are assumed to be similar. However, there is no semantic meaning associated with such image similarities. Quad-tree structure has been introduced into image indexing system by Climer and Bhatia [2]. They designed a JPEG image database indexing system based on quad-tree structure with leaves containing relevant DCT coefficients. Quad-tree is adopted as the signature of original image that stores the average DC coefficients which correspond to the average value of the 8×8 block. The system generates a set of ranked images in the database with respect to the query, and allows user to limit the output number of matched images by changing the match threshold. Considering the real time indexing efficiency, Feng and Jiang [3] calculated only the first two moments, mean and variance, of original images directly in DCT domain using DC and AC coefficients. Based on these two statistical features, their JPEG compressed image retrieval system is robust to translation, rotation and scale transform with minor disturbance.

The aforementioned work focused on the different applications by manipulating image data in DCT frequency domain. Those achieved efforts somehow exploited the possibility of using DCT coefficients for describing image information, which is also the base point for us to do digital photo processing in frequency domain.

2.1 Feature Extraction

JPEG derives its name from Joint Photographic Experts Group and is a well-established standard for compression of color and gray scale images for the purpose of storage and transmission [11][7]. In the last decade, JPEG has been developed as the most popular image format widely adopted in electronic equipments. As a result, most photos and pictures are stored and transmitted in this format.

The JPEG encoding standard for full-color images is based on Discrete Cosine Transform (DCT). The compression process is started by dividing the rectangular image into 8×8 blocks, and pixels from each block are transformed from spatial to frequency domain by DCT. The forward and inverse 2D DCT transform are given by:

Forward :

$$F(u, v) = \frac{C_u C_v}{4} \sum_{i=0}^7 \sum_{j=0}^7 \cos \frac{(2i+1)u\pi}{16} \cos \frac{(2j+1)v\pi}{16} f(i, j) \quad (1)$$

Inverse :

$$f(i, j) = \frac{1}{4} \sum_{u=0}^7 \sum_{v=0}^7 C_u C_v F(u, v) \cos \frac{(2i+1)u\pi}{16} \cos \frac{(2j+1)v\pi}{16} \quad (2)$$

where

$$C_u, C_v = \begin{cases} \frac{1}{\sqrt{2}} & : \text{for } u, v = 0 \\ 1 & : \text{otherwise} \end{cases}$$

According to JPEG codec standard as shown in Fig. 2, a full decompression process includes: (a) entropy decoding with Huffman coding tables, (b) DCT coefficients dequantization, and (c) inverse DCT to reconstruct the image blocks. Traditional image analysis and processing need to go through all the decompression steps before executing any pixel domain manipulation. To improve efficiency and utilize the well transformed frequency information, we extract the DCT coefficients at point "T" in the Figure 2, that is dequantizing to obtain all DCT coefficients but not doing inverse DCT transform.

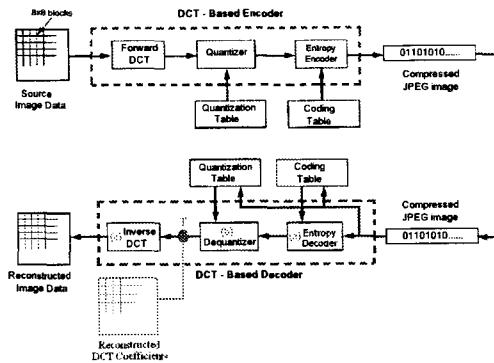


Figure 2: JPEG Codec Diagram

In each transformed 64-point (8×8 block) DCT frequency coefficients, the first top-left DC coefficient corresponds to the average intensity of the image component block, and the remaining AC values contain the information regarding intensity changes within a block along different directions at different scales. In JPEG, the zig-zag pattern approximately orders the basis functions from low to high frequencies. Depending on the compression ratio, most AC coefficients are equal to zero after quantization, especially those corresponding to the high frequencies in the zig-zag pattern. As a result, most energy of original image is concentrated on the non-zero low frequency coefficients and the rest zero AC coefficients can be ignored [4][3][2]. However, two totally different images may have the same average intensity value, i.e. the DC coefficient, we propose to adopt first few non-zero AC coefficients to represent the detailed texture information of original image as well. We build up our feature set as the method shown in Fig. 3.

We create the DC and AC_i ($0 \leq i \leq 63$) coefficient matrix, which consist of the corresponding DC and AC_i coefficients from each 8×8 block. Thus the size of these DCT coefficients matrix is less than that of original image by 64 times. The DC coefficients matrix can be regarded as the reduced and

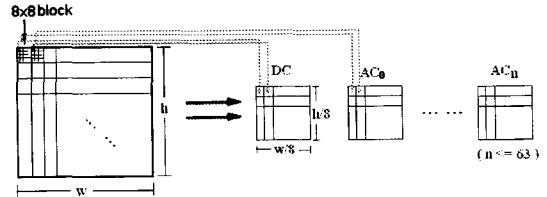


Figure 3: Feature set from frequency domain

smooth approximation of the original image, and these AC_i matrices represent the texture information along different directions.

As the digital photos are all in color, we should choose to adopt which color channel's frequency coefficients, Y channel alone or all of Y, Cb and Cr channels. Typically, the preferred algorithm is based on Y (luminance) component, that is because:

- i) human visual system is more sensitive to Y than to two other chrominance components.
- ii) both JPEG and MPEG standards retain more information in Y than the other two components.

We have performed the same sorting algorithm based on the frequency coefficients extracted from all YCbCr color channels and Y channel alone respectively. Experimental results suggest that DC combined with first 31 AC coefficients of Y luminance alone is most effective. They not only provide the sufficient texture information but also take far less computing costs than traditional methods, as these coefficients have already been calculated and are readily available in the frequency data of compressed image.

2.2 Energy Histogram

Histogram is one of the primitive tools used in image information analysis. It is generally tolerant to image rotation and modest object translation, and can include scale invariant through the means of normalization. Given a discrete color image defined by some color space, e.g. RGB, the color histogram is obtained by counting the number of times each discrete color occurs in the image array [10]. Statistically, color histogram denotes the joint probability of the intensities of these three color channels. Similarly, an energy histogram of DCT coefficients is obtained by counting the number of times an energy level appears in a DCT blocks set of compressed image. The energy histograms of DC and AC matrices can be written as:

$$H_{DC}[k] = \sum_{i=0}^m \sum_{j=0}^n \begin{cases} 1 & : \text{for } DC[i, j] = k \\ 0 & : \text{otherwise} \end{cases} \quad (3)$$

$$H_{AC_t}[k] = \sum_{i=0}^m \sum_{j=0}^n \begin{cases} 1 & : \text{for } AC_t[i, j] = k \\ 0 & : \text{otherwise} \end{cases} \quad (4)$$

where, $DC[i, j]$ denotes the dequantized DC value at the (i, j) location, $AC_t[i, j]$ denotes the t th AC coefficient value at (i, j) . $H_{DC}[k]$ and $H_{AC_t}[k]$ are the corresponding energy level with the value of k in DC and t th AC coefficients respectively.

The pixels of the original Y component in spatial domain are coded with 8 bits. However, after the DCT transform, the sizes of DC coefficients become 11 bits with the range of $[-1024, 1023]$. That means the number of original histogram bins should be 2048. However, the histogram bins of DC coefficients can be reduced to a smaller size, such as 1024, 512, or 256 [4]. Our experimental results suggest that 512 bins energy histogram is most effective.

Fig. 4 shows the spatial histogram and frequency energy histogram of the picture lena, where the DC and AC frequency coefficients are not quantized. From the energy histograms we can see that DC component is actually a coarse or rough approximation of the original image. AC coefficients approach to zero along the zig-zag scan sequence from low to high frequency. That is the higher AC frequency coefficients, the more probability it will be zero. Especially after performing quantization, the energy histogram of AC high frequency coefficients will concentrate on zero much more.

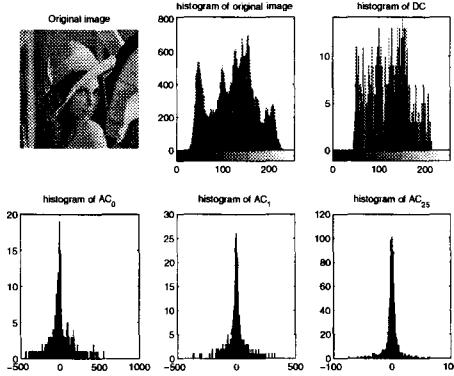


Figure 4: Spatial Histogram and Frequency Energy Histogram of Lena

We further normalized these energy histograms to the region $[0, 1]$ by dividing them with the total number of the coefficients appeared in corresponding vectors. The normalized histograms will have a more general probability form, which will allow us to do photo similarity analysis with different resolutions and more efficient in terms of computation.

2.3 Photo Distance

To evaluate how similar two images are, we explore a reasonable metric to calculate the distance value from their feature vectors. Lots of experimental works have been carried out to examine the best metric according to different applications[4]. In our project, we adopt histogram intersection as it is most effective and natural way for histogram similarity calculation.

The histogram intersection metric is first proposed by Swain and Ballard [10] for color image retrieval in the spatial domain, the formula is defined as:

$$H(X, Y) = \frac{\sum_{i=1}^n \min(X_i, Y_i)}{\sum_{i=1}^n Y_i} \quad (5)$$

where X and Y denote two n-bins color image histograms. $H(X, Y)$ turns out to be the normalized degree of similarity. Since we have normalized the histogram bins, i.e., $\sum_{i=1}^n Y_i =$

1 and $\sum_{i=1}^n X_i = 1$, based on this similarity value, we can obtain the corresponding distance simply by:

$$d(X, Y) = 1 - \sum_{i=1}^n \min(X_i, Y_i) \quad (6)$$

We obtain 32 DC and AC distance values by performing histogram intersection on those DC and AC energy histograms respectively. The final similarity distance of two images can be calculated by weighted summarization of them:

$$D(X, Y) = w_{dc} d_{dc}(X, Y) + \sum_{i=1}^{31} w_{ac_i} d_{ac_i}(X, Y) \quad (7)$$

where

$$w_{dc} + \sum_{i=1}^{31} w_{ac_i} = 1$$

$d_{dc}(X, Y)$ denotes the DC component distance, and $d_{ac_i}(X, Y)$ denotes the i th AC component distance. w_{dc} and w_{ac_i} are distance weights arranged to the DC and AC components respectively. $D(X, Y)$ is the final distance of image X and Y in terms of similarity.

We calculate the distances of all every two photos in the album, obtain a symmetric distance matrix D , where $D(x, y) = D(y, x)$ and $0 \leq D(x, y) \leq 1$, 0 means most similar, 1 means totally different, as shown in Fig.5, take 6 photos for example.

	1	2	3	4	5	6
1	0	0.26933	0.93812	0.53154	0.78208	0.077682
2	0.26933	0	0.10314	0.049354	0.88388	0.80128
3	0.93812	0.10314	0	0.096763	0.38972	0.084215
4	0.53154	0.049354	0.096763	0	0.86388	0.55988
5	0.78208	0.88388	0.38972	0.86388	0	0.078765
6	0.077682	0.80128	0.084215	0.55988	0.078765	0

Figure 5: Photo distance of 6 images

3. PHOTO ALBUM SORTING

We calculate the distances of all every two photos in the album, then got a symmetric distance matrix D , where $D(x, y) = D(y, x)$ and $0 \leq D(x, y) \leq 1$, 0 means most similar, maybe the same photos, 1 means totally different. Based on this distance matrix, we sorte all of the photos in the album.

Starting with the first photo P_0 in the album, we search the minimal distance value in all $D(0, i)$, $0 < i < n$, which represent the similarity distances from all other photos to P_0 . The minimal distance, for instance $D(0, j)$, means the most similar photo is P_j . Then we put the photo P_j adjacent to P_0 in the sorted photo queue. We set the sorting flag of P_j to be true, which means this photo has been sorted, avoiding the following process calculating and sorting it again. Iteratively, based on the last photo in the sorted photo queue, we search the most similar photo among the unsorted ones, and put it to the end of the sorted photo queue. This process execute repeatedly until all of the photos in album are sorted. As a result, the most similar photos will be adjacent to each other.

The following pseudocode demonstrates our photo album sorting algorithm. P is the input photo album, Q is the sorted photo queue, and N is the number of all photos.

Algorithm 3.1: PHOTOALBUMSORTING(P)

```

 $i \leftarrow 0;$ 
 $j \leftarrow 0;$ 
 $Q[i] \leftarrow P[j];$ 
for  $i \leftarrow 1$  to  $N$ 
    for  $j \leftarrow 0$  to  $N$ 
        if  $D[i, j]$  is minimal
            if  $P[j]$  is not sorted &  $P[j] \neq P[i]$ 
                 $Q[i] = P[j];$ 
return ( $Q$ );

```

4. PHOTO ALBUM CLUSTERING

Based on the sorting sequence of all photos, we can categorize them into different clusters according to the similarity distances. Because different photo cluster has different texture complexity and similarity distance distribution, as shown in Fig. 6, the distance variances of different clusters are variable. The distance variances among the clusters of sky and cloud photos are much more less than that of the clusters of people activities.

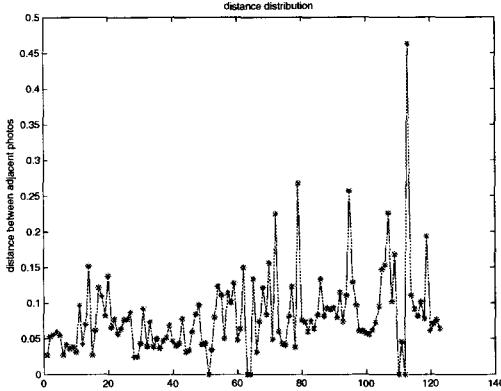


Figure 6: Distance distribution of adjacent photos along sorting sequence

A simple global threshold for cutting different clusters is proved ineffective since it can not adapt the local variation of different clusters. Statistically, the first two moments, i.e., the mean μ and variance σ , can describe the general distribution of a group variables. We propose to utilize the adaptive μ and σ to describe the distance distribution of updating clusters.

$$\mu = \frac{1}{N-1} \sum_{i=1}^{N-1} D(i, i+1) \quad (8)$$

$$\sigma = \left(\frac{1}{N-1} \sum_{i=1}^{N-1} (D(i, i+1) - \mu)^2 \right)^{\frac{1}{2}} \quad (9)$$

where $D(i, i+1)$ denotes the distance between adjacent photos i and $i+1$. We define the clustering threshold T as:

$$T = \mu + K \times \sigma \quad (10)$$

where K is a parameter we set to control the range of variance. Clustering threshold T is used to measure whether the current photo belongs to this cluster or not. The bigger K , the more probability this photo belongs to current cluster.

Beginning with the first photo, we examine the similarity distance $D(i, j)$ between every two adjacent photos i and j , where $j = i + 1$. We calculate the mean μ and variance σ of distance values in current cluster using Equation(8) and (9), and compare the distance $D(i, j)$ with the clustering threshold T .

If $D(i, j) \leq T$, photo j is classified to this cluster and used to update this cluster's μ and σ at the same time :

$$\mu_{new} = \frac{\mu_{old} \times N + D(i, j)}{N + 1} \quad (11)$$

$$\sigma_{new} = \left(\frac{1}{N} \sum_{i=1}^N (D(i, i+1) - \mu_{new})^2 \right)^{\frac{1}{2}} \quad (12)$$

And then the clustering threshold T of this cluster is therefore changed by μ_{new} and σ_{new} :

$$T_{new} = \mu_{new} + K \times \sigma_{new} \quad (13)$$

This updated threshold T_{new} will be used to examine the following photo $j+1$ along the sorted photo sequence in the next step.

Otherwise, if $D(i, j) > T$, photo j is different from the photos in current cluster in terms of similarity distance, we start a new cluster with it instead. The new cluster's μ and σ are initialized to 0.

After that, we carry out the same process to examine the distance of photo j and $j+1$ again. This operation is executed iteratively until clustering all photos. The following pseudocode describes our photo album clustering algorithm. Q is input sorted photo queue, C is the matrix of clustering results, k is the number of cluster, and j is the number of the photo belongs to this cluster.

Algorithm 4.1: PHOTOALBUMCLUSTERING(Q)

```

 $k \leftarrow 0;$ 
 $j \leftarrow 0;$ 
 $C[k, j] \leftarrow Q[0];$ 
for  $i \leftarrow 0$  to  $N$ 
    if  $D[Q[i], Q[i+1]] \leq T$ 
         $C(k, j) \leftarrow Q[i+1];$ 
         $\mu \leftarrow \frac{\mu \times (j-1) + D[Q[i], Q[i+1]]}{j};$ 
         $\sigma \leftarrow \left( \frac{1}{j} \sum_{t=0}^j (D[C[k, t], C[k, t+1]] - \mu)^2 \right)^{\frac{1}{2}};$ 
         $j \leftarrow j + 1;$ 
    else
         $k \leftarrow k + 1;$ 
         $j \leftarrow 0;$ 
         $C[k, j] \leftarrow Q[i+1];$ 
         $\mu \leftarrow 0;$ 
         $\sigma \leftarrow 0;$ 
return ( $C$ );

```

5. PHOTO ALBUM COMPRESSION

After clustering, we obtain several photo clusters, each contains the same daily event or almost the same scenery.

The changes of objects and background among the photos in the same cluster are not considerable, that provides good cross-image correlation for compression. Therefore, we can remove those redundancy from clusters by a MPEG-like algorithm.

5.1 Variable IBP frames

Based on the clustering results, we rearrange those photos in terms of compression. Basically, in MPEG, the fundamental element is frame, which corresponds to the photo in our album. There are three types of frames, i.e., I-frames, P-frames, and B-frames. The I-frames are intra coded, they can be reconstructed without any reference to other frames. The P-frames are forward predicted from the last I-frame or P-frame, it is impossible to reconstruct them without the data of another frame (I or P). The B-frames are both forward predicted and backward predicted from the last/next I-frame or P-frame, there are two other frames necessary to reconstruct them. P-frames and B-frames are referred to as inter coded frames.

GOP (Group of Pictures) in MPEG represents the distance between two adjacent I-frames, which allows random access because the first I-frame after the GOP header is an intra picture that means that it doesn't need any reference to any other picture.

In our program, one cluster is one GOP. The length of GOP is variable, which depends on the number of the photos in each cluster. We regard the first photo in a GOP as I-frame, the last one as P-frame, and the others between I and P frames as B-frames. Therefore, there is only one P-frame in our GOP , as shown in Fig. 7.

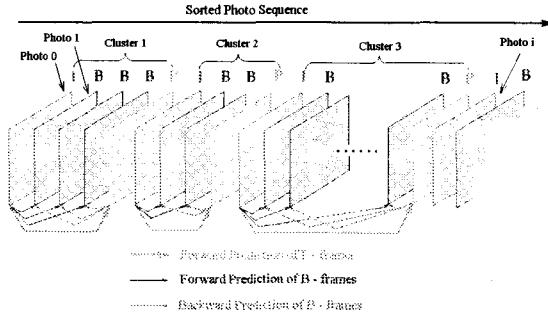


Figure 7: Frame Arrangement

We carried out some experiments to explore the optimal parameter K in Equation (10) for photo clustering. Currently, when K is equal to 1.0, the result is preferable in terms of image similarity, as described in Fig. 8.

5.2 Adaptive Search Window

After arranging all necessary frame structures, we design the search windows in motion compensation. The P-frame is forward predicted from the I-frame in current GOP with fixed search window size of 32×32 . For other B-frames in GOP, along the sorting sequence, the position of each B-frame implies the similar relationship of it to I-frame and P-frame. That means, if this B-frame is nearer to I-frame than that to P-frame, it is more similar to I-frame than that to P-frame. We therefore assign bigger search window in

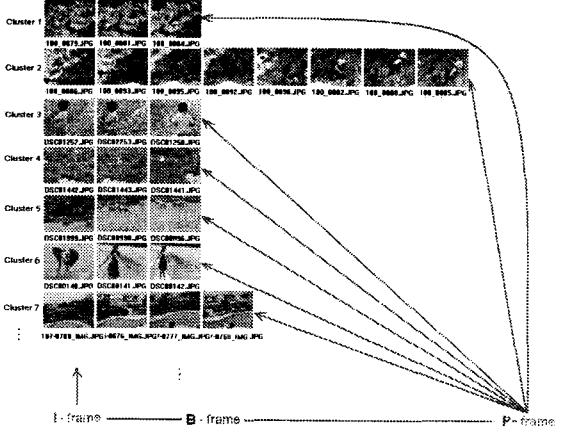


Figure 8: Clustering results

I-frame. Otherwise, we assign bigger search window in P-frame. Equation (14) and (15) illustrate the B-frame search window size calculation.

$$S_I(B_i) = 16 \times \left(\frac{N-i}{N} + 1 \right) \quad (14)$$

$$S_P(B_i) = 16 \times \left(\frac{i}{N} + 1 \right) \quad (15)$$

where $S_I(B_i)$ denotes i th B-frame's search window size from the I-frame, and $S_P(B_i)$ is its search window size from P-frame. N is the total number of B-frames in current GOP, i ($1 < i < N$) is the number of current B-frame.

6. EXPERIMENTS AND RESULTS

In our experiments, the digital photo album consists of approximately 130 digital camera images, with the size of 1024×768 each. They were captured by the different digital cameras, i.e., SONY, Canon, Kodak, and Olympus. There are many different types of daily life pictures in this digital album, such as sceneries, buildings, and people activities. In the landscape photos, there are seas, mountains, flowers and trees. In people activities photos, there are swimming, hiking, party, and holiday pictures. Moreover, there are some buildings and furniture photos as well.

We compared the performance with different sorting parameters by counting the number of sorting error happened, which means the times of separating the same class photos. Based on experimental results, we prefer adopting the DC and first 31 AC coefficients of Y channel in DCT frequency domain, using energy histogram intersection to measure the similarity distance between the JPEG compressed photos. We also arrange different weights to DC and AC components respectively to obtain the final distance between every two photos, i.e., $w_{dc} = 0.2$ and $\sum_{i=1}^{31} w_{aci} = 0.8$.

Our system begins with loading the digital photo album as user specified folder path. A dialog pops up for setting the photo similarity analysis parameters before executing the sorting algorithm, as demonstrated in Fig. 9. User can select to use what similarity distance calculation metric, what color channel's frequency coefficients (Y channel alone or all of Y, Cb and Cr channels), and how to arrange the distance

calculating weights to the DC and AC components respectively.

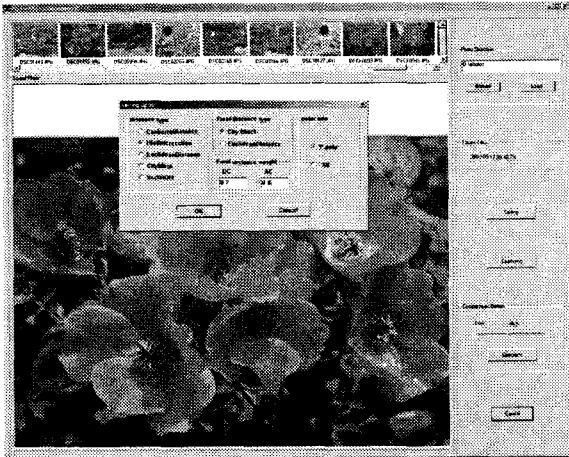


Figure 9: System Interface

The first list-box on the left part displays the thumbnails of all the unsorted digital photos in the original album, and the second list-box displays the thumbnail results of our sorting algorithm. When user clicks any thumbnail in these two list-boxes, the original digital photo corresponds to the highlighted thumbnail will be laid out in the following bigger image frame with original detailed information, and the taken time of this picture will be extracted from the metadata and displayed in text-box on the right side.

Fig. 10 gives the results of our sorting algorithm. The first lines of (a) and (b) are the unsorted photo sequences in the original album. The second lines are the sorted photos by our algorithm. Compared with the original unsorted photo sequence, we can reorder all the photos in current album according to their similarity degree with each other. As the clustering results output demonstrated in Fig. 11, we can group the most similar photos into the same cluster.

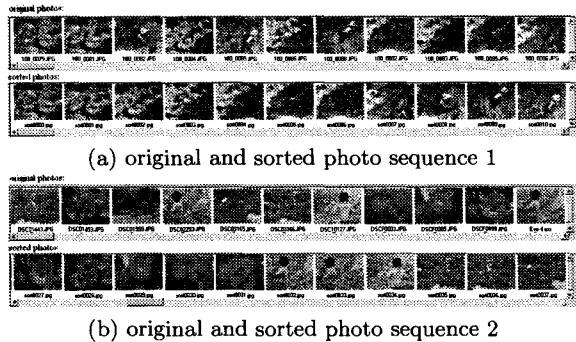


Figure 10: Sorting results output

Based on the clustering results, we arrange the IBP frames cluster by cluster, and perform MPEG-like compression scheme with adaptive search window according to the position of B-frames. The compression performance is given in Fig. 12.

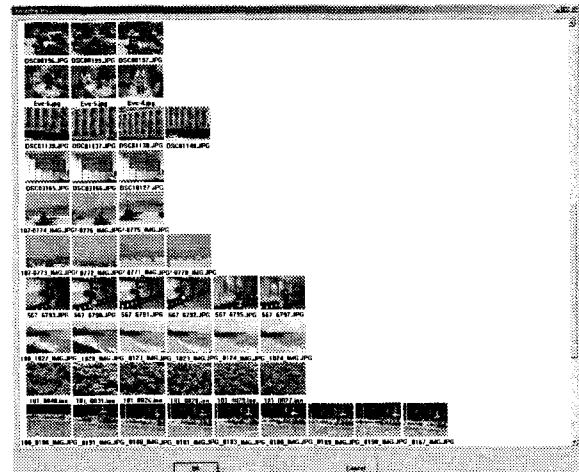


Figure 11: Clustering results output

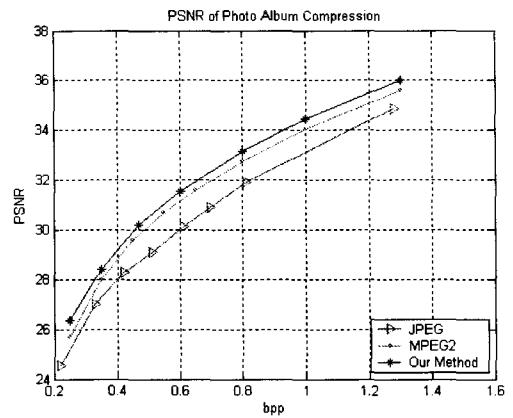


Figure 12: Compression performance

Take one picture for example, as shown in Fig. 13, traditional JPEG picture appears much arti-blocks in high compression ratio. MPEG picture also loses much detail information, for instance the people swimming in the sea, the texture of trees, and the wave movement. Our methods retain more image quality by intelligent motion estimation and compensation, which depend on proper sorting sequence, reasonable clustering groups, and the cross-image similarity (redundancy) correlation.

7. CONCLUSION

The flood of photographs presents a management and storage challenge: how can a user find a compact and reasonable method to search and store his or her collection? The enormous majority of the pictures produced in nowadays is lossy compressed by means of the JPEG standard. This is especially true for the photos captured using digital cameras and scanners, which produce the JPEG compressed images directly. Our problem therefore actually is how to store and manage the huge volume of JPEG photos.

Previous work in image browsing, searching, and management has concentrated on solving retrieval and clustering problems. And most of them is devoted to the pixel- or spatial-domain manipulation. This means all images must be decoded totally to the pixel domain before performing any image processing and similarity tests.

In this paper we have investigated the use of the energy histograms of the low frequency DCT coefficients in frequency domain for digital photo analysis. Experimental results prove that our photo album sorting and clustering algorithms can rearrange the randomly placed miscellaneous photos in terms of image similarity and group them into different clusters, which correspond to different daily events and sceneries. Finally, we compress all the digital photos in current album to obtain a compact storage format. Our photo album compression method outperforms the traditional JPEG and MPEG in high compression ratio when there are high similarity and redundancy information among the same cluster.

8. REFERENCES

- [1] S. Chang. Compressed domain techniques for image/video indexing and manipulation. *IEEE International Conference on Image Processing*, pages 314–317, 1995.
- [2] S. Climer and S. Bhatia. Image database indexing using JPEG coefficients. *Pattern Recognition*, 35(11):2479–2488, 2002.
- [3] G. Feng and J. Jiang. JPEG compressed image retrieval via statistical features. *Pattern Recognition*, 36(4):977–985, 2003.
- [4] M. Hatzigorgaki and A. N. Skodras. Compressed domain image retrieval: A comparative study of similarity metrics. *Visual Communications and Image Processing 2003. Edited by Ebrahimi, Touradj; Sikora, Thomas. Proceedings of the SPIE*, 5150:439–448, 2003.
- [5] J. H. Lim, Q. Tian, and P. Mulhem. Home photo content modeling for personalized event-based retrieval. *IEEE MultiMedia*, 10(4):28–37, Oct. 2003.
- [6] A. C. Loui and A. Savakis. Automated event clustering and quality screening of consumer pictures for digital albuming. *IEEE Transactions on MultiMedia*, 5(3):390–402, Sept. 2003.
- [7] W. Pennebaker and J. Mitchell. *JPEG Still Image Data Compression Standard*. van Nostrand Reinhold, New York, 1993.
- [8] M. Shneier and M. Abdel-Mottaleb. Exploiting the JPEG compression scheme for image retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(8):849–853, 1996.
- [9] B. Smith and L. Rowe. Algorithms for manipulating compressed images. *IEEE Computer Graphics and Applications*, 13(5):34–42, 1993.
- [10] M. J. Swain and D. H. Ballard. Color indexing. *International Journal of Computer Vision*, 7(1):11–32, June 1991.
- [11] G. Wallace. The JPEG still picture compression standard. *Communications of the ACM*, 34(4):30–44, 1991.
- [12] C. H. Yeh and C. J. Kuo. Iteration-free clustering algorithm for nonstationary image database. *IEEE Transactions on MultiMedia*, 5(2):223–236, June 2003.

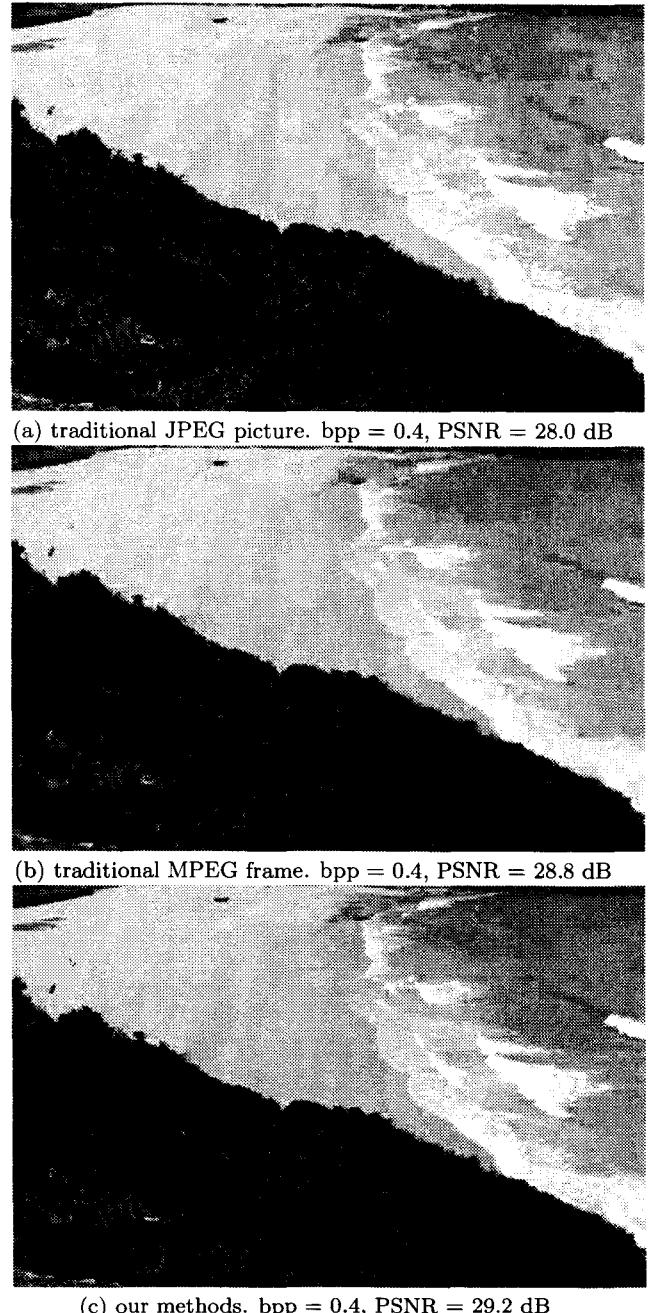


Figure 13: Compression performance comparison

Multiple Embedding Using Robust Watermarks for Wireless Medical Images

Dominic Osborne and Derek Abbott*

Centre for Biomedical Engineering (CBME)
The University of Adelaide, SA 5005, Australia

Matthew Sorell and Derek Rogers

School of Electrical and Electronic Engineering
The University of Adelaide, SA 5005, Australia

Abstract

Within the expanding paradigm of medical imaging and wireless communications there is increasing demand for transmitting diagnostic medical imagery over error-prone wireless communication channels such as those encountered in cellular phone technology. Medical images must be compressed with minimal file size to minimize transmission time and robustly coded to withstand these wireless environments. It has been reinforced through extensive research that the most crucial regions of medical images must not be degraded and compressed by a lossless or near lossless algorithm. This type of area is called the Region of Interest (ROI). Conversely, the Region of Backgrounds (ROB) may be compressed with some loss of information to achieve a higher compression level. This type of hybrid coding scheme is most useful for wireless communication where the ‘bit-budget’ is devoted to the ROI. This paper also develops a way for this system to operate externally to the Joint Picture Experts Group (JPEG) still image compression standard without the use of hybrid coding. A multiple watermarking technique is developed to verify the integrity of the ROI after transmission and in the situation where there may be incidental degradation that is hard to perceive or unexpected levels of compression that may degrade ROI content beyond an acceptable level. The most useful contribution in this work is assurance of ROI image content integrity after image files are subject to incidental degradation in these environments. This is made possible with extraction of DCT signature coefficients from the ROI and embedding multiply in the ROB. Strong focus is placed on the robustness to JPEG compression and the mobile channel as well as minimizing the image file size while maintaining its integrity with the use of semi-fragile, robust watermarking.

Keywords: Semi-fragile Watermarking, Authentication, Medical Images.

1 Introduction

Increasingly medical images are acquired, stored and transmitted digitally. This is especially the case for digital images that are used in radiology [Osborne et al. 2002]. As these types of images are typically of large size, compression allows for the cost of storage to be reduced and the speed of transmission to be increased. Although the cost of transmission bandwidth is decreasing there remains a need of authentication for these types of images and provision for compression so that feasible transmission is possible. The use of a simple system that provides some compatibility with current image compression standards is essential as complex compression schemes are expensive to develop and deploy [Clunie 2000] and allow for very limited usage. This paper presents a technique that can be used to verify the integrity of medical images prior to any diagnosis that is made after transmission over a wireless link through the use of semi-fragile watermarking, which is robust to JPEG [Wallace 1991] compression. This has provided a level of assurance that important detail is present and has not been lost as a result of incidental degradation over a noisy channel. The type

of image that will be used to highlight this type of detail includes a spiral hairline fracture, which is a classic example of this type of diagnostic detailed information. A special subset of authentication watermarking is implemented around the ROI into the ROB to provide authentication of these types of images. This can be extended to any image with a critically important region. As this type of watermarking specifically survives JPEG compression, transmission of a small image file is possible without sacrificing image quality in the ROI where no watermark is placed. This scheme can be applied in one of two ways illustrated in Fig. 1. The first operates externally to the JPEG standard and allows for the entire image to be compressed to specified level. This results in good bit rate performance consistent with typical levels of JPEG compression at the extent of some degradation of the ROI. The other scheme operates by embedding a watermark in an identical way, but encoding the DCT coefficients directly into a hybrid-coded JPEG-like file. This technique results in minimal degradation and near-lossless encoding of the ROI at the expense of a larger file size and slightly inferior bit-rate performance. This must function within the framework of the JPEG standard rather than operating as an external system.

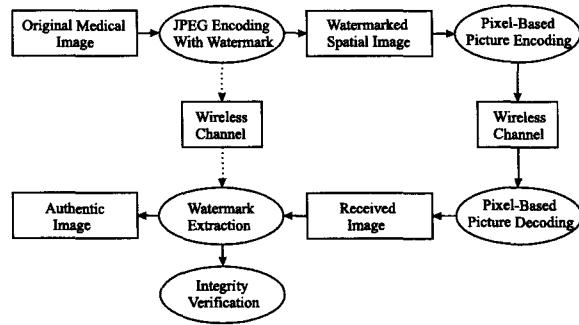


Figure 1: Two possible wireless image scenarios. The first transmission involves dual encoding of the ROI and ROB resulting in a larger file size and improved ROI quality, while the second transmission involves the complete lossy JPEG encoding of the entire image or any near-lossless pixel encoding methodology, such as GIF or JPEG-2000.

*e-mail: dosborne@eleceng.adelaide.edu.au

2 Problem Statement

2.1 The Problem of Authentication

Adding small amounts of noise to corrupt the bitstream of an image file that has been channel-coded does not usually affect the importance of the diagnostic features present in the image after transmission has taken place. Incidental distortions that are not corrected through channel decoding [Viterbi 1967] may slightly distort the file structure of the compressed image file without any noticeable change to perceptual quality. This could involve a loss of diagnostic feature information, which for medical images is detrimental as detailed density information is mandatory. Hence it is critical to authenticate image quality prior to any diagnosis that is made [Osborne et al. 2003]. A classic example of this type of feature information is shown in the Infant's Fracture of Fig. 2.

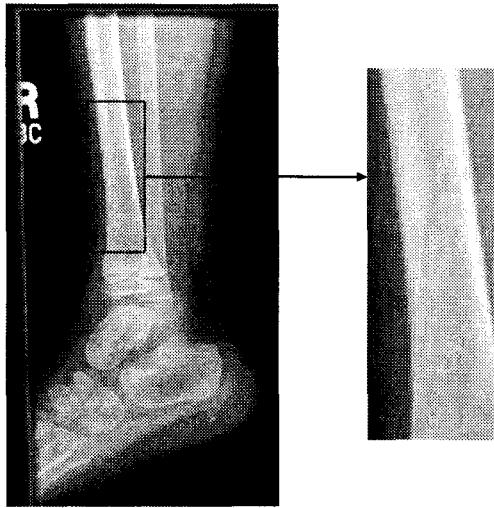


Figure 2: Non-displaced hairline fracture from the leg of an infant, which is often invisible on initial radiographs. If this type of image was transmitted from a hospital to a mobile hand-held device there would be an immediate need to evaluate image quality as a result of the possible loss of image transform coefficients in the ROI or unexpected levels of compression that might degrade feature content in this region.

2.2 The Problem of Large File Size

To maintain as much detail as possible, digital medical images are typically stored without loss of information using lossless compression schemes which allows for complete image restoration. The long term digital storage or mobile transmission of such images is prohibitive without the use of lossy image compression to reduce the image file sizes. As a typical example a mammogram may be digitized at 2048×2048 pixels at 16 Bits Per Pixel (bpp), leading to a file which is over 15 megabytes in size [Strom and Cosman 1997]. The use of lossless formats is widely accepted because no image information is discarded and data is interchangeable from one format to another. This simply leads to a different representation of the image file, but guarantees consistent visual appearance and diagnostic quality of the image. For widespread usage, lossy compression involves the use of JPEG standards. The most common of these which is implemented in most hardware is the well

established Baseline JPEG. This involves performing the Discrete Cosine Transform (DCT) [Ahmed et al. 1974] on 8×8 image pixels to create micro-blocks, quantization of these coefficients and entropy coding of the result. Contrary to excellent developments in lossy image compression, these types of schemes are viewed with suspicion by many members of the scientific and medical community who believe that image alteration may lead to loss of diagnostic or scientific value.

3 Previous Work in ROI Watermarking

The concept of ROI watermarking was first proposed by [Wakatani 2002] who placed signature information into the ROB. A progressively compressed version of a signature image is used and the most significant information is embedded into the region closest to the ROI. This method allows for the signature image to be detected with moderate quality from a clipped version of the image that included the ROI. This system was intended for use over web-based medical image database systems with primary focus placed on ensuring copyright and intellectual property protection. The ROI area in the original image is specified prior to compressing the signature image using a progressive encoding algorithm to generate a bitstream. This allows for increasing visual detail with as the extracted bitstream is followed. The payload is embedded into pixels around the ROI in a spiral way as depicted in Fig. 3. Another re-

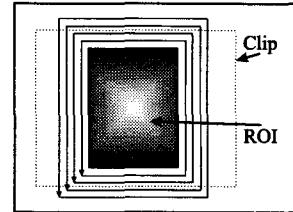


Figure 3: Since the ROI is the most critical aspect of a medical image, it may be clipped to include the ROI. The signature image is compressed using Embedded Zero-tree Wavelet (EZW) coding so that the whole image can be reproduced with average quality and the entire signature image can be retrieved. The quality of the resulting signature image is directly correlated to the length of the bitstream extracted.

cent ROI watermarking scheme is proposed by [Lie et al. 2003], which is designed to operate within the framework of the JPEG-2000 standard, targeting ROI compressed images. A dual watermarking scheme is proposed in which critical image content is to be authenticated. Two different types of watermarks are used, one being naturally fragile and the other robust. The embedding process for the robust watermark takes place at differing resolution layers to ensure that malicious changes are detected and provides flexibility in determining the extent of alteration to discriminate intended attacks from unintended ones. In order to accurately detect which areas have been altered, the first watermark W_1 , which is sensitive and fragile is hidden in the ROI. The second watermark, W_2 is composed of features of mid-frequency wavelet sub-bands and is robustly watermarked into the ROB using features from the ROI as the signature shown in Fig. 4. The robust watermark proposed is designed to survive after acceptable levels of low-pass filtering and JPEG-2000 compression and not to survive malicious attacks. This signature is based on wavelet coefficient properties of the ROI, where features are extracted based on absolute differences between corresponding coefficients in the $LH3$ and $HH3$ subbands on 8×8

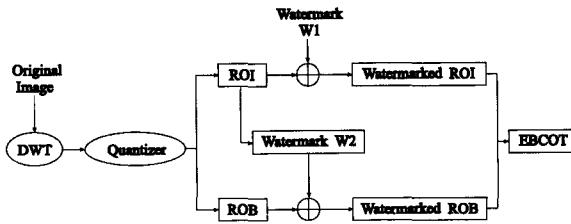


Figure 4: Block diagram of proposed dual watermarking scheme using ROI features as part a watermark to be robustly embedded in the ROB. This is designed to withstand typical levels of JPEG-2000 compression. To be consistent with this standard, scalable image coding is used.

blocks. Similarly in this work a signature is based on the absolute differences between corresponding coefficients in adjacent DCT tiles from inside the ROI, which have been uncorrelated as part of the signature extraction process. [Lie et al. 2003] mentions that his procedure degrades the ROB significantly, however this is not a primary concern as the ROB area is typically encoded at a low quality and gains minimal attention from users. The main focus of the works by [Lie et al. 2003] and [Wakatani 2002] is copyright protection and assurance that malicious attacks on the embedded watermarks are prevented. Our focus is primarily concerned with integrity verification as images are to be transmitted in error-prone lossy transmission channels, such as those encountered in mobile phone telephony to those in Wireless Local Area Networks (LANs.) The most useful contribution in our work is assurance of ROI image content integrity after image files are subject to incidental degradation in these environments. This is made possible with extraction of DCT signature coefficients from the ROI and embedding multiply in the ROB.

4 Authentication Watermarking Technique Used

If the signature information is lumped and localized within the ROB it is possible to authenticate and verify the diagnostic integrity of such images. A simple method to multiply watermark involves embedding in the same shape of the ROI in the eight regions surrounding the ROI or fewer regions if the space in the ROB is unavailable. A visual impression of this method is shown in Fig. 5. Multiple embedding can give the receiver additional confidence in the unlikely event that both a watermark and signature are corrupted in an identical way and the watermark is falsely detected as authentic. It may also be of benefit if one watermark is corrupted. Semi-fragile (or robust) watermarking is specifically designed to withstand application specific transformation operations, such as lossy compression and geometric distortions [Lin and Chang 2001], but is designed to be corrupted as a result of undesirable alterations such as malicious manipulations and incidental degradation over a mobile link which may or may not be perceptible to the receiver. Semi-fragile robust signature embedding ensures that the watermark survives JPEG compression or slight degradation up to a point where the value of the work is lost. Because ROI compression has been successfully subjectively evaluated in ROB of diagnostic medical images [Anastassopoulos and Skodras 2002], the radiologist can have greater confidence that the diagnostic value of the image has not been degraded.

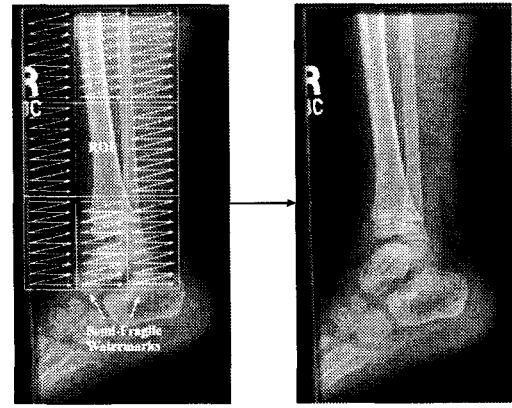


Figure 5: Multiple embedding in the ROB: The algorithm embeds a signature in the eight regions surrounding the ROI or in fewer regions if space is unavailable. Watermarking takes place following the direction of the arrows.

4.1 Basis of Signature Extraction

The basis of singular semi-fragile watermark extraction and embedding was initially developed by [Lin and Chang 2001]. Standard lossy image compression systems involve converting an image into some transform domain, such as wavelet or block DCT domain and quantizing the coefficients in order to reduce their entropy. Coefficients are quantized to a level proportional to how easy it is to perceive changes in them and the property of quantization of coefficients is exploited to remove redundancy in the image. Let $x \bullet q$ be the result of quantizing x to an integral multiple of a quantization step size, q :

$$x \bullet q = q \left[\frac{x}{q} + 0.5 \right]. \quad (1)$$

Consider s to be a real valued scalar quantity and q_1 and q_2 as quantization step sizes with $q_2 \leq q_1$, then

$$((s \bullet q_1) \bullet q_2) \bullet q_1 = s \bullet q_1. \quad (2)$$

If s is quantized to an even multiple of the larger step q_1 and then by a smaller step q_2 , the effect of the second quantization can be reversed. The watermark should survive as long as the quantization that is performed during compression uses smaller step sizes. The watermark embedding and extraction procedure is designed to survive typical levels of JPEG compression, where images are quantized in the block DCT domain. The quantization step size for each coefficient depends on its frequency. These step sizes are obtained by multiplying the transform coefficients by a predefined quantization table, which is scaled by a constant. A signature is extracted from the low frequency terms of the micro-blocks of the ROI and embedded into the high frequency terms of the ROB as a semi-fragile watermark, which is illustrated on a block level in Fig. 6. This is important as the low-frequency terms represent the most important picture information that cannot be degraded through incidental degradation. High frequency coefficients can be used for the embedding in the peripheral regions, as these areas are diagnostically less important and will be degraded through compression. A signature for the image is extracted by converting the medical image into its 8×8 block DCT representation and grouping blocks of the image into pseudo-random pairs according to a specified seed. For each pair of DCT blocks, 8 corresponding low frequency coefficients are compared to obtain 8 bits of the binary signature. Con-

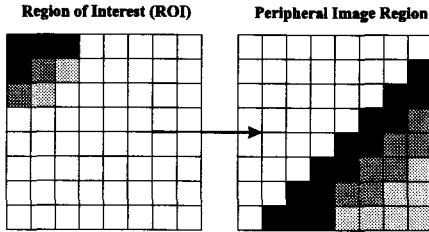


Figure 6: Spatial representation of the bits used for the signature (left) from the ROI and the bits used for the watermark (right) corresponding to the peripheral regions. Matching shades indicate where the payload bits go.

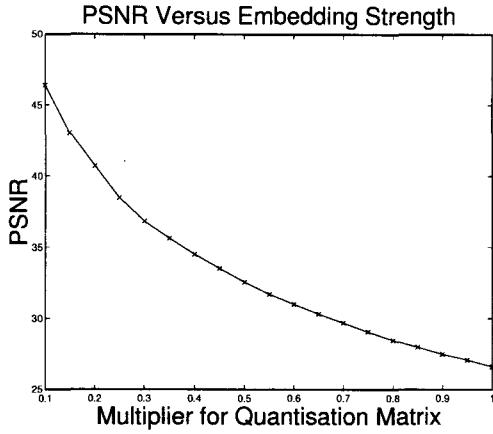


Figure 7: Deterioration of image quality with increasing embedding strength with resulting distortion quantified with the Peak Signal to Noise Ratio (PSNR). This is to be expected and is unavoidable as robust watermarking is used, which is nearly always perceptible to the observer.

sider two blocks that have been grouped C_a and C_b , then:

$$\text{signature bit} = \begin{cases} 0 & : C_a[i, j] < C_b[i, j] \\ 1 & : C_a[i, j] \geq C_b[i, j], \end{cases}$$

where i and j are the coordinates of a low frequency coefficient from Fig. 6. So that the reader can have some perspective on the extent of image degradation resulting from varying embedding strengths of watermarking. A hundred randomly selected grayscale images were tested for their Peak Signal to Noise Ratio (PSNR) after using embedding took place, as seen in Fig. 7. The greater the embedding strength, the more compression the image can survive and the more perceptible the watermark will be. This is not a problem as removal of the watermark can be performed easily at the receiving end.

5 System Implementation of Dual Encoding Scheme

The main sub-systems used in the systematic design of the ROI semi-fragile watermarking scheme designed to work within the framework of JPEG are illustrated in Fig. 8. The image undergoes a block-based DCT specified by a tile or block size, which is typically 8×8 pixels. These coefficients are rounded and quantized and

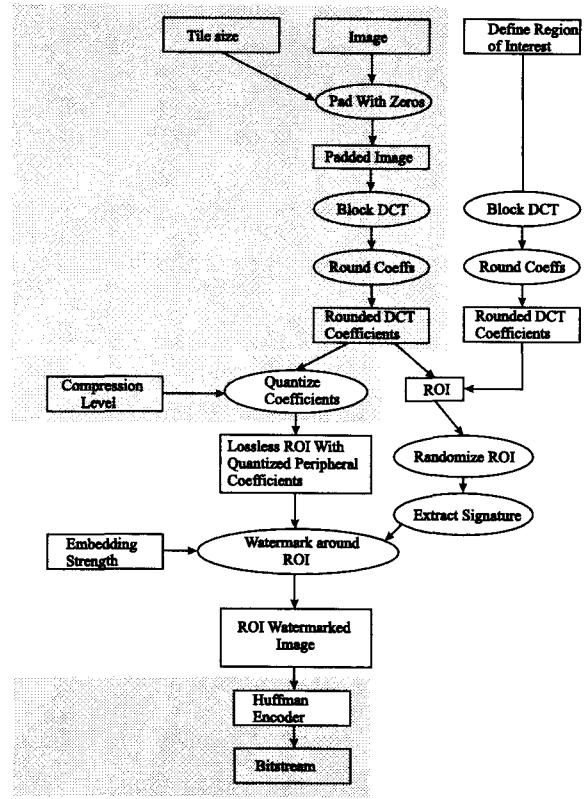


Figure 8: Dual encoding scheme designed to work within the framework of the JPEG standard. A ROI is specified and copied from an image that then undergoes a block-based DCT and quantization to minimize the number of non-zero coefficients for the purposes of high compression resulting in improved bit-rate performance. The compression level is specified by the user as a quantization table multiplier. Sub-systems standard to JPEG are shaded in grey.

entropy [Huffman 1962] encoded. Those areas not shaded in grey include operations within the framework of the standard that can be used for more accurate ROI integrity verification than the system that operates externally to JPEG. The ROI in its transform representation replaces the same region in the full image whose coefficients have been quantized. This ensures that the ROI is stored near-losslessly while the ROB is compressed using lossy JPEG compression and contains at least one watermark.

6 Performance through robustness testing

Survival of JPEG compression is one of the primary requirements of the ROI watermarking scheme. This is mandatory if operation external to JPEG is required, where the pixel-based image compression method is treated as part of the communication channel as shown in the flow diagram of Fig. 1, ignoring the wireless channel on the left side of this diagram. The watermarked image is permitted to undergo types of lossless compression, which will not degrade the image pixels or lossy JPEG, which can be applied up to a threshold specified by the user by the embedding strength. Robustness to varying levels of JPEG compression took place on 100 grayscale images of arbitrary types and varying resolutions from

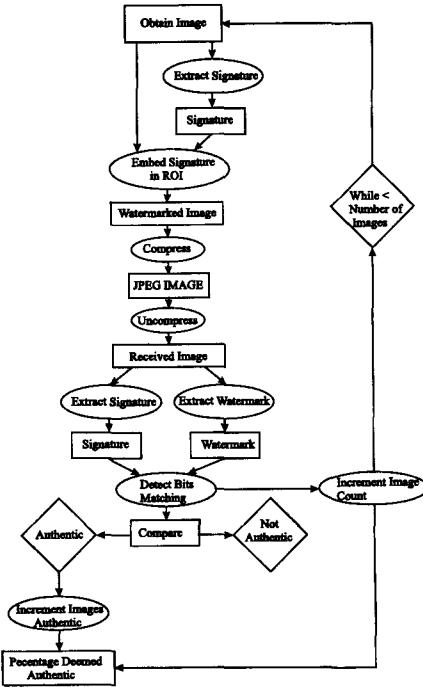


Figure 9: System to test the robustness of ROI watermarking system. The signature extracted from the ROI is embedded within the transform domain of the image. This is subjected to varying degrees of JPEG compression and converted back to a spatial representation. The signature and the watermarks are extracted and compared to authenticate the image. To accommodate minimal bit errors resulting from conversion from transform to spatial representation a 5 percent error is permissible on comparison of extracted signature with extracted watermark for a match.

256×256 to 1280×1280 pixels. The ROI was specified to occupy a sufficiently small area at the center of each image so that 8 watermarks could be embedded around this region. Results are shown in Fig. 9. The results of this test shown in Fig. 10 demonstrate that the ROI watermarking scheme survives JPEG compression levels up to and exceeding the watermark embedding strength used on 90% of the images. This is shown to be consistent for three typical JPEG compression levels. These results are almost identical to those obtained by [Cox et al. 2002] where the performance of a similar watermarking method was tested where a signature was extracted from an image and a singular watermark embedded in the same region. As the scheme developed involves embedding a signature into the same coefficients in 8×8 DCT transform blocks, it is expected to survive similar levels of compression resulting in correlating sets of results. Approximately 10% of the images do not survive JPEG compression for quantisation levels exceeding the embedding strength. This problem can easily be rectified by setting the embedding strength slightly above the level of required JPEG compression.

7 Bit-rate Performance of Hybrid Coding Scheme

If the image is sufficiently robustly watermarked and converted to spatial domain for complete JPEG compression, the resulting file

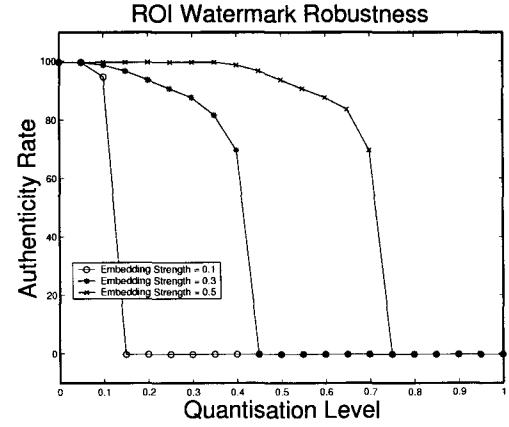


Figure 10: Testing the robustness of 100 images with the semi-fragile ROI watermarking scheme designed to withstand JPEG compression. As expected, the system fails the authentication test consistently after each of the three watermark embedding levels.

size is directly related to the level of compression. This results in a bit-rate performance that is identical to JPEG. If degradation of the ROI through JPEG quantization is not permissible and hybrid coding is preferred as illustrated in the flow diagram of Fig. 8 the bulk of the ‘bit budget’ will be stored in the ROI. This is because quantisation does not take place in this region and all ROI transform coefficients must be encoded, which are typically non-zero. As the ROI can undergo compression through quantisation, the majority of coefficients will be zero. This will result in a file size that is dependent on the size of the selected ROI. The larger this region is, the more near-lossless compression is required, the larger the file size. For a typical fracture or tumor, the area of the ROI does not typically extend beyond 20% of the entire image. This is also verified in work by [Foos et al. 2000] and [Anastassopoulos and Skodras 2002] where ROI Maxshift JPEG-2000 compression was utilized to compress these types of medical images. Strom [Strom and Cosman 1997] also validated the effectiveness of combined lossy and lossless JPEG compression with these types of ROI sizes. It was shown through extensive subjective testing that the diagnostic value of the medical image did not degrade for very low bit rate coding. These approaches reinforce that the ROI is exactly the area where all diagnostic information is located. Bit-rate performance was evaluated in Fig. 11 with and without the use of watermarking and with sizes of ROI varying from complete lossy compression, where the peripheral regions were the entire image to the extreme of having the entire image encoded near-losslessly as a ROI. The most practically applicable areas of these curves includes those areas up to and around having a 20 percent area devoted to the ROI. Within this area of the curve the use of one or more embedding regions does not significantly affect the size of the medical image file. It would appear that each embedding region in Fig. 5 increases the file size by approximately 0.1 bpp, which without watermarking results in a compression level of 2 bpp. The increase in file size is insignificant in comparison with complete near-lossless JPEG encoding that provides minimal compression of 5 bpp of the original grayscale image. After baseline JPEG compression, which corresponds to using a quantization multiplier of 1, it is typical for the resulting quantized 8×8 DCT coefficient ROI blocks to be reduced from 64 to an average of 2 coefficients. This confirms the majority of the bit-budget is devoted to the ROI, which makes the compression scheme useful for low bandwidth mobile communication. This does assume that the ROI chosen is relatively small in comparison

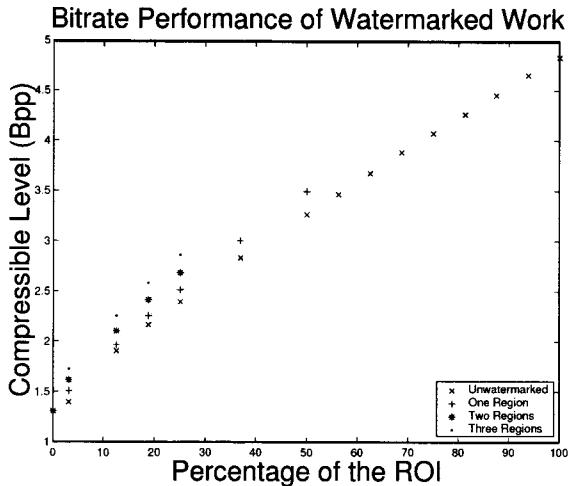


Figure 11: Bit-rate performance which can be compared with entire lossy JPEG compression, where the area of the ROI is zero to entire lossless compression in which the percentage of the ROI is 100.

to the rest of the image.

8 Conclusion

The use of watermarking can be used to verify the integrity of digital images. A method is developed which is designed to be used on images that have critically important regions. The scheme is designed to perform multiple watermarking around the ROI containing features from this region. This technique can provide assurance that this region has not been corrupted as a result of incidental degradation resulting from a wireless link or that the ROI has not exceeded a compression level threshold. As watermarked images can be compressed, a smaller file size is achievable. This facilitates verification of ROI integrity as well as wireless communication. The method used allows the user to evaluate quality of this region in a received image without the need of a reference image. This is most useful for transmitted medical images where high levels of quality assurance are mandatory prior to making any diagnosis. The technique used can be systematically designed in two ways, one which fits in a modular way into the JPEG standard resulting in minimal changes and hybirdly coding the ROI and the ROB resulting in superior ROI quality. Alternatively this method can operate extraneously to the standard providing greater compliance and improved bit-rate performance. This results in a degraded ROI when lossy JPEG is used on the watermarked image pixels or improved picture quality if lossless picture encoding techniques are used. This method could also be used to monitor image or video quality for quality control systems or benchmarking image/video processing systems and algorithms. A limitation is that authentication based on watermarking cannot replace classical cryptographic authentication protocols that protect communication channels. Embedded robust watermarking for ROI integrity verification can allow for compression and provision of image integrity. This can be most useful for medical images that must be transmitted quickly in a wireless environment.

References

- AGRAFIOTIS, D., BULL, D. R., AND CANAGARAJAH, N. 2003. Region of interest coding of volumetric medical images. In *IEEE International Conference on Image Processing (ICIP)*, vol. 2, 217–220.
- AHMED, N., NATARAJAN, T., AND RAO, K. R. 1974. Discrete cosine transform. *IEEE Transactions on Computer Theory C-23* (January), 90–94.
- ANASTASSOPOULOS, G. K., AND SKODRAS, A. 2002. JPEG2000 ROI coding in medical imaging applications. In *Proceedings of the 2nd IASTED International Conference on visualisation, imaging and image processing (VIIP2002)*.
- CLUNIE, D., 2000. Lossless compression of greyscale medical images - effectiveness of traditional and state of the art approaches.
- COX, I. J., MILLER, M. L., AND BLOOM, J. A. 2002. *Digital Watermarking*, 1 ed. The Morgan Kaufmann Series in Multimedia Information and Systems. Morgan Kauffman Publishers, 340 Pine St, Sixth Floor, San Fransisco, CA 94104-3205, USA.
- FOOS, D. H., MUKA, E., SLONE, R. M., ERIKSON, B. J., FLYNN, M. J., CLUNIE, D. A., HILDEBRAND, L., KOHM, K., AND YOUNG, S. 2000. JPEG2000 compression of medical imagery. 85–96.
- HUFFMAN, D. A. 1962. A method for the construction of minimum redundancy codes. *Proceedings of the IEEE 40*, 1098–1101.
- LIE, W.-N., HSU, T.-L., AND LIN, G.-S. 2003. Verification of image content integrity by using dual watermarking on wavelets domain. In *IEEE International Conference on Image Processing (ICIP)*, vol. 2, 487–490.
- LIN, C. Y., AND CHANG, S. F. 2001. A robust image authentication method distinguishing JPEG compression from malicious manipulations. *IEEE Transactions on Circuits and Systems of Video Technology 11*, 2, 153–168.
- OSBORNE, D., ABBOTT, D., COUTTS, R., AND ROGERS, D. 2002. An overview of wavelets for image processing for wireless communications. In *SPIE Smart Structures, Devices and Systems*, vol. 4935, 427–435.
- OSBORNE, D., ABBOTT, D., SORRELL, M., AND ROGERS, D. 2003. Embedded importance watermarking for image verification in radiology. In *SPIE BioMEMS and Nanotechnology*, vol. 5275, 383–390.
- PENEDO, M., PERAMAN, W. A., TAHOCES, P. G., SOUTO, M., AND VIDAL, J. J. 2003. Embedded wavelet region-based coding methods applied to digital mammography. In *IEEE International Conference on Image Processing (ICIP)*, vol. 2, 197–200.
- STROM, J., AND COSMAN, P. 1997. Medical image compression with lossless regions of interest. *Signal Processing 3*, 155–171.
- VITERBI, A. J. 1967. Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory IT-13* (April), 260–269.
- WAKATANI, A. 2002. Digital watermarking for ROI medical images by using compressed signature image. In *Proceedings of the 35th Hawaii International Conference on System Sciences. IEEE Computer Society*, vol. 6, 157–163.
- WALLACE, G. K. 1991. The JPEG still picture compression standard. *Communications of the ACM 34*, 4 (April), 30–45.

A Mediation Framework for Multimedia delivery

Onyeka Ezenwoye, Raimund K. Ege, Li Yang, Qasem Kharma

School of Computer Science
Florida International University
Miami, FL 33199

Telephone: +01 – (305) – 348 -1038

{oezen001, ege, lyang03, qkhar002}@cs.fiu.edu

ABSTRACT

We present a conceptual mediation framework that features three layers of mediators: *presence*, *integration*, and *homogenization* layers that work together in a peer-to-peer (p2p) manner to facilitate the delivery of multimedia data. On arrival of each request for data from a clientⁱ, a *global-mediator* is elected from a group of *integration* layer mediators to service that request. Using distributed hash table (DHT), the *global-mediator* dispatches the request to other integrator mediators to track down the data sources. Upon receipt of the results, from the source(s), the *global-mediator* presents the data to the client via a *presence-mediator*. The *presence-mediator* may need to reformat the data to suit the execution context of the client. This mediation process is context-aware, adaptive and dynamically structured. Quality of service (QoS) factors are taken into consideration in the retrieval and presentation of data.

Keywords

Mediator, middleware, heterogeneous data sources, multimedia delivery.

1. INTRODUCTION

The proliferation of the internet has enabled access, at least on a physical level, to a multitude of disparate but often related information, while scaling geographical barriers. This information, in the form of multimedia data is stored on and accessed from various kinds of heterogeneous devices, recently more of which are mobile. Multimedia data requires special attention to throughput, timeliness and other quality of service factors. There is a need for architectures to deal with buffering and the intermittent connection associated with mobility. Our approach to enabling high quality access is to build a layered framework of mediators [18]. Lower-layer mediators connect to the actual data sources, while higher-layer mediators provide a logical schema of information to applications.

Mediators are typically employed in a situation where the client data model does not coincide with the data model of the potential data sources. They are facilitators that search for likely resources and ways to access them [17]. They provide a mapping of complex models to enable interoperability between client and source(s). Although many mediator systems have been proposed for a variety of applications, a major problem often encountered is how to seamlessly query and integrate data

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

MUM 2004, October, 27-29, 2004 College Park, Maryland, USA.
Copyright 2004 ACM 1-58113-981-0 /04/10... \$5.00

from heterogeneous data sources. Hence there is a need to formulate a mediator language that provides support for complex and semi-structured data types; a language that allows communication of knowledge between the mediator and source as well as the mediator and the client [3].

To overcome the problems posed by heterogeneity of data sources, the language of choice for our system is XML. XML is clearly today's standard of choice for the representation and exchange of structured data, particularly where that data must be read and interpreted by different applications running of different kinds of devices. XML and XML Schema provide a convenient, potentially human readable, easily extensible representation standard. Therefore, all data exchanged between mediators would be as XML.

In this paper we describe a three-layer architecture for multimedia mediation. The paper is organized as follows: Section 2 presents some related work and briefly covers some differences and similarities between our architecture and existing ones. Section 3 describes each layer and what functions are performed therein. We also discuss the different classes of mediators in those layers. Section 4 covers the election of global mediators to handle specific queries and a brief overview of the proposed election algorithm. Section 5 explains the various classifications of Distributed Hash Table algorithms and their use in looking up peers in p2p networks.

2. RELATED WORK

A lot of work has been done on mediation systems [4, 16, 19, 12, 9, 8, 7]. As stated in [9, 7], most of these architectures however are centralized, in that, there is a single mediator through which query decomposition, result integration and access to heterogeneous sources is achieved. Like our architecture, some [9, 16, 19] mediator architectures are distributed and mediators are able to access and communicate with each other. [19] is a two-tier mediation model that comprises a *homogenization* and *integration* layer with mediators in each that playing similar roles as in our architecture. [9] on the other hand does not have any restrictions on mediator functions as each mediator can play the role of homogenization and/or integration. There is also no

ⁱ The use of the word client does not necessarily mean desktop PC. It could be any device with a digital heartbeat, mobile or immobile.

restriction as to the number of mediator tiers. [9] and [19] employ a similar integration process for homogenized sources [9]. Our architecture is a three layer model that consists of the *presence*, *integration* and *homogenization* layers. Our architecture does not only accommodate heterogeneous data sources but also with the aid of the *presence* layer mediators adapts to the heterogeneous nature of the client devices by taking into account various QoS issues of the client. [9] is a peer mediation system much like ours but unlike our model, it does not employ the use of the DHT in the distribution of source schema and peer lookup.

3. THREE LAYER ARCHITECTURE

The proposed three-layer mediation architecture is to handle requests (query or update) from a client which can be any special device or mobile computing unit.

The framework features three layers; *Presence*, *Integration* and *Homogenization*. A different class of mediator will be implemented within each layer (see Figure 1). A device may have all the three classes of mediators running on it at the same time. The mediators will transfer and negotiate on three kinds of information; the schema of the data stream, the type of operation required (e.g. query or update) and some quality of service (QoS) information specific to the client. The reason for exchanging QoS information is so that data streams can be tailored at the appropriate layers to suit the execution context of the client device.

3.1 Presence Layer

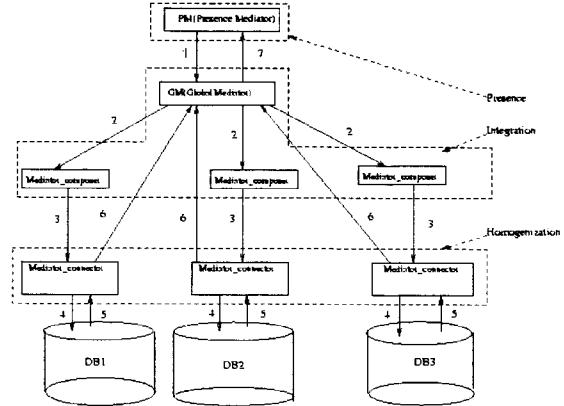
The primary functions performed in this layer are:

1. Attach required QoS parameters to queries.
2. Election of global mediators to handle the requests.
3. Continuously advertise changes in QoS parameters to global mediators.

At the inception of a user request, the system would create a *presence-mediator* to handle that request. So there is one *presence-mediator* for each request, a *presence-mediator* cannot handle more than one request and the lifespan of an instance of a *presence-mediator* is dictated by the duration for which the request is valid. Upon receipt of a request, the *presence-mediator* would conduct an election to elect a *global-mediator* to serve that request.

A *presence-mediator* serves as a go-between for the client device and the *global-mediator* for that request, continuously monitoring the status of the device and for changes in its QoS parameters. The request's QoS specification is a translation of the perceived execution context on the client application.

QoS management is essential to efficiently access pertinent information at the required level of quality. This function attempts to meet the level of quality required by user.



1: query and/or client QoS information

2,3,4: query

5,6,7: query result

Figure 1. Three-Layer Architecture.

The continuous nature of the QoS management is especially important in the event that the client device is mobile. Resources are scarce on mobile devices and the availability of a resource may vary significantly and unpredictably during the runtime of an application. In the absence of resource guarantees applications need to adapt themselves to the prevailing operating conditions.

Presence mediators also have the job of converting the results of the request from XML to a format that is required for the particular client device.

In an ambulatory environment, for instance, a doctor might need to get critical information about a patient. The doctor, armed with a PDA with a wireless link, submits a query (Figure 2) to retrieve the patient's medical records.

```

<xss:schema xmlns:xss = "http://www.w3.org/2001/XMLSchema">
  <xss:element name = "query">
    <xss:complexType>
      <xss:sequence>
        <xss:element name="patient_record" >
          <xss:element name="patient_id" >
            <xss:element name="name" >
              <xss:element name="date_of_birth">
            </xss:sequence>
          </xss:complexType>
        </xss:element>
      </xss:sequence>
    </xss:element>
  </xss:schema>

```

Figure 2. An example schema of a request for patient records.

Upon receipt of the query, the *presence-mediator* modifies the query by attaching the PDA's QoS parameters as illustrated in Figure 3.

```
<xs:schema xmlns:xs =
"http://www.w3.org/2001/XMLSchema">
<xs:element name = "query">
  <xs:complexType>
    <xs:sequence>
      <xs:element name="patient_record" >
        <xs:element name="patient_id" >
          <xs:element name="name" >
            <xs:element name="date_of_birth">
          </xs:sequence>
        </xs:complexType>
    </xs:element>
  <xs:element name = "qos">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="resolution" >
        <xs:element name="color_depth" >
        <xs:element name="bandwidth" >
        <xs:element name="power" >
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Figure 3. Modified schema of the request for patient records with QoS criteria.

3.2 Integration Layer

The mediators that comprise this layer are known as *mediator-composers*. These are the basic building blocks of the system. For a device to be considered a peer, it must implement a *mediator-composer*. It is this *mediator-composer* that upon receipt of a user request creates a *presence-mediator* for that particular request. Thus, the *presence* layer described in 3.1 only exists if there is at least one request being handled.

This layer is vital for the ability of the system to process queries. Because query processors may need to reformulate an initial query to enhance the chance of obtaining relevant data [1], this layer of mediators may need to translate the XML schema of the query into the schemas supported by other mediators. Because the mediators are p2p, each mediator will have specific knowledge about the supported data and schemas of its “neighbor” mediators. In the event that the *global-mediator* (or other *mediator-composers*) has no knowledge about others’ schema, the original schema is forwarded to its known peers (line 2 in figure 1) unaltered.

In other words, *mediator-composers* have the ability to reconstruct XML schemas for requests. When a *mediator-composer* receives a request, it may need to simplify the request before forwarding it. If the *mediator-composer* has some knowledge about the request, it simplifies the request according to its knowledge. The *global-mediator* is informed whenever the schema of its request is altered. The *global-mediator* keeps track of where what is found with the use of a “mediated schema”. This mediated schema will also be used to reassemble the results of the query that were obtained from different sources and further query.

The *integration* layer basically reformulates the client request into a set of queries over the data sources using the appropriate schemas. *Mediator-composers* set up contracts between multiple data sources in order to satisfy requests. They are in charge of finding systems that meet the specified QoS criteria.

3.3 Homogenization Layer

In our architecture, each *mediator-connector* (*homogenization* layer mediator) will be directly associated with a physical source. Note that (as stated in section 3) that a device can have all three types of mediators on it. A *mediator-connector* is implemented only if it is associated with a persistent data source.

Mediator-connectors do not change the XML for the request; they retrieve data from data sources and converts query result into a stream of XML data which is submitted to the *global-mediator* for the request (line 6 in figure 1).

Data from relational databases can be mapped to XML by table-based mapping. The advantage of this mapping is its simplicity: because it matches the structure of tables and result sets in a relational database. This type of mapping however has several disadvantages; primarily, it only works with a very small subset of XML documents. It also does not preserve physical structure (e.g. character and entity references, CDATA sections, character encodings, and standalone declaration) or document information (e.g. document type or DTD), comments, or processing instructions.

Because table-based mappings only works with a limited subset of XML documents, some middleware tools, most XML-enabled relational databases, and most XML-enabled object servers use a more sophisticated mapping technique called object-relational mapping. This models the XML document as a tree of objects that are specific to the data in the document; it then maps these objects to the database.

Most XML schema languages can be mapped to databases with an object-relational mapping. The exact mappings depend on the language.

4. GLOBAL MEDIATOR ELECTION

Once the *presence-mediator* (in the *presence* layer) receives a request for which it was instantiated, it creates an XML schema for this request coupled with some QoS criteria specific to its client device and that’s best suited for that type of request. The *presence-mediator* then conducts an election to select a *mediator-composer* (in the *integration* layer) to be the *global-mediator* for that request. The election is conducted in order to find the best possible *mediator-composer* to serve the request. Selection criteria include but are not limited to available bandwidth, network traffic and load. The elected mediator will be the one that best meets the required QoS criteria and is able to carry out the search, cache, integrate and return the result. In explaining the mediator election, it is important to note that:

A peer knows at least one other peer otherwise the system isn’t p2p.

1. A *mediator-composer* can be elected to serve as *global-mediator* for more than one request at the same time.
2. Each request is serviced by only one *global-mediator*.
3. In the event that a *global-mediator* fails, another one is elected.

As of this writing, our election algorithm of choice is the ring algorithm. Our ring algorithm is based on the ring algorithm [15] with some modifications.

¹ When a mediator-composer is elected to serve a request, it becomes the global-mediator for that request

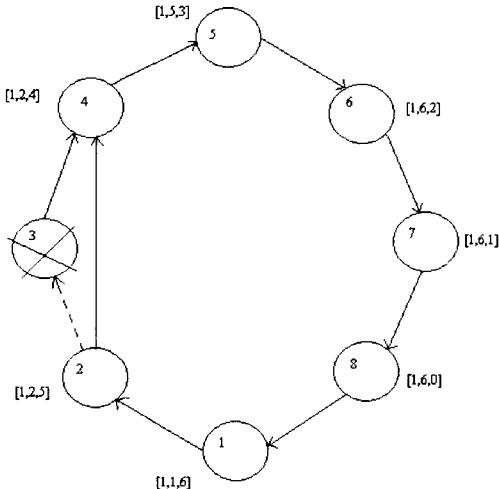


Figure 3: The Ring Election Algorithm.

For the purpose of clarity, we will use Figure 3 above to illustrate the algorithm. It is also important to note that for the purpose of brevity, the following description has been simplified. The election message is a 3-tuple $[i, j, k]$, where i = election initiator; j = best candidate so far; k = hop count. The hop count is used to make sure that the election message does not travel perpetually because a p2p network may not actually be physical ring.

When there is a need to elect a *global-mediator*, the peer that is initiating the election (node 1) sends an election message to its successor. In Figure 3, node 1 sets i and j to 1, thereby making itself a candidate for the election. $k = 6$ for the purpose of illustration. By setting j to 1, we mean that node 1 attaches to the election message its status information. This could be information such as load and bandwidth. It is possible for the initiating node to be elected because (as stated in section 3.2), a *mediator-composer* must reside on the physical device for that device to be considered a peer. It is this *mediator-composer* that creates the *presence-mediators* (section 3.1) to handle requests. A *presence-mediator* - in its quest to find the best suitable peer (*mediator-composer*) to act as *global-mediator* for the query - may end up electing the same *mediator-composer* that created it. Thus electing its own peer.

Upon receipt of the election message, the receiving node compares the status information contained in the election message with its own status information. If it determines that its status is superior to that which is contained, it replaces the status information with its own (e.g. node 2 in figure 3). It checks to see that the $k > 0$ then forwards the message to its successor after reducing the hop count by 1. If its successor is down (e.g. node 3 in Figure 3), the message is sent to the next successor. The election only terminates in these three cases:

1. If a node's only successor is down, the message is sent to i and j is elected.
2. If $k = 0$, the election message is sent to i and j is elected.
3. If a node's successor is i (a ring), j is elected.

In our example in Figure 3, node 6 is elected as *global-mediator*.

5. PEER LOOKUP

After electing the *global-mediator*, the *global-mediator* will coordinate with other *mediator-composer(s)* and/or *mediator-connector(s)*, in order to serve the request. Interaction between mediators is P2P in which peers share distributed files. The most recent lookup algorithms for P2P system are based on distributed hash table (DHT). In general, these algorithms routing time complexity is $O(\log N)$ where N is the number of nodes (peers) in the system. [2] classifies DHT algorithms into three categories:

1. Skiplist-like routing algorithm:

Chord algorithm [14] is an example of skiplist-like routing algorithm. In Chord, every node in the system maintains information about $O(\log N)$. The hash function assigns an m -bit identification key using SHA-1 as a base function to map the IP address. The nodes in the system are arranged in an identifier circle. Each node on this circle maintains a finger table containing the IP addresses of $n+2i-1$ successors where n is the node ID and $1 \leq i \leq m$. In other words, this finger table maintains the IP addresses of halfway, quarter-of-the-way, eighth-of-the-way, and so forth. As a result this algorithm can find the required node in $O(\log N)$ time.

2. Tree-like algorithms:

Tree-like algorithms, such as Pastry [13], Tapestry [6], and Kademlia [10], use structured prefix to maintain the location of nodes. Each node maintains IP addresses of some other nodes in its leaf.

3. Routing in Multiple dimensions:

CAN [11] is an example of routing in multiple dimensions. Each node in CAN maintains chunk of DHT called zone. These zones are distributed in d -dimension. In addition to storing a chunk of DHT in the zone, each zone maintains information about its neighbors in the d -dimension. The routing time complexity for this algorithm is $O(d N^{1/d})$.

The reader can observe that these algorithms are similar in the following aspect:

1. With the use of DHT, each node maintains information about its neighbors only, not all the nodes in the system
2. Their time complicity for routing is $O(\log N)$ for most of them.

These algorithms however differ in many aspects, but the most important difference is how each algorithm determines "neighbor". In general, each node should maintain minimum knowledge about other nodes in the systems. A hash function, i.e. SHA-1, maps keys onto values where values could be file names, IP addresses, or any naming to be looked up. In our case, we are interested in mapping XML schema tags and we will use Cord algorithm [14] because of its performance [5] in comparison to other algorithms.

In order for a new node to join the system and become a peer, it will send a "join" message [14]. After locating the position of this node, the new node will join the P2P system as a composer. This new peer may have an associated *mediator-connector* if there is an associated database.

Despite of the role of *global-mediator* in the presentation, all *mediator-composers* need to cooperate in order to find the connector(s) (*mediator-connectors*) to the desired data source(s).

To find the connectors(s), the route from the *global-mediator* through composers can be found using DHT instead of having a central repository of the connectors' XML schemas. All messages between mediators are in XML format and each composer maintains some XML schema which will be used to decompose/compose the XML request in order to match a XML schema that is stored in a connector. The hash function maps the XML tags or elements onto keys which will be distributed over the peers. Assume the following is a valid XML schema:

```
<?xml version="1.0"?>
<xss:schema xmlns:xss="http://www.w3.org/XMLSchema">
  <xss:element name="PatientXrays">
    <xss:complexType>
      <xss:attribute name = "ssn" type="xss:string">
      <xss:attribute name="fullname" type="xss:string">
      <xss:attribute name="xray" type="xss:image">
    </xss:complexType>
  </xss:element>
</xss:schema>
```

The hash function maps ssn, fullname², and xrays onto keys which will be distributed over the peers (*mediator-composers*). *Mediator-composers* will generate the XML tree for the XML schemas which have been sent from *mediator-connectors* to be mapped. *Mediator-composers* hash the nodes, which correspond to elements in the XML schema in the corresponding tree and distribute the generated key with the element to a node in the system which maintains the range of that key.

The *mediator-composers* decompose the incoming request or simplify the incoming request by adding subtree(s) to the original request until all the tree leaves represent connectors. In decomposing a query, if the *global-mediator* which has been elected to handle this request cannot solve the FullName for instance, it will forward the request to one of its neighbors. Eventually, one of the composers will decompose the FullName into FirstName and LastName. After that, all the leaves in the tree can be directed to the corresponding connector using the DHT.

6. CONCLUSION AND FUTURE WORK

The interchange of data between client and heterogeneous sources requires an efficient and dynamic approach to mediation. The framework described in this paper features three layers of mediators: *presence*, *integration*, and *homogenization*. On arrival of a request for data, a *mediator-composer* is elected as *global-mediator* that is responsible for data caching and service provision. The *global-mediator* dispatches the data stream request to other *mediator-composers* in order to track down the adequate sources. The results are then integrated and sent back to the user in a way that best suits the execution context of the user device.

The advantage of our mediation process is its adaptive and dynamic nature. The framework is designed to uniquely determine how to fulfill each query while taking properties of delivery into consideration. The presence-mediator takes into

account the heterogeneous nature of client devices and is meant to tailor the query formulation and presentation of results to suit the execution context of the client. This is especially important for mobile devices give their limited resources. Our mediation architecture is a work-in-progress and there are many research issues that will be encountered during the course of this project, they include but not limited to, defining of communication protocols with specific focus on QoS, how to deal with real-time data and mobility (e.g. temporary loss of connectivity in mobile devices, failure of the *global-mediator*), security issues involved with the distribution and access of data across a p2p network and how to intelligently decompose and integrate XML schemas while avoiding loss of information.

7. REFERENCES

- [1] Arens, Y., Knoblock, C. and Shen, W. Query Reformation for Dynamic Information Integration. *Journal of Intelligent Information Systems: Integrating Artificial Intelligence and Database Technologies*, 1996;6(2-3):99-130.
- [2] Balakrishnan, B., Frans Kaashoek, M., Karger, D., Morris, R. and Stoica, I. Looking up data in P2P systems, *Communications of the ACM*, volume 46 , Issue 2, February 2003.
- [3] Buneman, P., Raschid, L. and Ullman, J. Mediator Languages – a Proposal for a Standard, *Report of DARPA I3/POB working group*, University of Maryland, 1996.
- [4] Garcia-Molina, H., Papakonstantinou, Y., Quass, D., Rajaraman, A., Sagiv, Y., Ullman, Y. D., Vassalos, V., and Widom, J. The TSIMMIS approach to mediation: Data models and languages. *Journal of Intelligent Information Systems*, 8(2):117 - 132, 1997.
- [5] Gummadi, K., Gummadi, R., Gribbl, S., Ratnasam, S., Shenke, S., and Stoica, I. The Impact of DHT Routing Geometry on Resilience and Proximity. In *Proceedings of SIGCOMM'03*, August 25–29, 2003, Karlsruhe, Germany.
- [6] Hildrum, K., Kubiatowicz, J., Rao, S., and Zhao, B. Distributed Object Location in a Dynamic Network. In *Proceedings of 14th ACM Symposium. on Parallel Algorithms and Architectures (SPAA)*, August 2002.
- [7] Josifovski, V., and Risch, T. Comparison of Amos II with other Integration Projects, Technical Report, EDSLabs/IDA, Linköping University, April 1999
- [8] Karjalainen, M. Integrating Heterogenous Databases with the Functional Data Model Approach, <http://www.cs.chalmers.se/~merjaka/report04d.pdf>, January, 2004
- [9] Katchaounov, T. *Query Processing for Peer Mediator Databases*, Doctoral thesis, Uppsala University, 2003
- [10] Maymounkov, P., and Mazieres, D. Kademia: A peer-to-peer information system based on the XOR metric. In *Proceedings of the 1st International Workshop on Peer-to-Peer Systems*, Springer-Verlag version, Cambridge, MA, Mar. 2002.
- [11] Ratnasamy, S., Francis, P., Handley, M., Karp, R., and Shenker, S. A scalable content-addressable network. In *Proceedings of ACM SIGCOMM*, San Diego, CA, August 2001.

² Some elements in the client XML schema might need to be decomposed; i.e. the fullname could be decomposed into lastname and firstnames, and that way multiple composers may cooperate to serve the client.

- [12] Risch, T., Josifovski, V., and Katchaounov, T. AMOS II Concepts, http://www.dis.uu.se/~udbl/amos/doc/amos_concepts.html, June 23, 2000
- [13] Rowstron, A., and Druschel, P. Pastry, Scalable, distributed object location and routing for large-scale peer-to-peer systems, In *Proceedings of the 18th IFIP/ACM Int'l Conf. on Distributed Systems Platforms*, Heidelberg, Germany, pages 329-350, Nov. 2001.
- [14] Stoica, I., Morris, R., Karger, D., Frans Kaashoek, M., and Balakrishnan, H. Chord: A scalable peer-to-peer lookup service for Internet applications. In *Proceedings of ACM SIGCOMM*, San Diego, August 2001.
- [15] Tanenbaum, A. S., Van Steen, M. *Distributed Systems: Principles and Paradigms*. Pearson Education, September 2001, page 263.
- [16] Tomasic, A., Raschid, L., and Valduriez, P. Scaling access to heterogeneous data sources with DISCO. *IEEE Transactions on Knowledge and Data Engineering*, 10: 808 – 823, 1998.
- [17] Wiederhold, G., and Genesereth, M. The Conceptual Basis for Mediation Services. *IEEE Expert*, Vol.12 No.5, Sep.- Oct. 1997, pages 38-47
- [18] Wiederhold, G., Mediators in the Architecture of Future Information Systems, *IEEE Computer*, 25(3):38–49, Mar. 1992
- [19] Yan L., Tamer Özsü, M., Liu, L., Accessing Heterogeneous Data Through Homogenization and Integration Mediators, In *Proceedings of the Second IFCIS International Conference on Cooperative Information Systems*, pages 130-139, June 24-27, 1997

Task Computing

Zhexuan Song Ryusuke Masuoka Yannis Labrou
Fujitsu Laboratories of America, Inc.
8400 Baltimore Avenue, Suite 302
College Park, Maryland 20740-2496, USA
{zsong, rmasuoka, yannis}@fla.fujitsu.com

Description

Task Computing enables a user to compose and execute complex tasks in application-, device- and service-rich environments. Task Computing fills the gap between what users want to do and the devices and/or services that are available in the current environment. Through the use of a Task Computing Client (TCC), users can construct tasks from available, semantically described services, using basic task computing functions to discover, create, compose, manage and manipulate these services.

The “Task Computing Environment (TCE)” is a framework that supports task computing by providing services and interfaces for its workflows, semantic service descriptions and service management for end-users. In architectural terms, TCE consists of four distinct layers (Figure 1): realization layer, service layer, middleware layer, and presentation layer. The separation of these layers is both logical and implemental. TCE enables the user to perform tasks that have not been (either implicitly or explicitly) designed into the system, thus multiplying the potential uses of the sources of functionality (devices, applications, content and e-services).

TCE establishes and implements the complete separation of semantic service descriptions and service implementations; separation between the discovery mechanisms and discovery ranges as well as the ability to manipulate services within and between those ranges; the ability for users (and services) to dynamically create and manipulate services that can be made available and shared with others (or made unavailable when necessary);

together providing a variety of services that enable interesting and truly useful tasks.

Task Computing adopts the following standards in implementation: RDF [1], OWL [2] and OWL-S [3] for semantic service description, UPnP [4] for service publishing and discovery, and Web service (SOAP and WSDL) for service invocation. Task Computing provides a higher level of interoperability between devices and services, along with its novel user experience.

We have designed the demonstration based on our latest TCE (version 1.0), that consists of five different types of TCC’s, including the latest voice-triggered interface based on a complete set of web services; many Semantically Described Services (SDS’s), including a set of multimedia services; and Semantic Service Publishing and Management (PIPE and White Hole).

Some of the semantically-described multimedia services we have implemented include an audio and video player, a security surveillance video streaming service, a digital photo frame, a plug-and-publish application which allows user to share images, multimedia files immediately with others, and a smart object reader that can read out almost all semantic object instances.

Through the demonstration, we would like to show that Task Computing is a flexible and universal framework. For example, the forwarding of a security video to any of a number of display devices without manually connecting cables can be a complex task when done in traditional ways, if it has not been pre-programmed. In Task Computing, it can be done with a couple of point-and-click

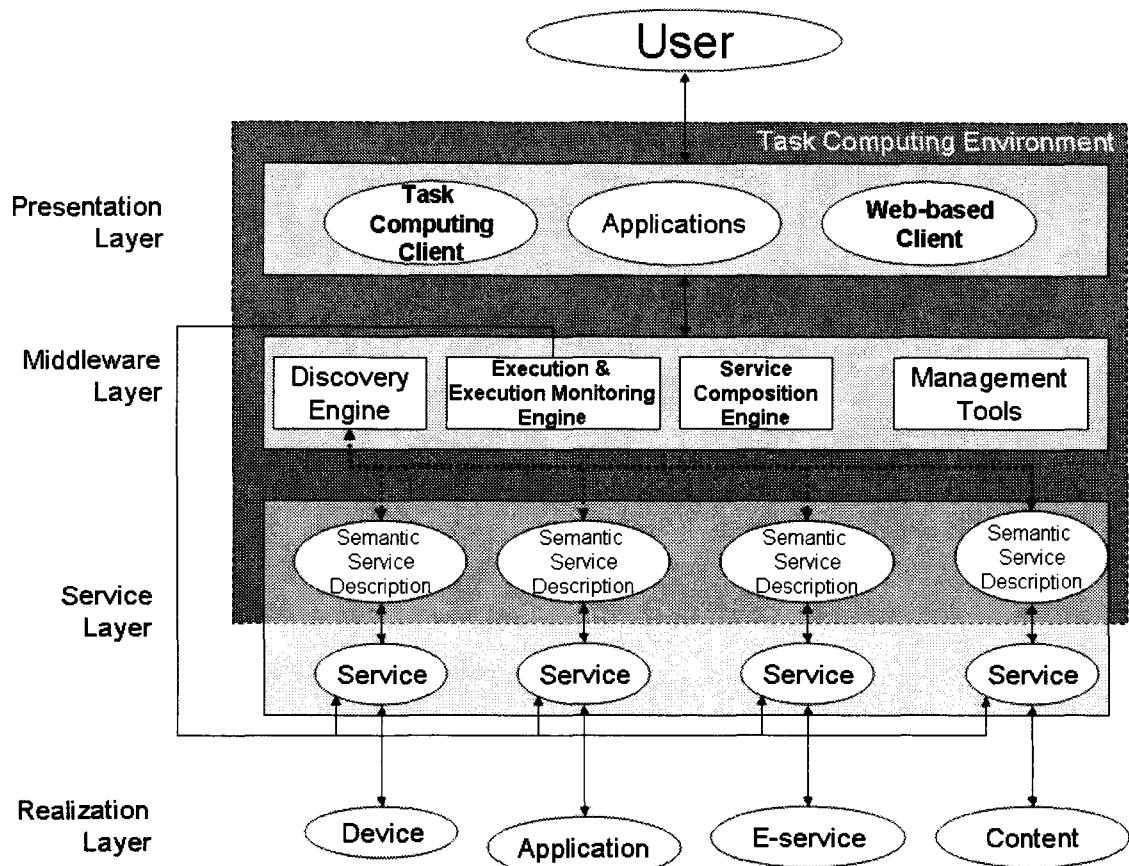


Figure 1: General Architecture of Task Computing

operations or with a voice command issued by a user.

We will also show how easy a user can publish semantically described entities such as operating system functions or application objects in TCE and disseminate them with our tool set. For example, an end-user can plug her digital camera to a computer and the images in the camera are published within the sub-network. The semantically described objects (image files) can be immediately chained with a “Digital Photo Frame” service to display the file, or be chained with a Bank service to share with other users (even after she leaves the environment).

We envision Task Computing as a very useful and powerful framework in pervasive environments like hospitals, offices, and homes where the end-user can seamlessly integrate and manipulate functionalities on her own computer, the devices

around her, and remote web services, enabling her to easily and dynamically define, execute and monitor complex tasks, in ways that can only be accomplished today by painstaking, design-time integration provided by experienced programmers.

Reference:

1. Resource Description Framework (RDF), <http://www.w3.org/RDF/>
2. OWL Web Ontology Language Overview, <http://www.w3.org/TR/owl-features/>
3. OWL-S 1.0 Release, <http://www.daml.org/services/owl-s/1.0/>
4. Universal Plug and Play, <http://www.upnp.org/>

Author Index

-A-

- Aalto, Lauri*.....149
Abbott, Derek.....245
Akkiraju, Srikanth.....193
Ala-Kurikka, Jukka.....63
Arhippainen, Leena.....79

-B-

- Bederson, Benjamin B.*.....19
Bessa, Maximino.....5
Bianchi, Michael.....117
Bove, Jr, V. Michael.....1
Brunnberg, Lilseott.....33
Burak, Asaf.....93

-C-

- Cable, Adrian*.....1
Chalmers, Alan.....5
Chen, Chang Wen.....125
Chen, Yi-Chao.....141
Chu, Hao-Hua.....141
Coelho, António.....5

-D-

- Davidyuk, Oleg*213

-E, F-

- Ege, Raimund K.*251
Emerit, Marc229
Ezenwoye, Onyeka251
Feijs, Loe M.G.11
Fujinami, Kaori.....55

-G, H-

- Häkkilä, Jonna*.....133, 179
Harjula, Erkki.....63
Heng, Pheng-Ann.....237
Henrysson, Anders.....41
Holmquist, Lars Erik.....47
Hori, Taro.....207
Hsu, Jane Yung-Jen.....141

-I, J-

- Inakage, Masa*.....199
Ishikawa, Hiroo.....55
Jacucci, Giulio.....157
Jun, L.....165
Jurmu, Marko.....85

-K-

- Kallio, Sanna*.....25
Känsälä, Ilkka.....133
Karvonen, Jari.....171
Katayama, Atsushi.....101, 109
Kazaura, K.....165
Kela, Juha.....25, 133, 157
Kharma, Qasem.....251
Khella, Amir.....19
Kotabe, Taku.....199
Korhonen, Jani.....149
Korpipää, Panu.....25, 133

-L-

- Labrou, Yannis*.....257
Latvakoski, Juhani.....71
Li, Houqiang.....125
Lu, Yang.....237

-M-

- Mäntyjärvi, Jani*.....25, 179
Masouka, Ryusuke.....257
Matsumoto, Mitsuji.....165, 207

-N-

- Nanda, Gauri*.....1
Nakajima, Tatsuo.....55
Nakamura, Takao.....101, 109
Niemelä, Eila.....71

-O-

- Ollila, Mark*.....41
Ojala, Timo.....149
Osborne, Dominic.....245

-P-

- Parhi, Pekka*.....149
Pernaux, Jean-Marie229
Plomp, Johan157

-Q, R-

- Qian, Yuechen*.....11
Raghavan, Sriram193
Rautio, Ville-Mikko.....79, 213
Repo, Pertti221
Riekki, Jukka85, 213, 221
Rogers, Derek245
Ronkainen, Sami133

-S-

- Sanneblad, Johan*.....47
Sauvola, Jaakko63, 85
Sharon, Taly.....93
Smith, Marvin L.187
Sonehara, Noboru101, 109
Song, Zhexuan.....257
Sorell, Matthew245
Sridhar, Santhosh.....193
Sun, Jun-Zhao85, 213
Sutinen, Tiia149

-T-

- Tähti, Marika*79
Teng, Chao-Ming141
Tokuhisa, Satoru199
Tokunaga, Eiji55
Touimi, Abdellatif Benjelloun231

-U, V, W-

- Wang, Yi*125
Warsta, Juhani171
Wong, Tien-Tsin237
Wu, Chon-In141

-X, Y, Z-

- Yamamoto, Masashi*101, 109
Yliantila, Mika63
Yang, Li251
Zahid, K.K.A.165

