Improving Accuracy and Practicality of Accelerometer-Based Hand Gesture Recognition

David Mace
Archbishop Mitty High School
davidmace13@
mittymonarch.com

Wei Gao
Boston University
ECE Department
davidgao@bu.edu

Ayse K. Coskun
Boston University
ECE Department
acoskun@bu.edu

ABSTRACT

Wrist-watches are worn by a significant portion of the world's population, but their potential usefulness is not only limited to checking the time. Watches are located in a prime position to retrieve valuable position and acceleration data from a user's hand movements. In this paper, we explore the plausibility of using watches containing accelerometers to retrieve acceleration data from hand gesture motions for use in human-computer interaction tasks.

We compare two approaches for discerning gesture motions from accelerometer data: naïve Bayesian classification with feature separability weighting and dynamic time warping. We introduce our own gravity acceleration removal and gesture start identification techniques to improve the performance of these approaches. Algorithms based on these two approaches are introduced and achieve 97% and 95% accuracy, respectively. We also propose a novel planar adjustment algorithm to correctly recognize the same gestures drawn in different planes of motion and reduce spatial motion dissimilarities.

Author Keywords

gesture recognition; accelerometer; watch gesture recognition; Bayesian classifier; feature separability weighting; dynamic time warping; plane adjustment

ACM Classification Keywords

I.5.2[Pattern Recognition]:Design Methodology – Classifier design and evaluation

INTRODUCTION

There is immense untapped potential for more natural human-computer interaction that lies within watches. Introducing computing power into watches by adding accelerometers and wireless transmission capabilities will allow us to increase the diversity of the ways in which we use watches in our daily lives.

Gesture recognition is a growing area of interest because it provides a natural, expansive interface for humans to communicate with computers. The increased versatility and fluidity of hand gesture motions in comparison to key presses and finger swipes allows people to more seamlessly communicate with digital devices. Accelerometers implanted in wrist watches worn on users' hands can register unique acceleration signatures of motions that can be processed into simple motion types for use in various applications.

Using a watch with an accelerometer has lower complexity and cost compared to camera-based gesture recognition [1]. In addition, gesture recognition with accelerometers worn on the hands is simpler to set up than camera-based gesture recognition because a user does not need to face a particular direction or sit in front of a screen. For example, a user wearing a watch can control a stereo with a wave of the hand while sitting in a different room or scroll through a public display from a distant seat.

In this paper, we discuss two approaches, (1) Feature Weighted Naïve Bayesian Classifiers [3] and (2) Dynamic Time Warping [4], which require a smaller number of training samples but still provide high accuracy. We also introduce our own improvements to these algorithms that improve their usefulness in accelerometer-based gesture recognition.

Previous work has explored watch-based gesture recognition using dynamic time warping [9]. In this paper, we attempt to expand on previous research by testing the efficacy of rotationally normalizing gestures and applying feature weighted naïve Bayesian classification to gesture recognition.

EQUIPMENT

Our implementation uses a TI eZ430-Chronos watch, which is cheap and simple to use, as the accelerometer data provider. The watch contains a VTI-CMA3000 3-axis accelerometer, with a measurement range of 2g, 8-bit resolution, and 100Hz sampling rate.

IUI 2013 Workshop: Interacting with Smart Objects, March 18, 2013, Santa Monica,CA, USA

We use an ASUS TF300T Android tablet to run our algorithms (which are all implemented with Java); however, our implementation can be used with any Android device and can be ported to other mobile platforms. The tablet receives accelerometer data from the watch through an RF-receiver with USB interface, which is recognized as a serial port inside of Android.

Although we use a TI EZ430 Chronos Watch in our trials, any watch that can transmit data to a digital device could be used to achieve the same purpose.

METHODS

The proposed gesture recognition methods can be split into three main phases. The preprocessing phase converts the acceleration measurements into a form that is more easily recognizable. The plane adjustment phase makes the acceleration data rotationally independent. The gesture identification stage uses either weighted feature classification or dynamic time warping to predict the most likely gesture given the acceleration measurements.

Preprocessing:

The raw data set received from the accelerometer is noisy, contains still frames, and is skewed by gravity so the data must be adjusted before they can be properly classified.

The first step in the preprocessing phase is the removal of the acceleration caused by gravity from the watch's acceleration measures. Assuming the rotation of the watch is held reasonably constant throughout the gesture, the average of all of the acceleration measurements on each axis in practice approximately represents the constant value of gravity on that axis. To eliminate the effects of gravity, this average value is subtracted from each axis at each frame.

Still frames at the beginning and end of the data that are not part of the gesture are also removed. Still frames are detected by checking the average acceleration in each 0.5 second window. If the acceleration in a window is below a constant threshold, then that window is removed from the gesture.

The jolty nature of hand motions and the discrete sampling of the gestures contribute white noise to the data. A low-pass filter is used to extract the main gesture motion from the noisy accelerometer data. This common process is integral in recognizing gestures because it eliminates high-frequency noise while revealing underlying low-frequency patterns in the data. Figure 1 shows the difference between the acceleration data before and after the low pass filter is applied.

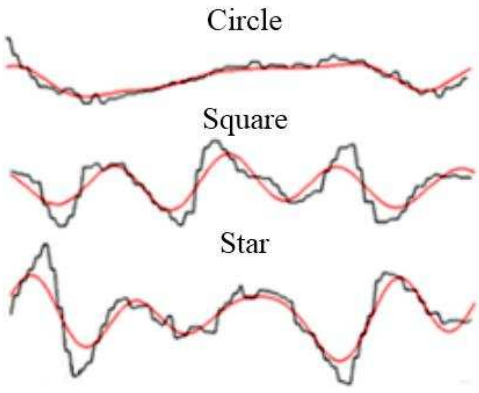


Figure 1. Acceleration graphs of gesture trials. The black line is a graph of the acceleration magnitude from the watch vs time. The red line represents the acceleration graph after the low pass filter has been applied. These graphs only show the one dimensional x-axis acceleration.

Plane Adjustment:

One issue in gesture recognition that has not been explored in depth in prior work is recognizing gestures in different planes of motion as the same gesture. Sometimes when a user is told to trace a circle, he or she does so in the xy plane, but other times he or she might trace it in the yz plane, or in some plane in between.

Even if the user is trying to make all of the motions in a single plane, there are also usually slight discrepancies in the planes of motion among different gesture trials. To allow for more realistic and orientation-independent communication through the watch, a plane adjustment phase is included in our algorithm.

In this phase, first, the best-fit plane (shown in red in Figure 2) of the acceleration vectors is found. The rationale behind this is that if the motion lies in a single plane, then the acceleration vectors of a closed shape (e.g., a circle) should on average lie in that main plane. As there could be many motion vectors in the motion that do not lie in the main plane even after using a low-pass filter, all acceleration segments between points of inflection are added up to form one vector. In this way, we can identify the general direction of the user's motion, rather than identifying each individual motion segment.

If these gesture segments are represented as a set of vectors $\{p_i = \langle x_i, y_i, z_i \rangle\}_{i=1}^{n}$ and the plane is represented by the equation $z = Ax + By + C$ then the best fit plane is found by minimizing the error, which is

$$\sum_{i=1}^{n} (Ax_i + By_i + C - z_i)^2.$$

To find the best fit plane, the following matrix is solved using Gaussian Elimination [5].

$$\begin{bmatrix} \sum_{i=1}^{m} x_i^2 & \sum_{i=1}^{m} x_i y_i & \sum_{i=1}^{m} x_i \\ \sum_{i=1}^{m} x_i y_i & \sum_{i=1}^{m} y_i^2 & \sum_{i=1}^{m} y_i \\ \sum_{i=1}^{m} x_i & \sum_{i=1}^{m} y_i & \sum_{i=1}^{m} 1 \end{bmatrix} \begin{bmatrix} A \\ B \\ C \end{bmatrix} = \begin{bmatrix} \sum_{i=1}^{m} x_i z_i \\ \sum_{i=1}^{m} y_i z_i \\ \sum_{i=1}^{m} z_i \end{bmatrix}$$

After the best fit main plane is found, each vector is normalized relative to this plane (shown in Figure 2).

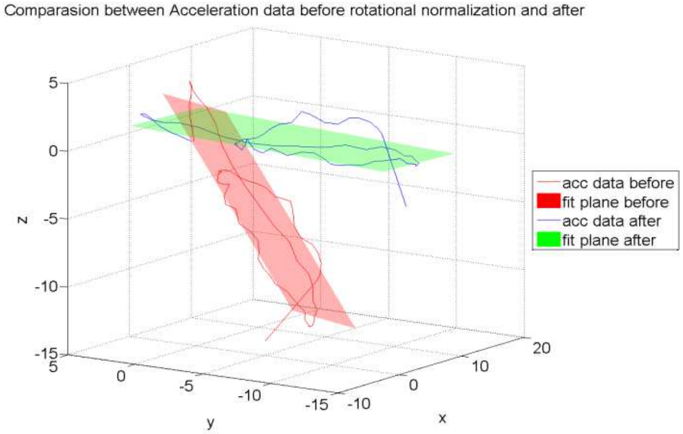


Figure 2. The red curve represents a circle gesture performed in the yz plane and the blue curve represents the same gesture after its acceleration data has been reoriented relative to the xy plane.

The previous step takes into account rotation about the x and y axes, but does not account for rotation about the z axis. To fix this, the approximate best-fit line inside the best-fit plane is found. To approximate the best fit line, the lines extending at angles of α = 22.5°, 45°, 67.5°…180° from the origin are tested and the best fit line of these is chosen.

For the set of points $\{p_i = \langle x_i, y_i, z_i \rangle\}_{i=1}^{n}$the best fit α value is obtained by minimizing $\sum_{i=1}^{n} |\alpha - \tan^{-1}(y_i/x_i)|$. This equation was chosen because it calculates the sum of diffences between acceleration vector angles and the candidate best fit line. We want to find the line on which most acceleration vectors approximately fall, so using the differences between angles is logical.

Once α is found, a new angle $\beta_i = \tan^{-1}(y_i/x_i) - \alpha$ is calculated for each vector. A final vector $u_i =< \sqrt{x_i^2 + y_i^2} * \cos\beta, \sqrt{x_i^2 + y_i^2} * \sin\beta, z_i >$, which is the original vector adjusted relative to the best fit line, replaces each original acceleration vector.

Gesture Identification

We compared two approaches to identify gestures based on a user's acceleration data.

(a) Feature Weighted Naïve Bayesian Classification:

Naïve Bayesian Classification [3] is a promising technique in gesture recognition because it can make accurate predictions by using statistical measures to calculate membership probabilities. In our implementation of this algorithm, twenty statistical features are extracted from the acceleration data. These include common statistical measures such as interquartile range, average energy, maximum of absolute value, and standard deviation.

Before a user operates the system, the user registers a set of training gestures. A weight between 0 and 1 is calculated for each feature type based on the similarity of feature measures of the different trained gestures of the same gesture type. A weight value close to 1 represents very precise measures and a value close to 0 represents imprecise measures.

When the user is running the gesture recognition system, feature measures are extracted from the user's registered gesture. The proximity of each feature measure to the average trained feature measure of each gesture type is calculated by a normal distribution by the following equation:

$$proximity = e^{-(feature\ measure - trained\ avg)^2/(trained\ \sigma)}$$

Then this proximity value is multiplied by the feature weight that was calculated in the training phase. All of these multiplied values are summed up and the system predicts the user's gesture to be the gesture type with the greatest calculated value.

(b) Dynamic Time Warping (DTW):

DTW is a widely used algorithm in gesture recognition that calculates the similarity between two time-series data sets. This algorithm is based on the idea that to find the time-independent similarity between a gesture and a template, the i^th point of the gesture can be aligned (warped) to the j^th point of template [4].

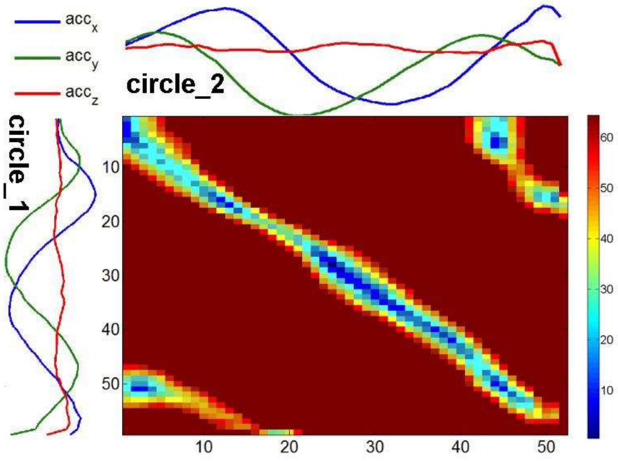


Figure 3. Each point in the grid represents the geometric distance between Circle_1 at index y and Circle_2 at index x. For example, to match up Circle_1 at index 10 with Circle_2 at index 3 requires a geometric distance of about 45.

Figure 3 provides a visual illustration of the process of DTW of two sets of data. In this algorithm, first a matrix **A** is calculated. Each element a(i,j) in the matrix represents the geometrical distance between the sample data at time t(i) and template data (collected in training phase) at time t(j). Any gesture that is "close" to the template data is likely to be of the same gesture type. Second, a path in the matrix **A** is found so that among all of the paths from a(0,0) to a(n,m), the sum of all the elements on the path (P_sum) is minimized.

The above two steps give a value P_sum representing the similarity between one sample data set and one template (training) data set. Then these steps are completed for all of the sample/template data pairs. The pair that has the smallest "path sum value" indicates the predicted gesture.

RESULTS

Naïve Bayesian and Dynamic Time Warping:

We tested both techniques using five gesture samples of four gesture types from five different people. The tested gesture types were circle, figure eight, square, and star. The average accuracy was 97% for the feature separability weighted Bayesian Classifier, and 95% for the dynamic time warping.

Both of the proposed methods have comparable accuracy with previously tested Hidden Markov Models and k-mean algorithms [6,7]. However, feature separability weighted naïve Bayesian classifiers and dynamic time warping run faster on large data sets and require a smaller number of training samples [2].

Plane Adjustment:

When five training samples per gesture type are used, the average success of the feature separability weighted naïve Bayesian classification with plane adjustment is 83.75%, compared to 72.5% success without plane adjustment. When 10 training samples per gesture type are used in training, classification accuracy with plane adjustment improves to over 90%. Table 1a and 1b show the specific performance of plane adjustment for each gesture type when naïve Bayesian classification is used.

With Plane Adjustment					
		Predicted Class			
		Circle	Figure8	Square	Star
Actual Class	Circle	20	0	0	0
	Figure8	2	16	2	0
	Square	0	2	18	0
	Star	0	0	7	13

Table 1a. Results when plane adjustment is used. Each gesture type on the top is how the algorithm classified the motion and each gesture type on the left is how the motion should have been classified.

Without Plane Adjustment					
		Predicted Class			
		Circle	Figure8	Square	Star
Actual Class	Circle	20	0	0	0
	Figure8	0	14	2	4
	Square	2	0	12	6
	Star	0	6	2	12

Table 1b. Results for the same gesture motions when plane adjustment is not used. Each gesture type on the top is how the algorithm classified the motion and each gesture type on the left is how the motion should have been classified.

APPLICATIONS

Watch Gesture Recognition

The use of a common watch equipped with an accelerometer is sufficiently cheap and non-invasive to make it practical for real-world use in a variety of applications.

The most direct application of this technology is in more natural and flexible communication with digital devices such as tablets, televisions, and stereos. When sitting in front of a screen, a user could rotate a graphic on a presentation by moving his or her hand in a circular motion, automatically bookmark a webpage by making a star motion, or use a hand gesture as a shortcut to go to his or her emails. Of course, this form of interaction could not replace a keyboard and mouse for certain tasks, but it still opens the door for more diverse and seamless interaction with users.

This setup could also allow a user to remotely control devices when he or she is unable to or does not want to touch a device. A user could use the watch as a more natural universal remote to change the channel on the television, turn off a computer, or turn off the lights. Also in a public situation in which diseases can be spread by touch, users could interact with public displays like ATMs and airport kiosks through the watch instead of by touch.

Also there are many situations where people want to control a digital device but touch or keyboard control is impractical. When a user is cooking, wearing gloves, or driving, he or she may be unable to control a stereo, computer, or other device. Accelerometer-based gesture recognition through a watch is a feasible solution to this problem because a user could perform a hand gesture to control a device when they cannot access it directly.

Additionally there is tremendous potential for watch accelerometer based gesture recognition in immersive game technologies. This was recently evinced by the tremendous success of the Nintendo Wii. The Wii is primarily focused on recognizing short, linear motions and using a remote to track a cursor on the screen. On the other hand, our setup is more concerned with recognizing gestures that can be added to in-game controls. These include using a circle to turn around and an up down motion to unlock a door.

We built an intelligent alarm clock Android application that uses the Chronos watch to detect if a user is asleep by checking for simple gestures [8]. We are also in the process of building Android applications that leverage the Chronos watch and gesture recognition in password detection and hand motion controlled features in media players.

Plane Adjustment

Normalizing the rotation of gestures can improve the accuracy and the flexibility of gesture recognition. An example of the usefulness of this technique is in public situations where devices communicate with different users. This form of user-independent communication is susceptible to different users using different rotations of the same gesture.

Interestingly, our plane adjustment algorithm improves gesture recognition not only in different planes (plane adjustment), but also when the watch is held in different orientations (rotation normalization). Figures 4 and 5 contain an example of the same gesture motion being performed with different watch orientations. Rotation nornalization is useful because an accelerometer device is not always fixed in one direction each time the user holds it. Often a watch is fixed at an angle as long as it’s worn on someone’s wrist. Other accelerometer-containing devices that a user might hold instead of wearing, however, would not be fixed in one orientation, so the idea of rotation normalization could be extended to these devices.

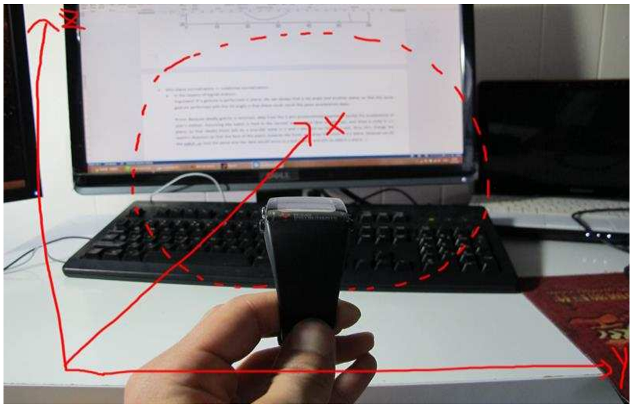


Figure 4. A circle is drawn (the dotted line) in the yz plane when the watch is tilted up.

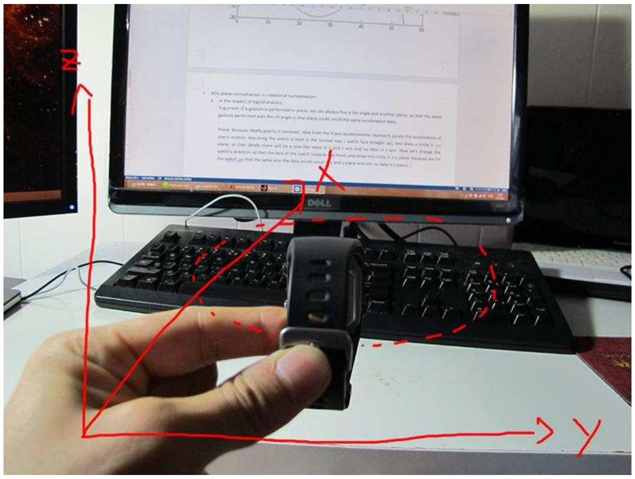


Figure 5. A circle is drawn in the xy plane when the watch is tilted to the side. This is the same motion as in Figure 4 because the watch is tilted at the same angle relative to the plane of motion.

ACKNOWLEDGEMENT

Part of this research has been conducted during David Mace’s internship in Boston University’s Research Internships in Science and Engineering (RISE) program.

REFERENCES

[1] Patlolla, C.; Mahotra, S.; Kehtarnavaz, N.; , "Real-time hand-pair gesture recognition using a stereo webcam," Emerging Signal Processing Applications (ESPA), 2012 IEEE International Conference on , vol., no., pp.135-138, 12-14 Jan. 2012 doi: 10.1109/ESPA.2012.6152464

[2] L. R. Rabiner and B. H. Juang, "An Introduction to Hidden Markov Models," in IEEE ASSP Magazine, 1986, pp. 4-15.

[3] P. Paalanen. “Bayesian Classification Using Gaussian Mixture Model and EM Estimation: Implementations and Comparisons.” Lappeenranta University of Technology, Information Technology Project Report. http://www2.it.lut.fi/project/gmmbayes/downloads/doc/report04.pdf

[4] E. Keogh, C. A. Ratanamahatana, “Exact Indexing of Dynamic Time Warping”, In Proceedings of International Conference on Very Large Data Bases (VLDB), pp. 406-417, 2002.

[5] R. Sedgewick. "GaussianElimination.java." Web. 17 Jul. 2012. http://introcs.cs.princeton.edu/java/95linear/GaussianElimination.java.html

[6] T. Schlomer, B. Poppinga, N. Henze and S. Boll, “Gesture Recognition with a Wii Controller”, In Proceedings of International Conference on Tangible and Embedded Interaction (TEI), pp. 11-14, 2008.

[7] Klingmann, Marco. "Accelerometer-Based Gesture Recognition with the IPhone." http://www.doc.gold.ac.uk/~mas02mb/Cognitive%20Computing/Dissertation/iPhone%20gestures.pdf

[8] Persistent alarm clock http://processors.wiki.ti.com/index.php/Persistent_alarm_clock

[9] Jiayang Liu, Jehan Wickramasuriya, et al. “uWave: Accelerometer-based Personalized Gesture Recognition and Its Applications.” In Pervasive Computing and Communications, 2009.

Further Investigating Pen Gesture Features Sensitive to Cognitive Load

Ling Luo, Ronnie Taib
National ICT Australia
13 Garden Street
Eveleigh, NSW 2015, Australia
{ling.luo, ronnie.taib}@nicta.com.au

Lisa Anthony, Jianwei Lai
UMBC Information Systems
1000 Hilltop Circle
Baltimore, MD 21250, USA
{lanthony, jianwei1}@umbc.edu

ABSTRACT

A person's cognitive state and capacity at a given moment strongly impact decision making and user experience, but are still very difficult to evaluate objectively, unobtrusively, and in real-time. Focusing on smart pen or stylus input, this paper explores features capable of detecting high cognitive load in a practical set-up. A user experiment was conducted in which participants were instructed to perform a vigilance-oriented, continuous attention, visual search task, controlled by handwriting single characters on an interactive tablet. Task difficulty was manipulated through the amount and pace of both target events and distractors being displayed. Statistical analysis results indicate that both the gesture length and width over height ratio decreased significantly during the high load periods of the task. Another feature, the symmetry of the letter 'm', shows that participants tend to oversize the second arch under higher mental loads. Such features can be computed very efficiently, so these early results are encouraging towards the possibility of building smart pens or styluses that will be able to assess cognitive load unobtrusively and in real-time.

Author Keywords

Gesture features; cognitive load; user study; pen-based input; vigilance; attention; human-computer interaction.

ACM Classification Keywords

H.5.2. [Information interfaces and presentation]: User interfaces – Evaluation/methodology; Input devices and strategies.

INTRODUCTION

Cognitive load represents the mental effort imposed on a participant's cognitive system when performing a particular task [8]. When a task demands very high quality of performance, like air traffic control or machine operation, the degradation in quality caused by too-high cognitive load may lead to accidents or serious consequences. For example, it was found that the lack of "at least some cognitive availability and understanding of the situation" may have led pilots to ignore continuing alarms during the fatal accident [3] on the Rio to Paris flight AF447, which disappeared over the Atlantic. Being able to assess cognitive load in real-time can allow intervention when levels become too high and can prevent such accidents. Furthermore, in other less safety-critical settings, cognitive load assessment can still be useful. For example, in an educational setting, assessing students' cognitive states could help teachers to better control teaching content and pace, and thus improve learning effectiveness and efficiency. Therefore, research into ways to estimate human cognitive state and capability is critical to improving the quality of human computer interaction, increasing task performance, and developing optimized human-decision-support applications. Practically speaking, it is important to look for good methods of measurement and estimation, which will be not only accurate, but also unobtrusive and real-time, so they can reduce noise, unpredictable factors, and disruptions to the cognitive process.

There are four different ways to measure cognitive load explored in the literature [8]: (i) subjective assessment techniques; (ii) task and performance based techniques; (iii) behavioral measurements; and (iv) physiological measurements. Typically, more than one method is used, so their combination can improve accuracy. In our user study, both behavioral and physiological measures were used, but this paper focuses on behavioral measurement. Behavioral measurement is here defined as non-obtrusive data collection during natural multimodal interaction. In this paper, we report on a user study in which participants performed a vigilance-oriented, continuous attention, visual search task [2], controlled by handwriting single characters on an interactive tablet. We examine behavioral features of the pen gestures input in two cognitive load levels to identify efficient features that can be computed in real-time.

Prior work has looked at how gesture features are related to changes in cognitive load induced by task complexity [10] and task memory demands [11] for simple shapes (circles and crosses). That work found that features such as shape degeneration [10] and pen trajectory duration, speed, and length [11] are correlated with increases in cognitive load. Our study expands on that prior work by (a) using another way to induce cognitive load (i.e., task speed), (b) probing

IUI 2013 Workshop: Interacting with Smart Objects, March 18, 2013, Santa Monica, CA, USA

additional pen input features (i.e., pressure, bounding box size, and geometric features such as the symmetry of letter 'm'), and (c) using a wider variety of pen input shapes (i.e., letters). Prior analysis based on the same experiment reported in this paper [2] found that gesture duration, number of points (correlated with speed), and gesture length were all significantly affected by cognitive load.

In this paper we extend the list of features examined and find that features such as normalized gesture length and width over height ratio decreased significantly during the high load periods of the task. Also, after examining visualizations of all recorded gestures, we noticed that some letters seemed to exhibit different forms as cognitive load varied, for example, the symmetry of the letter 'm', showed that participants tend to oversize the second arch under higher mental loads, as illustrated in Figure 1. Such features can be computed very efficiently, so these early results are encouraging towards the possibility of building smart pens or styluses that will be able to assess cognitive load unobtrusively and in real-time.

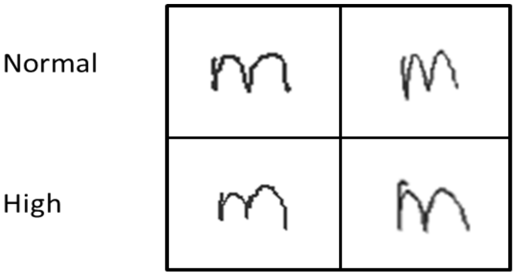


Figure 1. Sample Input from two Participants.

MOTIVATION

Cognitive Load Impacts Performance

Cognitive load is closely related to the capacity of working memory, which, according to the cognitive load theory [8] refers to the brain system providing temporary storage of the input necessary to acquire information, process it, and prepare feedback actions when completing tasks. The capacity of working memory is limited; accepted estimates of the amount of information it can hold at a time is restricted to 7 ± 2 items [5]. When cognitive load exceeds working memory's capacity limit, the participant's performance starts to degrade [10]. This degradation could lead to longer reaction times, higher error rates, and decreased control of actions and movements [11]. Therefore, we are examining methods of unobtrusively detecting cognitive load spikes, such as using the gesture features discusses in this paper, to allow systems to intervene to reduce negative impacts of higher load.

Pen Gesture Input as an Unobtrusive Sensor

Pen gestures are input produced through a pen or stylus during a user's interaction with a computer [9]. Previous research indicated that some gesture features can be used as indicators of cognitive load imposed by a task [10, 11, 12]. Compared to other modalities, pen gesture offers benefits such as naturalness for the user, low intrusiveness, and the possibility to automatically analyze data in real-time. It can capture variations in performance implicitly without interrupting the task, and the data is available for analysis once the current gesture is finished [2, 10]. In the market, there are already digital pen and paper systems, such as Anoto, which support gesture capture and online/offline analysis [1]. Prior research has shown that for specific tasks, for example, math problem solving by high school students, pen-based systems provide cognitive support and produce better learning results than traditional keyboard and mouse graphical user interfaces [6], so we believe there is similar potential in developing pen-based adaptive systems for both safety-critical and other tasks.

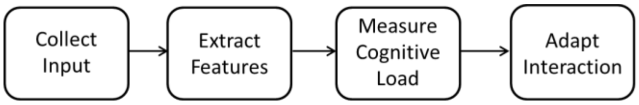

Collect Input
Extract Features
Measure Cognitive Load
Adapt Interaction

Figure 2. Smart Pen Workflow.

In our work, we use the workflow in Figure 2 as the accepted model of how smart pens process and react to user input. Based on the written content, the pen extracts geometric features and automatically classifies them according to pre-built models, possibly trained for specific users. Depending on the application, high cognitive load detection can be used to trigger alerts, e.g. when a mission-critical operator is experiencing high cognitive load, a manager may be alerted to provide additional resources or a break for the operator. In other contexts, the pace of the content can be adapted, e.g. when a student is learning online content using an interactive tablet.

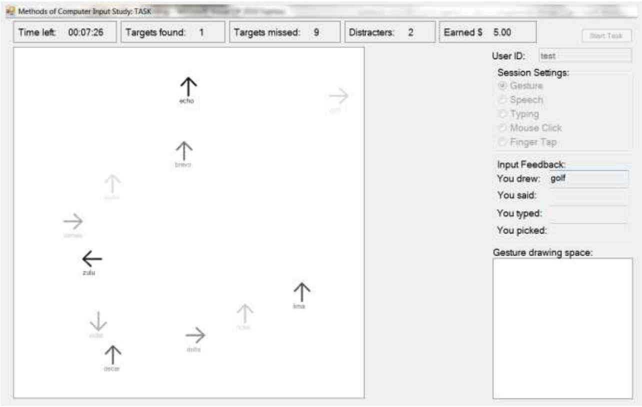
Simulating Real-World Tasks

Previous research has used a variety of experiment designs to collect pen gesture input and correlate it to cognitive load. One example is the map tasks in [7, 10], in which participants are asked to look for routes and organize a green light corridor on a city map. There are also tasks instructing participants to compose sentences from three predefined words [12] or to solve mathematics problems [6], requiring participants to write down all the intermediate processes. In this paper, we use a continuous attention, visual search task, which simulates real-world vigilance tasks such as air traffic control or information analysis. Our task has two cognitive load levels, and our analysis focuses on specific geometric features of the single letter inputs.

EXPERIMENT DESIGN

Participants performed a vigilance-oriented continuous attention and visual search task [2]. During the experiment, arrows facing one of four directions (↑, ↓, ← and →) were displayed sequentially (with some overlap) on the screen, and each of them was companied by a text identifier underneath. There were 12 possible identifiers: {**a**lpha,

bravo, **d**elta, **e**cho, **g**olf, **h**otel, **i**ndia, **l**ima, **m**ike, **o**scar, **r**omeo, **z**ulu}. At any moment, all the identifiers visible on the screen were unique. The participants were instructed to detect any arrow facing down ↓ while ignoring all the other objects (distractors) on the screen, and to write down the first letter (the highlighted character in the above list) in a "gesture drawing space" located at the bottom right of the screen. The user interface is shown in Figure 3.



Methods of Computer Input Study: TASK

Time left: 00:07:26

Targets found: 1

Targets missed: 9

Distracters: 2

Earned $ 5.00

User ID: test

Session Settings:

- ● Gesture
- ○ Speech
- ○ Typing
- ○ Mouse Click
- ○ Finger Tap

Input Feedback:

You drew: golf

You said:

You typed:

You picked:

Gesture drawing space:

Figure 3. Experiment User Interface.

There were two levels of cognitive load in the task, labeled Normal and High, and the level was manipulated by controlling the time interval between arrows and the frequency of occurrence of target objects. During High periods, the higher frequency of actions required increased intrinsic cognitive load, and the higher number of distractors increased extraneous load, so this condition is labeled high load in our analysis.

There were 12 participants (7 males and 5 females) who used the pen modality to perform the task. Two participants (both female) were excluded from data analysis due to post-hoc analysis showing that in-task recognition of their gestures had had very low accuracy (i.e., less than two standard deviations below the mean), leaving an N of 10.[1] The equipment used to collect gestures during the experiment was a Tablet PC (Fujitsu Lifebook T-900). The system collected all task information, including the target and distractor objects and their order of appearance (same sequence for every user), task performance (recognized result and the system response: True Hit, Miss, etc.), and pen input trajectories. The analysis is based on these trajectories, which store each sampled gesture point (timestamp, character written, coordinates, pressure), and whole gesture information (start and end indicators).

DATA ANALYSIS RESULTS

Bounding Box

The term bounding box refers to the smallest box that can contain a gesture entirely. The bounding box has several geometrical features, including height, width, area and the width to height ratio (width/height, which is also cot α in Figure 4).

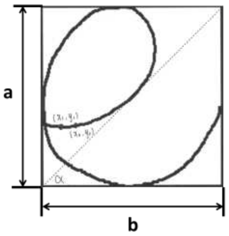


Figure 4. Defining a bounding box for the gesture.

The mean width over height ratio across all gestures is 0.851 (σ = 0.203, N = 10) in the Normal condition and 0.768 (σ = 0.169, N = 10) in the High condition. The decreasing trend of mean width over height ratio between Normal and High is consistent for all but one participant. The result of a two-tailed t-test showed the width over height ratios varied significantly between Normal and High ($t(9) = 3.05$, $p < 0.05$).

However, when comparing all letters together, we must take into account that the generic letter shapes exhibit different width over height ratios. For example, the width over height ratio for **l**ima and **i**ndia are quite small compared to the ratio for **m**ike. Moreover, the frequency of occurrence of different letters was not completely balanced between the two conditions because there were many more targets in the High condition. For example, **l**ima or **i**ndia appear 9 times as the targets in the High condition, but only once in the Normal condition. Therefore, the significant result above may be partly due to a bias linked to targets (letters) with smaller width over height ratio occurring more frequently in the High condition.

In order to mitigate the effect of the shape of the letter, the ratio of each gesture was normalized by a "standard" ratio for each specific letter, reflecting the shape of the letter. The standard ratio is calculated as the average width over height ratio across all occurrences of that letter from all participants, across both conditions. (The limited number of gestures from each participant did not permit us to establish a standard ratio per letter per participant.) Each individual occurrence of a letter is normalized by dividing its ratio by the standard ratio for that letter:

$$\text{normalized\_ratio} = \text{original\_ratio} / \text{standard\_ratio}$$

The standard ratios in Table 1 validate that different ratios apply to different letters. For example, **i**ndia and **l**ima have relatively small values, whereas **m**ike and **z**ulu are larger.

[1] This low accuracy could have caused an additional load on the user and, for this investigation, we wanted to isolate the load caused by the task difficulty manipulation only.

After that processing, a two-tailed t-test was used once more on the normalized width over height ratios, but found no significant differences between Normal and High conditions when controlling for properties of the letter entered (t(9) = -0.32, n.s.).

Letter	Standard Ratio	Letter	Standard Ratio
alpha	1.13	**i**ndia	0.19
bravo	0.62	**l**ima	0.40
delta	0.60	**m**ike	1.48
echo	1.03	**o**scar	0.79
golf	0.53	**r**omeo	0.97
hotel	0.64	**z**ulu	1.87

Table 1. Standard Letter Ratio for each letter in the study.

Gesture Pressure

Every point of a gesture has a pressure value as sensed by the hardware during input, and we define gesture pressure as the mean pressure value across all points of that gesture.

The pressure sensing capability malfunctioned during the study, so another two participants had to be excluded from just this analysis (leaving an N of 8). The mean values of gesture pressure for the remaining 8 participants were 25,743 screen dots in the Normal condition and 26,164 dots in the High condition (the TabletPC range was [0~32,767 screen dots]). A similar normalization process to the one described for the bounding box was used here:

$$\text{normalized\_pressure} = \text{current\_pressure} / \text{standard\_pressure}$$

where the standard pressure for a specific letter is calculated as the average pressure across all occurrences of that letter from all participants, across both conditions. The mean values after normalization were 0.936 ($\sigma = 0.141$, N = 8) and 0.938 ($\sigma = 0.126$, N = 8) for Normal and High conditions, respectively. These values indicate that participants tended to press slightly harder in the High condition than in the Normal condition. However, a two-tailed t-test indicated that this trend was not significant (t(7) = -0.26, n.s.), reducing the utility of this feature for detecting changes in cognitive load.

Gesture Length

Gesture length is the sum of the Euclidean distances between every two consecutive points in a single gesture, which is computed by the following formula:

$$\sum_{i=0}^{n}(\sqrt{(x_{i+1}-x_i)^2+(y_{i+1}-y_i)^2})$$

where (x_i, y_i), (x_{i+1}, y_{i+1}) are the coordinates for two consecutive points, and the gesture length is the sum of Euclidean distances between every two consecutive points.

Extending prior results [2], here we normalize length using a standard length (defined as the mean length across all occurrences of each letter from all participants, across both conditions). After such normalization, the mean values of gesture length were 1.08 ($\sigma = 0.335$, N = 10) for Normal and 0.93 ($\sigma = 0.29$, N = 10) for High, indicating a shorter gesture length in the High condition which was significant by a two-tailed t-test (t(9) = 3.79, $p < 0.05$), further supporting this feature's relationship to cognitive load.

Symmetry of letter 'm'

To examine whether our anecdotal observations of differences in the symmetry of the letter 'm' were supported by the data, we compared the widths of the left arch and right arch in High and Normal load conditions.

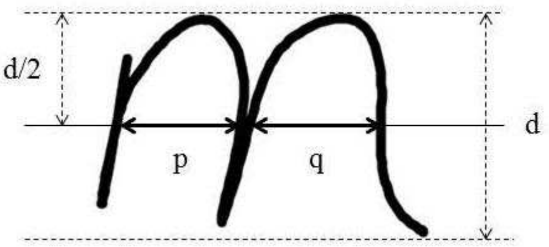


Figure 5. Symmetry of the letter 'm'.

Figure 5 illustrates how to extract the feature. The first step is to find the bounding box of the letter, then compute the distance d between the top and bottom lines, and then draw a straight horizontal line in the middle of the box, d / 2 away from top and bottom lines. Typically, there will be about 5 crossing points between the line and the gesture, but the actual number of crossing points may vary according to the way the letter was written. After that, the ratio between the two longest segments of the horizontal line, q and p is used to check the symmetry of this letter:

$$\text{Symmetry} = q / p$$

The closer to 1 this value is, the more symmetrically the letter was formed; otherwise, the left arch may be much larger or smaller than the right one. Although this method is only an estimation of symmetry, it is easy to implement and can be computed very efficiently.

The mean symmetry value in the Normal condition was 0.88 ($\sigma = 0.751$, N = 10), which was smaller than 1.28 ($\sigma = 0.418$, N = 10) observed in the High condition, but a two-tailed t-test (t(9) = -1.59, n.s.) showed that this difference was not significant. We also assessed the same feature at different horizontal cross sections of the letter, for example moving the crossing line up or down by 10% away from the middle height, but there were still no significant differences in the t-test results. Hence, while there was a trend for participants to write the right arch wider than the left arch under higher mental load, the fluctuation was not significant from a statistical viewpoint.

DISCUSSION

As mentioned, prior work has examined what gesture features are related to changes in cognitive load [10, 11, 12]. Our study expands on prior work by (a) exploring another way to induce cognitive load (i.e., task speed), (b) probing additional pen input features (i.e., pressure, bounding box size, and geometric features such as the symmetry of letter 'm'), and (c) using a wider variety of pen input shapes (i.e., letters). Shape degeneration [10] and pen trajectory duration, speed and length [11] have been found in other experiments to correlate to cognitive load. Previous analysis based on the same experiment reported in this paper [2] also found that gesture duration, number of points (correlated with speed), and gesture length were significantly affected by difficulty-induced cognitive load. We extended this past work by looking at even more gesture input features and explored how they responded to changes in cognitive load, all in search of a canonical set of gesture features that can be efficiently computed in real-time systems and are responsive to all types of cognitive load.

In the feature analysis, normalization over the standard value of the features per letter played a critical role, by effectively decreasing the impact of factors that vary from letter to letter. These factors included, among others, bounding box size and gesture length. Normalization also compensated for the unbalanced letter distribution of the task design. During the analysis, features like width over height ratio and gesture length showed statistically significant relationships to cognitive load originally. However, after normalization, differences in the width over height ratios between load conditions were not statistically significant, indicating that previous positive results may be affected by the letters that were input.

In summary:

- Both gesture length and normalized gesture length exhibited significant relationships with cognitive load.
- The bounding box width over height ratio showed significant differences as the load increased, but after normalization (over standard letter ratio), it was not significantly affected.
- Neither gesture pressure nor normalized gesture pressure were significantly affected by increasing cognitive load.
- The symmetry of the letter 'm' exhibits an increased trend in the high load condition, but it is not significant.
- From a practical perspective, a simple feature like gesture length can estimate cognitive state unobtrusively and can be computed very efficiently, making it a good candidate for a smart pen or stylus.

The dimensions of the gesture bounding box are important features. The results showed declining trends during High load (although not significant), indicating that cognitive load may impact fine-grained handwritten production, although degeneration in gesture shapes were observed in past research [10]. However, it has also been postulated that handwriting skills are based on primary communication systems (folk psychology) and hence should not get taxed by variations in cognitive load [4]. Further experimentation is required to determine which explanation prevails. In particular, in future explorations, we would change the experiment design in order to balance written input and collect more samples of each type of letter from more participants. Such changes would also allow us to group participants based on post hoc analysis of their individual inputs, for example, users who tend to write larger or smaller or experience larger impacts due to cognitive load.

The symmetry of the letter 'm' is an early attempt at exploring specific geometric features of individual pen gesture shapes. The results highlighted an increasing trend during High load (although not significant), which means higher cognitive load may have an effect on the way people form their letters, especially towards the end of the gesture: the right arch tended to increase in width compared to the left one under higher load. Again, a possible reason for the non-significant trend might be that there are individual differences among participants which could be explored by capturing more data in the future.

CONCLUSIONS AND FUTURE WORK

Focusing on smart pen or stylus input, this paper explores features capable of detecting high cognitive load in a practical set-up. Participants performed a vigilance-oriented, continuous attention, visual search task, controlled by handwriting single characters on an interactive tablet. Task difficulty was manipulated through the amount and pace of both target events and distractors being displayed. Both gesture length and width over height ratio decreased significantly in the high load session. Another feature, the symmetry of the letter 'm', showed that participants tend to write the right arch wider than the left one under higher mental load. Gesture pressure and bounding box size were not significantly affected by cognitive load, though. Features such as gesture length can be computed very efficiently, making them good candidates for a smart pen or stylus to assess cognitive load unobtrusively in real-time.

In the future, more research will be needed to validate these results and to explore more gesture features to detect changes in cognitive load robustly. For example, other geometric features will be explored, such as angles or curvature of segments composing letters.

In order to ensure the high load we impose on the participants is actually high enough, we are planning to modify the experiment design through different timings and distracters, and by adding other sources of load, for example, using a dual task methodology. We will also balance the number of individual letters collected under each condition, and increase the number of inputs elicited so we can analyze the gesture data on a per-user basis.

We believe that a combination of features will be required to estimate cognitive load from handwritten input. Once identified, this work can lead to the construction of smart pens and styluses that will be able to monitor a user's performance and adapt the task at hand implicitly to moment-to-moment fluctuations in cognitive load.

ACKNOWLEDGEMENTS

The authors acknowledge Andrew Sears, Patrick Carrington, Peng Chu, and Christopher Kidd for contributions to the original study. We also thank Northrup Grumman for partial support of the original study.

REFERENCES

[1] Anoto website. http://www.anoto.com/about-anoto-4.aspx.

[2] Anthony, L., Carrington, P., Chu, P. and Sears, A., Gesture Dynamics: Features Sensitive to Task Difficulty and Correlated with Physiological Sensors, ICMI 2011, MMCogEmS.

[3] BEA, Final Report On the accident on 1st June 2009 to the Airbus A330-203 registered F-GZCP operated by Air France flight AF 447 Rio de Janeiro - Paris (2012).

[4] Geary, D. C., Principles of evolutionary educational psychology. Learning and Individual Differences, 12, 317-345.

[5] Miller, G., The Magic Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information, The Psychological Review, vol. 63 (1956), 81-97.

[6] Oviatt, S., Human-centred Design Meets Cognitive Load Theory: Designing Interfaces that Help People Think, MM 2006.

[7] Oviatt, S., Coulston, R. and Lunsford, R., When Do We Interact Multimodally? Cognitive Load and Multimodal Communication Patterns, ICMI 2004.

[8] Paas, F., Tuovinen, J.E., Tabbers, H. and Van Gerven, P., Cognitive Load Measurement as a Means to Advance Cognitive Load Theory, Educational Psychologist, 38(1) (2003), 63-71

[9] Rubine, D., Specifying Gestures by Example, Computer Graphics, Vol. 25, No. 4 (1991).

[10] Ruiz, N., Taib, R., Shi, Y., Choi, E., and Chen, F., Using Pen Input Features as Indices of Cognitive Load, ICMI 2007.

[11] Ruiz, N., Feng, Q.Q., Taib, R., Handke, T. and Chen, F. Cognitive skills learning: pen input patterns in computer-based athlete training. Proc. ICMI-MLMI, 2010, ACM Press (2010).

[12] Yu, K., Epps, J. and Chen, F., Cognitive Load Evaluation of Handwriting Using Stroke-level Features, IUI 2011.

Modeling socially apt smart artifacts

Juan Salamanca
Icesi University
Calle 18 No 122 – 135 Cali,
Colombia
jsalam@icesi.edu.co
+57 2 5552334

ABSTRACT

Although smart artifacts could be designed as agents with whom humans interact, the resulting interaction between them is asymmetrical if the smart artifacts are designed solely to support the accomplishment of human plans and goals. The ontological asymmetry between both human and non-human agents prevents designers of smart artifacts to consider them as actual social actors capable of performing a social role instead of just being tools for human action. In order to overcome such asymmetry this research repositions smart artifacts as mediators of social interaction and introduces a triadic framework of analysis in which two interacting humans and a non-human agent are regarded as networked and symmetrical actors.

The implementation of the triadic framework in a staged study revealed that, in the achievement of personal goals, groups of people exhibit a social viscosity that hinders people's interactions. The mediation of purposely designed smart artifacts can reduce such social viscosity and facilitate cooperative and collaborative interactions between networked actors if they prompt the preservation of social balance, enhance the network's information integrity, and are located at the focus of activity.

Author Keywords

Interaction design; smart artifact; adaptive mediator; Actor-Network Theory; ubiquitous computing.

ACM Classification Keywords

H.5.m. Information interfaces and presentation: Miscellaneous; I.6.m. Simulation and Modeling: Miscellaneous; I.2.11. Artificial Intelligence: Intelligent agents.

INTRODUCTION

With the advent of ubiquitous computing, interaction design has broadened its object of inquiry into how smart computational artifacts inconspicuously act in people's everyday lives. User-centered design (UCD), based on the humanist assumption that people have the control over computational systems, has been the dominant methodology for the design of human-computer interaction. Although UCD approaches remain useful for exploring how people cope with interactive systems [19], they cannot fully explain how such new breed of smart artifacts mediate people's social interaction. While UCD approaches assume that human agents control interactive systems, it disregards the potential for agency of smart artifacts [1]. Other theoretical frameworks better explain social interaction mediated by artifacts such as Distributed Cognition [9], Activity Theory [10], or Actor-Network Theory [14][12][13]. The ideas discussed in this paper adopt Actor-Network Theory (ANT) as their theoretical ground.

Post-humanist thinkers such as Callon[4], Law [14], Latour [12] and Knorr-Cetina [11] contend that we are increasingly living in an object-centered society where the roles of objects are not only defined as commodities or equipment but also as activity partakers. In that vein, smart artifacts could be defined as agents involved in social practices mediating and cohering both humans and other artifacts together. According to ANT, both humans and smart artifacts are social actors who can assemble hybrid social collectives while they interact with each other. This paper offers a triadic structure of networked social interaction as a methodological basis to investigate i) how collectives of humans and smart artifacts get assembled, ii) how smart artifacts could understand their social setting and finally iii) how smart artifacts adaptively mediate people's interactions within social activities.

A future scenario of smart urban mobility reveals the intertwinement of human and non-human actors. Let us picture pedestrians and drivers intermingling with smart artifacts such as smart vehicles, smart traffic lights, adaptive speed signs and intelligent crosswalks as they circulate, coordinate turns, allow traffic flow and control agent's speed. In this ecology of actors a smart traffic light, is not only a piece of urban equipment that regulates the flow of traffic, but a networked social mediator of a complex adaptive system. Instances of smart traffic signs can be observed today in the City of Seattle. The city's active traffic management system analyses real time traffic flow and signals the best speed to individual drivers via

IUI 2013 Workshop: Interacting with Smart Objects, March 18, 2013, Santa Monica, CA, USA.

adaptive speed limit signs, aiming to procure the efficient flow of the whole community of commuters.

The goal of this paper is to present some considerations for the design of smart artifacts that can perform as social signifiers for the promotion of coordinated social interaction.

DEFINITION OF NETWORKED COLLECTIVES OF HUMANS AND SMART ARTIFACTS

A smart artifact is a scripted agent that autonomously acts in the world by adapting its own structure while preserving its organization. Smart artifacts are scripted with one or more programs-of-action by its designer. A *program-of-action* is a program of what an actor can do. As an example, a traffic light is smart if it interprets the dynamics of what drivers and pedestrians do and consequently adapts its timing to benefit people's flow preserving their integrity.

A *collective* is a hybrid social actor constituted when humans subscribe themselves to smart artifacts' programs-of-action. As an example, drivers constitute a collective with the smart traffic light (smartTL) if the former abide by the signals of the latter. The actions of the constituted collective are meaningful not only to pedestrians and drivers present at the collective's scope but to the whole network of actors participating in the practice of commuting.

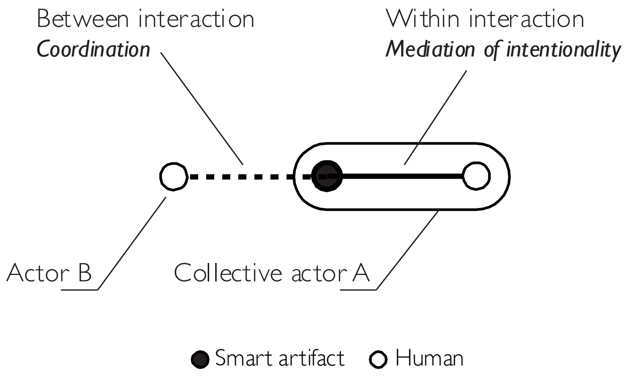


Figure 1. Triadic structure of networked social actors and its within and between interactions

This research offers a triadic structure of actors as a unit of analysis that accounts for the interactions within human-nonhuman collectives and between hybrid social actors in the actor-network. It is a triadic structure because it is composed at least of two interacting humans and one non-human agent. This triad is the basic unit of a network of symmetrical social actors. In order to exemplify within and between interactions let us picture the collective actor *A* in Figure 1 as the collective pedestrian- smartTL and the actor *B* as a car driver. The car driven by the driver is omitted in the example for simplification purposes. The *within interactions* are those that hold together humans and smart artifacts inside a collective, and put forward the collective's assembled meaning for other actors in the network. In the case of the collective pedestrian- smartTL, the interactions of both agents sensing their proximity and mutually adapting their actions to the ongoing situation hold them together within the same collective. Moreover, a car driver does not interpret the actions of a pedestrian at the street corner just as a single walker wandering around but as a pedestrian whose intention is tightly related to the current status of the nearest traffic light. The pedestrian together with the smartTL constitute a signifier for car drivers.

The *between interactions* are the social interactions that occur between collectives and characterize the dominant socio-relational model of the actor-network [8]. There is no unified classification of social interactions. Conflict, negotiation, cooperation, violence are kinds of social interaction that might emerge between social actors. This research project is particularly interested in cooperation. The interaction between the collective pedestrian- smartTL and the driver usually ends up in coordinated turn taking because the interacting collectives require the same right of passage concurrently. Turn taking is a form of cooperation. In some countries the driver yields the right of passage to the pedestrian-smartTL. But in other countries this is not the case, the socio-relational model between drivers and pedestrians privileges vehicular traffic over walkers flow.

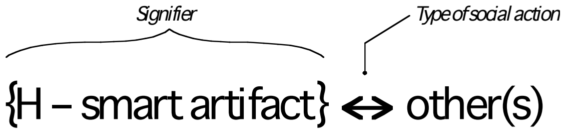


Figure 2. Notation of the triadic structure of networked social actors

Figure 2 presents a text-based form of notation of the triadic structure. The bracketed collective represents the *within* interaction and the arrow represents the *between* interaction.

As an example, {pedestrian-smartTL}→driver means that the social meaning of the collective {pedestrian-smartTL} is put forward for drivers as long as the collective persists. The within interaction of {pedestrian-smartTL} exhorts the regulation of driver's circulation flow. The between interaction corresponds to the coordination of passage between {pedestrians-smartTL} and drivers.

A NOTION OF AGENCY AND THE SYMMETRY OF ARTIFACTS AND HUMANS AS SOCIAL ACTORS

As surveyed by Bullington [3] the research on computational agency in social interaction has two major strands of research. On the one hand, we have the human-agent approach represented by the goal of the Turing test. Its object of concern is the socialization of humans with artificial agents [2][17][5]. On the other hand, the structuralist approach focused on the analysis of the structure of social groups that emerges from the inter-subjectivity of agents. Its object of concern is the bonding structures from which a collective of agents emerge and evolve [7][15].

ANT aligns with the latter approach. The symmetry proposed by ANT endows both human and nonhumans with the capacity for social action. Such symmetry does not reduce humans to mere objects, nor does it grant intentionality to objects. To be clear, symmetry does not have a geometrical meaning. The symmetry of social actors is an analytical viewpoint that positions people and objects as members of a social set without dichotomizing them. Under ANT, there is no hierarchy between human and nonhuman actors. Human and nonhumans are social actors that are placed on an equal footing, whose forms of actions simply differ. As Law puts it by drawing a distinction between ethics and sociology, the symmetry between human and nonhuman actors "is an analytical stance, not an ethical position" [14].

The fact that human and nonhuman actors are not dichotomized enables us to declare them as instances of the same class of behavioral agents. The main attribute of this class is embodiment, and the class' primary function is to react. *Behavioral social action* was described by Schutz as a reactive action triggered by external conditions [18]. *Proactive social action* as explained by Schutz is a complementary type of action, characterized as intentional and intrinsic to the acting agent. Simple artifacts are behavioral agents, but both smart artifacts and humans exhibit proactive action. Figure 3 depicts how the Proactive agent class inherits the embodiment attribute and reaction function from the Behavioral agent class, and extends its functions by implementing a higher-level function: to act.

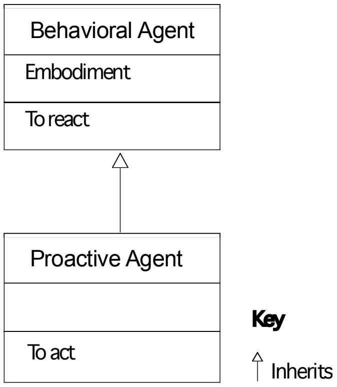


Figure 3. Class structure of behavioral and proactive agency

ANT does not claim that artifacts plan actions but rather they enact programs-of-actions. Albeit nonhuman agency appears to be a contradiction, it is systematically displayed in programs-of-action that involve the participation of artifacts [4]. In the case of humans, it is associated with their intentions. In the case of artifacts, it is associated with the criteria for social action inscribed by their designers. The significance of nonhuman action comes to light as artifacts "allow, afford, encourage, permit, suggest, influence, block, render possible, forbid […]" [13] states of affairs.

Going back to our scenario of smart urban mobility, SmartTLs could be scripted with a program-of-action that privileges pedestrians over manned and unmanned vehicles. Drivers are agents with their own behavioral and proactive programs-of-action. Table 1 presents a simplified description of the actors' programs-of-action.

Table 1. Example of behavioral and proactive programs-of-action

Agent	Type of program-of-action	Description of program-of-action
Smart Traffic light	Behavioral	Change light colors recursively
	Proactive	Privilege pedestrians flow and override behavioral program-of-action
Pedestrian	Behavioral	Avoid collisions while walking
	Proactive	Walk safely to his/her destination
Human driver	Behavioral	Abide by traffic rules
	Proactive	Drive safely to his/her destination

INTERPRETATION AND ACTION IN A SOCIAL SETTING

According to Schutz, the building blocks of an action are simple acts [18]. When an observer perceives an agent acting out its program-of-action some of its acts have been executed, whereas others are yet to be executed. The set of executed acts is referred to as *executed-program-of-action* (EPA), while the set of the yet-to-be-executed acts is referred to as *remaining-program-of-action* (RPA).

For example, Figure 3 presents the program-of-action of a person driving to a meeting composed of the following acts: A: get on the car, B: drive for ten blocks, C: park the car, D: get to the meeting on time. The RPA has a subjective meaning that is only known by the driver, i.e., no body knows where he/she is driving. In contrast, the EPA has an objective meaning because it has already been enacted in front of other agents including smart artifacts, i.e., he/she is driving somewhere. At the step *present time* in the time flow depicted in Figure 3, the EPA has an objective meaning for observers and smart artifacts, whereas the RPA has a subjective meaning known only by the driver.

By using Rough Set Theory [16] as a pattern finding technique this research proposes that smart artifacts can predict the remaining-program-of-action of human actors enrolled in a collective if the smart artifacts have a robust collection of their own executed-programs-of-action**.**

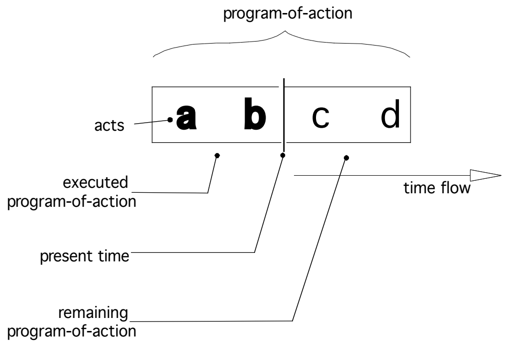


Figure 4. A program-of-action decomposed in single acts. The portion of the program-of-action enacted before the present time corresponds to the Executed program-of-action. The yet-to-be-executed portion corresponds to the Remaining program-of-action

In the execution of programs- of-action both human and nonhuman actors get intertwined and social dynamics emerge. While drivers drive, they must abide by the traffic rules reified as smart traffic lights. Concurrently, smart traffic lights adapt their timing as they sense the traffic and pedestrians approaching the intersection where they are located switching from red, to yellow, to green, accordingly and regulating the flow of traffic.

Going back to the driver's example, if at *present time* the smart traffic light turns red, it blocks the driver's action, delaying the execution of the driver's RPA – acts C and D. But, at the same time it enables the programs-of-action of pedestrians and other drivers who were waiting for their right of passage.

In ANT terms, when the actor's programs-of-action get intertwined, it is said that a human-nonhuman collective is composed. A network of collectives of behavioral and proactive agents therefore constitutes our notion of sociality. Such collectives emerge and dissolve themselves in the execution of their programs-of-action.

PROOF OF CONCEPT

An early analysis of pedestrians' trajectories in the wild revealed that it is possible to determine the subscription of actors to a crosswalk program-of-action by determining the spatial alignment of their executed-programs-of-action. The analysis showed that there is evidence of a pedestrian's subscription to a crosswalk when his/her executed program-of-action is aligned to the intended direction of travel defined by the crosswalk design, i.e. walking straight across corners. In contrast, pedestrians are not subscribed when they exhibit trajectories other than the ones outlined by the crosswalk. For example, a walker wandering erratically on the crosswalk while he/she smokes a cigarette or talks over his/her mobile phone is not subscribed to the crosswalk's program-of-action. Subscribed and unsubscribed trajectories are both socially valid, but the former is prone to elicit cooperation or collaboration among walkers present on the crosswalk concurrently, whereas the latter can drive conflicting interactions.

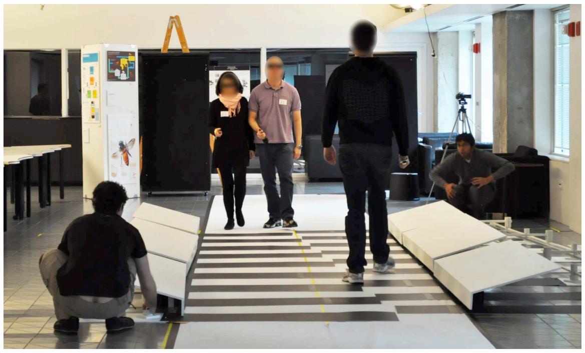


Figure 5. Wizard of Oz prototype of the study deployed at the laboratory

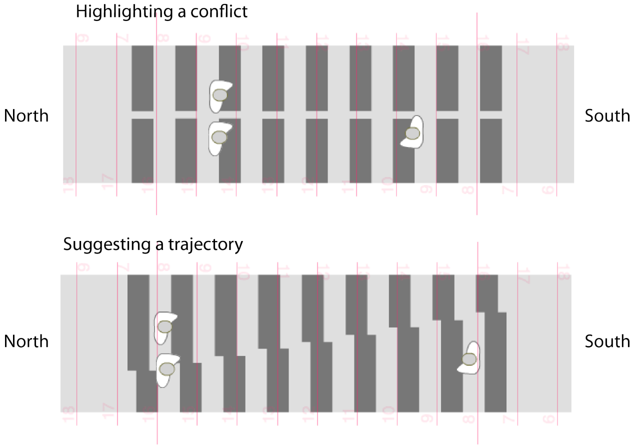


Figure 6. A smart crosswalk signaling forecasted conflicts to pedestrians

Study description

Based on the above observation, a smart crosswalk was designed and deployed in a laboratory. The smart crosswalk was scripted to dynamically signal the best distribution of the walking space among concurrent pedestrians. To do so, the crosswalk interprets the EPAs of each pedestrian and forecasts their RPAs. The assessment of multiple RPAs allows the crosswalk to identify potential conflicts in the ongoing social interaction and signals a suitable space distribution accordingly. The design tested in the laboratory consists of a striped pattern split along the north-south axis. Figure 6 shows the status of two distributions. The top illustration shows the halves of the striped pattern sliding sideways, the bottom one shows the result of the halves sliding both sideways and backwards.

Two smart crosswalks' signaling patterns were tested: i) highlighting a conflict of trajectories (Figure 6 top) and ii) suggesting trajectories to circumvent potential conflicts (Figure 6 bottom). The highlighting signaling pattern is intended to raise pedestrians' awareness to estimated trajectory conflicts. Such crosswalk's intervention is neutral because any potential trajectory negotiation is left to the concurrent group of pedestrians. The suggesting signaling pattern is intended to do a more active intervention because it suggests trajectory deviations to concurrent pedestrians biasing the outcome of any trajectory negotiation.

Sixteen subjects, selected from a pool of volunteers recruited by email on social networks, were asked to walk on both a smart crosswalk prototyped with the Wizard of Oz technique [6] and a staged regular crosswalk. Subjects were grouped in groups of up to three people. In a series of 10 runs, subjects randomly assigned to two groups located on both ends of the smart crosswalk were asked to walk from the north to south end of the crosswalk or vice versa. The data collected were: i) the pedestrians' trajectory at each step, ii) stride speed and iii) target accuracy.

Study observations

Overall, studies found that people walking on smart crosswalks have smaller trajectory deviations and higher target accuracy than people walking on regular crosswalks. However the walking flow of people on smart crosswalks slowed down. It appears that there was an inverse correlation between the trajectory disturbances and the walking speed. In other words, in order to walk fast pedestrians needed to sort out disturbances. Such disturbances were represented by the presence of other human actors enacting their own programs-of–action. The general observation is that pedestrians hinder the execution of each other's programs-of-action forcing themselves to constantly adapt or overwrite their original programs-of-action.

Analysis of observations and results

The following analysis applies the triadic model described above to the interaction of pedestrians mediated by the smart crosswalk. The two human actors of the triad are the pedestrian or group of pedestrians heading north (PHN) and the pedestrian or group of pedestrians heading south (PHS). These two actors are subscribed to the smart crosswalk as an instance of a nonhuman actor. The network of actors has two triads: *{PHN – smart crosswalk}* → *PHS* and *{PHS – smart crosswalk}* → *PHN.* The programs-of-action of both human and nonhuman actors in the network are presented in Table 2.

The within interaction of the collective *{PHN – smart crosswalk}* holds these two actors together, co-shaping the mediating meaning of a hybrid signifier. Such signifier is composed by the pattern signaled by the crosswalk and the actions of the pedestrians heading north on the smart crosswalk. The PHS actor interprets the signifier and adapts its actions accordingly. The between interaction of the triad can be observed in the dynamic negotiation of trajectories carried out by both groups of pedestrians circumventing potential collisions. Conversely, the complementary triad *{PHS – smart crosswalk}* → *PHN* has the same within and between interactions. Such networked triads constitute an adaptive system in which the modification of one actor's program-of-action affects the enaction of others' programs-of-action.

Table 2. Programs-of-action of pedestrians and smart crosswalk in the proof of concept study

Agent	Type of program-of-action	Description of program-of-action
Smart crosswalk	Behavioral	Afford pedestrians crossing from one end to the opposite
	Proactive	Either highlight potential conflicts or suggest trajectory deviations
Pedestrians heading north or south	Behavioral	Avoid collisions while walking
	Proactive	Walk to his/her/their destination preserving their clique's cohesiveness

The observations of the walking flow in both regular and smart crosswalks show that the within and between interactions have a double-edged effect in the actor-network. While the within interactions pull actors together, the between interactions offer resistance to the execution of the human actors' programs-of-action. As a result, people cooperate when they have conflicting programs-of-action or collaborate when they have aligned programs-of-action. Both cooperation and collaboration require that people coordinate their actions.

While smaller collectives coordinate easily, larger ones struggle to maintain coordination. The high trajectory disturbance observed in the study reveals the actor's friction enacting their programs-of-action. Such friction, which ultimately renders the actor-network *viscous*, seems to thicken when people act under limited access to environmental information. It is under such limited conditions when actions of smart artifacts have higher impact in the actor-network's viscosity and benefit communal action flow across the actors in the network. This research defines *social viscosity* as the natural resistance of an actor-network to the fluidity of its actors' actions caused by the mutual disturbances elicited while they enact their programs-of-action.

While well-coordinated action reduces actors' mutual disturbances, the process of achieving such coordination hinders the fluidity of actors' actions. The empirical studies show that the mediation of social interaction by means of smart artifact mediators improved human actors' degrees of coordination if such mediation i) prompts the preservation of social balance by enacting the dominant socio-relational principles, ii) enhances actor's information about the whole actor-network, and iii) is present at the focus of the social activity.

CONCLUSION

The articulation of Actor-Network Theory principles with interaction design methods opens up the traditional user-artifact dyad towards triadic collective enactments by embracing diverse kinds of participants and practices, thus facilitating the design of enhanced sociality.

Smart artifacts that put forward forecasted conflicts between networked human actors are prone to facilitate either kind of social interaction: cooperation or collaboration. Cooperation and collaboration are two types of social interaction akin to balanced forms of sociality.

Smart artifacts can be designed not only as tools that allow people to accomplish their tasks, but also as relational objects that step into social activity by suggesting actions that may benefit the whole community. As the example *{pedestrian – smart crosswalk} → pedestrian* shows, smart artifacts can act as signifiers of the social activity of a group of people and mediate forms of coordination between them. Cooperation is only one type of social action, however, the position offered here could be extended to other types of social action such as collaboration, conflict resolution or adhesion.

The design of socially apt smart artifacts demands that designers decompose social action by identifying the programs-of-action of all the interacting parties. The position discussed in this paper suggests a new role for smart artifact designers: the delineation of artifact's programs-of-action. By identifying potential triadic structures in the network of actors, and analyzing how action unfolds in each triad, designers can refine the social responsiveness of smart artifacts rendering them more socially apt.

Finally, social viscosity is the natural resistance of an actor-network to the fluidity of its actors' actions. It has a direct correlation to the size and density of the network.

REFERENCES

1. Agre, P. *Human-computer interaction*. Lawrence Erlbaum Associates, Hillsdale, N.J., (2001), 177-192.
2. Breazeal, C.L. *Designing sociable robots*. MIT Press, Cambridge, Mass, 2002.
3. Bullington, J. Agents and Social Interaction: Insights from Social Psychology. In: Trajkovski, G., Collins, S.G. (Eds.), *Handbook of research on agent-based societies : social and cultural interactions*. Information Science Reference, Hershey, PA, (2009), 35-80.
4. Callon, M., Law, J. *Agency and the hybrid collectif.* South Atlantic Quarterly 94, (1995), 481.
5. Cassell, J.B.T. Negotiated Collusion: Modeling Social Language and its Relationship Effects in *Intelligent Agents. User Modeling and User-Adapted Interaction* 13, (2003), 89-132.
6. Dahlbäck, N., Jönsson, A., Ahrenberg, L. Wizard of Oz studies: why and how. *Proc. IUI 93*. ACM, Orlando, Fl, United States, (1993), 193-200.
7. Epstein, J.M., Axtell, R. *Growing artificial societies : social science from the bottom up*. Brookings Institution Press, Washington, D.C. 1996.
8. Fiske, A.P. Relational Models Theory 2.0. In: Haslam, N. (Ed.), *Relational Models Theory, A Contemporary Review*. Lawrence Erlbaum Associates, Inc, Mahwah, NJ, (2004), 3-24.
9. Hutchins, E. *Cognition in the wild*. MIT Press, Cambridge, Mass. 1995.
10. Kaptelinin, V., Nardi, B.A. *Acting with technology : activity theory and interaction design*. MIT Press, Cambridge, Mass. 2006.
11. Knorr-Cetina, K. Sociality with Objects. *Theory, Culture & Society*, (1997), 1-30.
12. Latour, B. Where are the Missing Masses? A Sociology of Few Mundane Objects. In: Bijker, W.E., Law, J. (Eds.), *Shaping Technology/Building Society. Studies in Sociotechnical Change*. MIT Press, Cambridge, Mass., (1992), 151-180.
13. Latour, B. *Reassembling the social : an introduction to actor-network-theory*. Oxford University Press, Oxford ; New York. 2005.
14. Law, J. *Notes on the Theory of the Actor Network: Ordering, Strategy and Heterogeneity*. Centre for Science Studies, Lancaster University, Lancaster. 1992.
15. North, M.J., Macal, C.M. *Managing Business Complexity. Discovering Strategic Solutions with Agent-Based Modeling and Simulation*. Oxford University Press, New York. 2007.
16. Pawlak, Z. Rough set theory and its applications. *Journal of Telecommunications and Information Technology* 3, (2003), 7-10.
17. Picard, R.W. *Affective computing*. Perceptual Computing Section, Media Laboratory, Massachusetts Institute of Technology, Cambridge, Mass. 1995.
18. Schutz, A. *The phenomenology of the social world*. Northwestern University Press, Evanston, Ill. 1972.
19. Shackel, B. Human-computer interaction - Whence and whither? *Interacting with Computers,* 21, 5-6, (2009), 353 – 366.

Fast and Comprehensive Extension to Intention Prediction from Gaze

Hana Vrzakova
University of Eastern Finland
School of Computing
hana.vrzakova@uef.fi

Roman Bednarik
University of Eastern Finland
School of Computing
roman.bednarik@uef.fi

ABSTRACT

Every interaction starts with an intention to interact. The capability to predict user intentions is a primary challenge in building smart intelligent interaction. We push the boundaries of state-of-the-art of inferential intention prediction from eye-movement data. We simplified the model training procedure and experimentally showed that removing the post-event fixation does not significantly affect the classification performance. Our extended method both decreases the response time and computational load.

Using the proposed method, we compared full feature sets to reduced sets, and we validated the new approach on a complementary set from another interaction modality. Future intelligent interfaces will benefit from faster online feedback and decreased computational demands.

Author Keywords

Intentions; Prediction; Eye-tracking; SVM; Gaze-augmented interaction; Dwell-time

ACM Classification Keywords

H.5.m. Information Interfaces and Presentation (e.g. HCI): Miscellaneous

INTRODUCTION

Prediction of user interests and intention to interact is the primary task of user interface designers. Best UI designs are those that tap into users' preferences and provide a seamless interaction where the interface *'knows'* what are the intentions of the user at any time. While anticipating future interactions, designers can impersonate a typical user, can try to estimate his model in head, gain understanding of the needs, and express that in terms of the design of the interface that matches the interaction model of the user. If they succeed, the interface is perceived as natural, user friendly, responsive, immersive and intuitive, to name few.

An everyday experience unfortunately indicates that such user interfaces are rare. One reason for it is that the designers fail to engineer the proper interaction model and because of the mismatches between the current user perception of the system and the real state of the system. If a user interface can effectively predict that a user wants to interact in a certain way even though the current state of the system does not expect such to happen, interaction errors can be avoided. For instance, misplaced interactions, such as expecting to type in an input box but not having the cursor at the input box, can be efficiently corrected when an intention to type can be predicted early enough.

All interactive actions begin with an intention to interact. Specifically, the formation of the intention to explicitly interact is a stage preliminary to interaction [13]. For example, to press a button a user has to first formulate an intention to interact and then execute the hand movement and finger flex to issue the button press. In this paper we deal with the deep detailed level of interaction in terms of predicting the intentions to interaction.

Eye tracking as source for user modeling

Eye-tracking data can be used to discover user's cognitive state [8, 14], workload [2, 1], expertise [9] or to predict the context of interaction [11, 6]. Eye-tracking is also expected to become a ubiquitous interaction technique [12, 5]. If eye-tracking is indeed going to be a standard source of user data, the implicit behavioral information can be used for modeling of user states.

Previous work on intention prediction has shown that employing eye-movements and machine learning is a feasible modeling technique to achieve good levels of prediction in human-computer interaction. Bednarik et al. formulated a machine learning pipeline for eye-tracking data that performs training of a classifier to detect whether a user, engaged in gaze-augmented interactive problem solving, aims to change the state of the problem using a button press [4]. Using their framework, they achieved a classification accuracy of 76% (AUC = 0.81). Thus, intention prediction is achievable with levels far above the level of chance, although the total training time reported was over 180 hours.

In this paper, we report on a systematic study to advance the state-of-the-art of automatic inferential intention prediction for HCI. We 1) employ another modality to evaluate the generalizability of the pipeline, 2) present effects of simplification of the training, 3) investigate new options of feature extraction, and 4) compare the performance of the feature sets of the state-of-the-art system with performance based on reduced feature sets.

IUI 2013 Workshop: Interacting with Smart Objects, March 18, 2013, Santa Monica,CA, USA

In particular, we compare performance differences in intention prediction for gaze-augmented and traditional mouse-based interaction. With an objective to significantly reduce training time, we designed a less comprehensive training and evaluate the effects of the simplification on the performance.

The original study has employed *fixational sequences* centered around the observed intention. It implies that in real-time implementations the prediction component would be able to report on a predicted intention with some delay. The underlying question, however, concerns maximalizing the chance of discovering intentions from short fixational sequences. Optimally, we would wish to be able to predict an incoming interaction *before* it happens. Therefore, we systematically shift the extracted sequences before and after the interactive action and compare the effects on the prediction performance.

Finally, we perform a standard feature selection to reduce available feature space, by analyzing inter-feature correlations.

METHOD

Training dataset: Interactive problem solving

The datasets that we employ in this research were originally collected for another purposes and has been described in [3]; the original study examined the effects of interaction modality on problem-solving activities.

Figure 3 presents a studied user interface from the original study. The task of the user was to arrange a shuffled set of tiles into a required order. As users engaged in the problem solving through various interaction modalities, the original studies have discovered that gaze-augmented interaction is superior over the traditional mouse-based interaction.

The data were collected in a quiet usability laboratory. The eye movements of the participants were collected using a Tobii ET1750 eye-tracker, sampling at 50Hz. Each participant interacted with the interface individually and participants were required to think aloud. There were altogether three sessions from which data has been collected.

Here we use two datasets from those interactions: first, the same gaze-augmented interaction dataset as employed in the benchmark report by Bednarik et al. [4]. Second, the difference here is that the evaluation of the new method employs also a dataset containing mouse-controlled interactions. Thus, in the first dataset gaze is used in a bidirectional way, while in the mouse-based problem-solving gaze is used only for perception of the problem-space.

The button press in both conditions sets the boundaries for the fixational sequence extraction. The corresponding sequence of eye tracking data is related to this event. The sizes of the extracted datasets are shown in Table 1.

Extension of prediction method

The experiments in this study take as a baseline the prediction framework from [4]. The prediction framework performs detection of intentions using fixational sequences wrapped around the interaction event. It employs numerous eye-tracking features computed from the sequences, and cross-validation for prediction model training. A parameter grid search is employed and Support Vector Machine is used as a classifier.

Table 1. Dataset distributions according to interaction style

Type of interaction	Intent [n]	Non-Intent [n]	Total [n]
Gaze Augmented	2497	22119	24616
	10.14%	89.86%	100%
Mouse only	2823	18714	21537
	13.11%	86.89%	100%

In this work, we propose two extensions to the previous work. Figure 1 illustrates the prediction pipeline and introduces the main contributions of the present work as presented in the following sections.

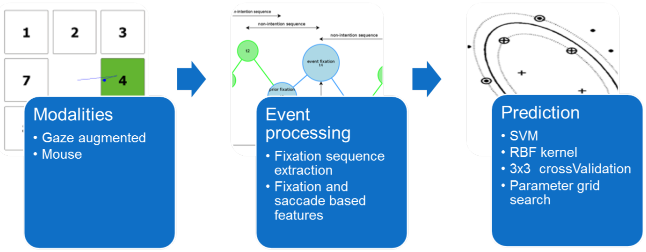


Figure 1. Extensions in prediction pipeline.

Disregarding future information

The first modification concerns the extraction of the fixational sequences. The original work focused on wrapping the feature extraction around the so-called *event* fixation: whole sequence consisted of the event fixation reflecting the interaction event, one fixation before and one fixation after the event.

Here we introduce *pre-event* and *post-event* fixations. Using this scheme, illustrated in Figure 2, we created three datasets: one consisting of sequences composed from two pre-event and one event fixations (denoted hereafter by '2+1+0'), one consisting of pre-event, event, and post-event fixations (1+1+1), and one of one event and two post-event fixations (0+1+2). Such settings, we believe, may reveal contribution of fixations surrounding interaction events.

The second expansion focuses on the type of computed features. We employ fixation and saccade features only and disregard pupil-dilation based features. Although prior research proved a link between pupil dilation and cognitive processes [1], it has also revealed a tangible time delay between cognition and pupillary response [7]. Such delay would deteriorate

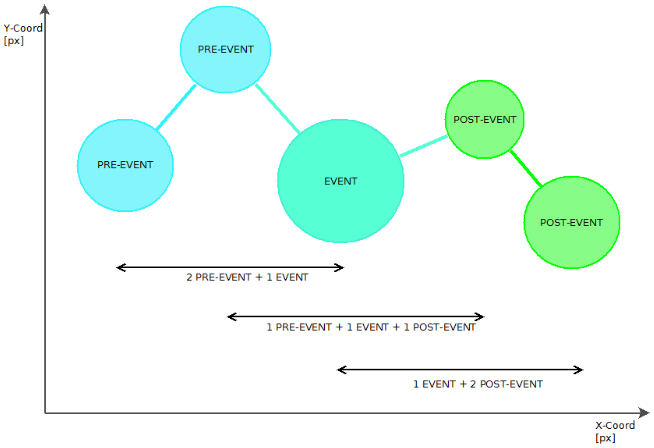


Figure 2. Fixational sequences: Pre-event, event and post-event fixations

the performance of an eventual real-time classifier. The fixation and saccade based features are presented in Tables 2 and 3.

Table 2. Eye movements features computed from fixations. Adopted from[4]

Eye movement feature	Description
Mean fixation duration	The average time of fixation duration in the observed sequence
Total fixation duration	Sum of fixation durations in the observed sequence
Event fixation duration	Duration of the fixation for the ongoing interaction
Prior fixation duration	Duration of the fixation before intention occurrence

Table 3. Eye movements features computed from saccades. Adopted from [4]

Eye movement feature	Description
Mean saccade duration	The average saccade duration in the observed sequence
Total saccade duration	Sum of saccade durations in the observed sequence
Last saccade duration	Duration of the fixation before event occurrence
Mean saccade length	The average distance of saccade in the observed sequence
Total saccade length	Sum of saccade distances in the observed sequence
Last saccade length	Distance of the saccade before event occurrence
Mean saccade velocity	The average speed of saccades in the observed sequence
Last saccade velocity	Speed of the saccade before event occurrence
Mean saccade acceleration	Acceleration of saccade during the observed sequence

Faster training

Third, *simplified* the parameter search in prediction model training by reducing the number of folds in the two nested cross validations from 6 x 6 to 3 x 3. Such settings reduce computational time. More importantly we investigate whether it affects the classifier performance.

Fourth, we created an additional dataset by filtering out correlated features. Such reduced dataset may have comparable or better performance under lower computational costs.

Baseline settings

For comparison purposes, we created a *balanced* dataset of intent and non-intent feature vectors. We used all the intentions and randomly chose a corresponding number of non-intention feature vectors (see Table 1). In real interaction, the proportion of intentions is much lower, however, balanced experimental settings serve for baseline comparison and show the limitations of the weight settings in case of an unbalanced training dataset.

The remaining settings (parameter grid search and SVM kernel) were kept the same as in the prior study [4].

RESULTS

The systematic evaluation, reported here, presents 18 experiments, with a total duration over 135 hours of computational time, which presents reduction around 30% compared to prior study in [4], when using a comparable hardware.

A typical processed sequence of fixations containing an event is shown in blue color in Figure 3.

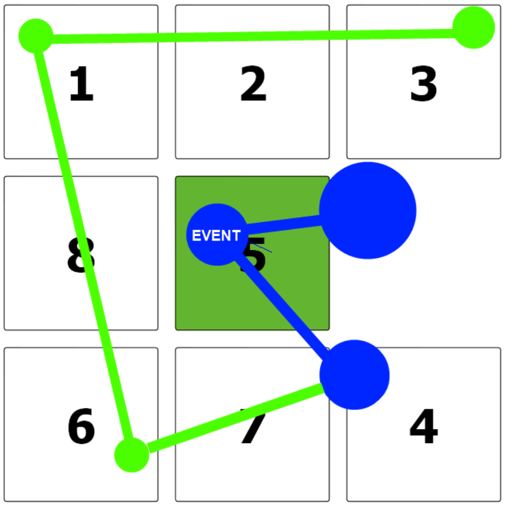


Figure 3. Typical intent (blue) and non-intent (green) fixation sequences. The relationship between features computed from the sequences are shown in Figures 4 and 5

Understanding how much each feature contributes to the class recognition belongs to the features selection problems and lead to another computationally demanding tasks. For estimation of such contribution, Figure 4 and Figure 5 demonstrate a percentage change of averaged features, when compared to the non-intention ones, and how observed interaction style influenced the ratio. A baseline indicates that intention and non-intention feature vectors would be same, positive ratio shows greater mean values of the intention-related features while negative presents the smaller ones.

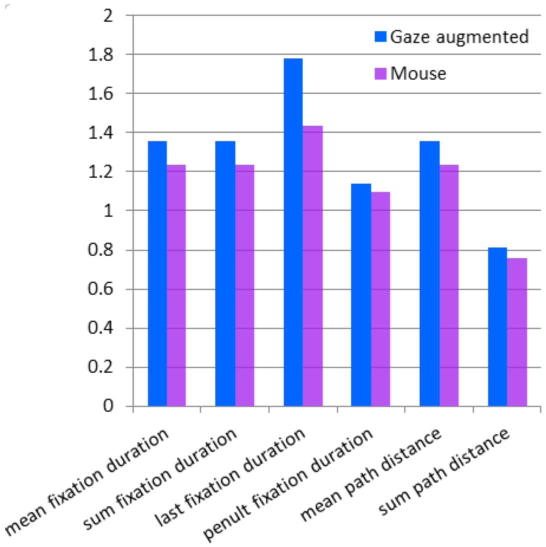


Figure 4. Effects of intentions on fixation based features. During intentional interaction, fixation derived metrics increased compared to baseline (non-intentional interaction). The comparison of interaction styles introduced higher increase in the gaze-augmented interface.

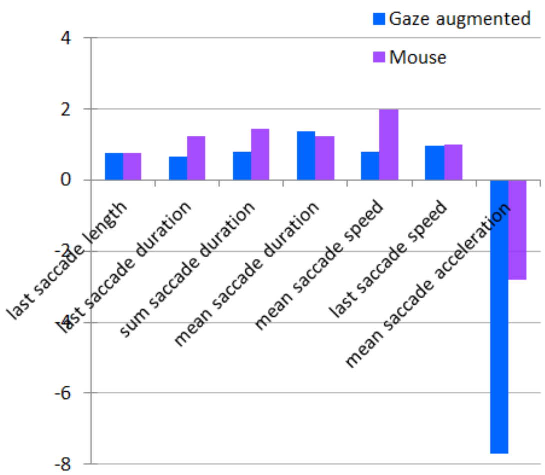


Figure 5. Influence of intentions on saccade based features. Intentions to interact increased metrics of saccade duration and speed and decreased saccade acceleration, when compared to baseline (non-intentions). The mouse interaction style reflects more in the duration and speed features, while acceleration corresponds with the gaze-augmented interaction.

Table 4 shows an overview of all experiments. We report on Area Under the Curve (AUC) as the primary performance measure. In case of the balanced data, also accuracy is a reliable metric of classification performance.

The performance of the classifiers is comparable to the baseline results achieved previously in [4]; best performance on an imbalanced data was AUC of 0.79 which is just 0.02 below the 0.81 reported before. However, the best performance here was achieved using much simpler training procedure. The effect of feature selection was minor, however noticeable.

Effects of Fixation sequence

A comparison of the processing pre-event and post-event fixational sequences showed minor differences in each training group. In gaze-augmented The highest performance was reached up to AUC of 0.8 in gaze augmented interaction (SVM.C = 94.868, SVM.Gamma = 3.684E-7), and AUC of 0.73 in the traditional mouse interface (SVM.C = 30.800, SVM.Gamma = 1.0E-8). Although in several cases of mouse modality AUC resulted better performance for post-event dataset, the better performance in gaze augmented interface was gained using pre-event (2+1+0 and 1+1+1) fixational sequences.

Predictability of intentions across modalities

A comparison of interaction modality showed that intention prediction was better performed using gaze-augmented interaction rather than mouse-based one. The best performance for gaze-augmented interface reached up to AUC 0.8, while the best mouse-based prediction was still 0.07 lower. Such results indicate that interaction intention prediction is tightly dependent on the observed modality, and prediction model needs training for each modality separately.

DISCUSSION

This paper presented two contributions. The main novelty presented here is in showing that interaction intention prediction from gaze does not need to rely on post-event fixations. This finding has important implications, both on the research of understanding of interactions from physiological signals and on the applications and implementations of the inference algorithms in real time.

We reported the cross-validation results of the extended intention prediction pipeline. Here we compared them to the prior baseline study, reported in [4].

Predicting interaction before actions

The findings show that it is not necessary to postpone action detections until post-event data becomes available. This can be considered as a breakthrough result, give the fact that research that employs EEG signals for detection of interaction errors reports the shortest achievable time for a computing system to realize action to be about 150 - 300ms after the user-event [10].

A question arises regarding the information contained in the fixational sequences. Where one should look for a reliable source of information about interaction intention? According to Figures 4 and 5 and the ratios of averaged feature vectors, the two interaction modalities resulted in observable differences between averaged intention and non-intention feature vectors. In other words, the gaze behavior around interactive actions differed across modalities. We observed that gaze-augmented interface affected more the features related to fixation, whereas the interface with the mouse influenced saccade based features. Therefore, the answer to the question seem to depend on the interaction modality in use.

Table 4. Overview of results

Modality	Training	Fixation sequence	AUC	Accuracy	Recall	Precision
Gaze augmented	State of the art. Adapted from [4]	1 + 1 + 1	0.81	0.76	0.69	0.31
Gaze augmented	Simplified	2 + 1 + 0	0.78	0.82	0.54	0.29
		1 + 1 + 1	0.78	0.80	0.57	0.27
		0 + 1 + 2	0.79	0.82	0.57	0.29
	Simplified + Without correlated features	2 + 1 + 0	0.72	0.75	0.53	0.21
		1 + 1 + 1	0.72	0.75	0.54	0.21
		0 + 1 + 2	0.75	0.77	0.54	0.23
	Simplified + Balanced	2 + 1 + 0	0.77	0.71	0.66	0.74
		1 + 1 + 1	0.80	0.73	0.72	0.73
		0 + 1 + 2	0.78	0.72	0.67	0.75
Mouse	Simplified	2 + 1 + 0	0.69	0.70	0.65	0.23
		1 + 1 + 1	0.72	0.68	0.65	0.23
		0 + 1 + 2	0.72	0.67	0.65	0.23
	Simplified + Without correlated features	2 + 1 + 0	0.67	0.58	0.74	0.19
		1 + 1 + 1	0.69	0.62	0.66	0.20
		0 + 1 + 2	0.71	0.64	0.70	0.22
	Simplified + Balanced	2 + 1 + 0	0.70	0.64	0.55	0.67
		1 + 1 + 1	0.71	0.66	0.65	0.66
		0 + 1 + 2	0.73	0.66	0.59	0.69

Reduction of the computational load

The second contribution of this study lies in showing that reducing the comprehensive search for optimal parameters during training is justified by minimal decrease in classification performance. This is a major improvement, because the decreased complexity and costs of the training lead to less computational load. In sum, a less comprehensive search in the feature space does not necessarily imply worse performance of the classification when using SVM classifiers.

Considering the implications of the findings on the real-time implementations, we have virtually removed the need to delay classification decisions till post-even fixations. Thus, an effective inference can be carried out at nearly the real-time of the event. We are currently investigating possibilities to even a further shift – in the sense of employing more data from the past– however, there are new challenges arising. For example, the previous events that are close to the event of interest create overlapping data and thus ground truth labeling is difficult.

Finally, to demonstrate the robustness and generalizability of the new approach, we evaluated its performance on a complementary dataset. Although the features differ because of a different interaction modality, the performance of the intention classification pipeline only decreases by about 5-9% on AUC.

Applications of intention inference

The research presents eye-tracking as a feasible and fast method for intention classification. Although the datasets on which we developed the methods have been captured from a rather traditional WIMP paradigm, we believe that in contexts beyond a computer desktop our approach can as well be applied.

Event though the current wearable eye-trackers do not achieve high sampling rates, and thus the temporal resolution is low to allow accurate identification of fast eye-movements, future technologies will likely be able to overcome this drawback. Then, methods such as ours can be used for detection of user intention to interact with surrounding objects. For a pervasive interaction, not only the objects can be made gaze-aware [15], but can be made even mode intelligent by sensing the nuances of user interaction with them.

CONCLUSION AND FUTURE WORK

The ability to predict user intentions is one of the primary challenges in building smart intelligent interfaces. Our study extends the argument that eye movements can reveal interaction intentions and their relationship to interaction style using intention prediction model.

In comparison to prior research, we lowered computational demands of 30% using balancing dataset, reducing number of folds in cross validations, and removing correlated features. Even though a comparison of classification performance revealed a decreased ability to differentiate between intentions and non-intentions, such approach motivates for further research since the overall classification performance was reduced just in acceptable units of AUC. For future real-time classifications, methods of optimized prediction are more promising than the demanding parameter search in a large feature space.

REFERENCES

1. Bailey, B. P., and Iqbal, S. T. Understanding changes in mental workload during execution of goal-directed tasks and its application for interruption management. *ACM Trans. Comput.-Hum. Interact. 14*, 4 (Jan. 2008), 21:1–21:28.

2. Bartels, M., and Marshall, S. P. Measuring cognitive workload across different eye tracking hardware platforms. In *Proceedings of the Symposium on Eye*

Tracking Research and Applications, ETRA '12, ACM (New York, NY, USA, 2012), 161–164.

3. Bednarik, R., Gowases, T., and Tukiainen, M. Gaze interaction enhances problem solving: Effects of dwell-time based, gaze-augmented, and mouse interaction on problem-solving strategies and user experience. *Journal of Eye Movement Research 3*, 1 (2009), 1–10.

4. Bednarik, R., Vrzakova, H., and Hradis, M. What do you want to do next: a novel approach for intent prediction in gaze-based interaction. In *Proceedings of the Symposium on Eye Tracking Research and Applications*, ETRA '12, ACM (New York, NY, USA, 2012), 83–90.

5. Bulling, A., and Gellersen, H. Toward mobile eye-based human-computer interaction. *Pervasive Computing, IEEE 9*, 4 (2010), 8–12.

6. Bulling, A., Roggen, D., and Troster, G. What's in the eyes for context-awareness? *Pervasive Computing, IEEE 10*, 2 (2011), 48–57.

7. Einhäuser, W., Koch, C., and Carter, O. L. Pupil dilation betrays the timing of decisions. *Frontiers in human neuroscience 4* (2010).

8. Eivazi, S., and Bednarik, R. Inferring problem solving strategies using eye-tracking: system description and evaluation. In *Proceedings of the 10th Koli Calling International Conference on Computing Education Research*, Koli Calling '10, ACM (New York, NY, USA, 2010), 55–61.

9. Eivazi, S., Bednarik, R., Tukiainen, M., von und zu Fraunberg, M., Leinonen, V., and Jääskeläinen, J. E. Gaze behaviour of expert and novice microneurosurgeons differs during observations of tumor removal recordings. In *Proceedings of the Symposium on Eye Tracking Research and Applications*, ETRA '12, ACM (New York, NY, USA, 2012), 377–380.

10. Ferrez, P. W., and Millán, J. D. R. You are wrong!: automatic detection of interaction errors from brain waves. In *Proceedings of the 19th international joint conference on Artificial intelligence*, IJCAI'05, Morgan Kaufmann Publishers Inc. (San Francisco, CA, USA, 2005), 1413–1418.

11. Hradis, M., Eivazi, S., and Bednarik, R. Voice activity detection from gaze in video mediated communication. In *Proceedings of the Symposium on Eye Tracking Research and Applications*, ETRA '12, ACM (New York, NY, USA, 2012), 329–332.

12. Jacob, R. J. K., and Karn, K. S. Commentary on section 4. eye tracking in human-computer interaction and usability research: Ready to deliver the promises. In *The Mind's Eye: Cognitive and Applied Aspects of Eye Movement Research*. Elsevier Science, 2003, 573–605.

13. Norman, D. A. *The Design of Everyday Things*. Basic Books, New York, 2002.

14. Simola, J., Salojärvi, J., and Kojo, I. Using hidden markov model to uncover processing states from eye movements in information search tasks. *Cogn. Syst. Res. 9*, 4 (Oct. 2008), 237–251.

15. Vertegaal, R., and Shell, J. Attentive user interfaces: the surveillance and sousveillance of gaze-aware objects. *Social Science Information 47*, 3 (2008), 275–298.

A Query Refinement Mechanism for Mobile Conversational Search in Smart Environments

Beibei Hu
MAS Laboratory, Ecole Centrale Paris
92295 Chtenay-Malabry Cedex, France
beibei.hu@ecp.fr

Marie-Aude Aufaure
MAS Laboratory, Ecole Centrale Paris
92295 Chtenay-Malabry Cedex, France
marie-aude.aufaure@ecp.fr

ABSTRACT

A key challenge for dialogue systems in smart environments is to provide the most appropriate answer adapted to the user's context-dependent preferences. Most of the current conversational search is inefficient for locating the target choices when user preferences depend on multiple attributes or criteria. In this paper, we propose an architecture which incorporates a context-dependent preference model for representing weighted interests within utility functions, and a query refinement mechanism that can incrementally adapt the recommended items to the current information needs according to user's critiques. Our preliminary evaluation results based on a scenario of interactive search demonstrate that the query refinement mechanism supported by our architecture can enhance the accuracy of mobile search.

Author Keywords

preference modeling; context-awareness; query refinement.

INTRODUCTION

Recent trends in Spoken Dialogue Systems (SDS) are towards personalized and interactive search in smart environments, which can recommend items that may be of interest to mobile users (e.g., tourists) given their target topics, such as hotels and transportation schedules. One of the main challenges is to exploit preferential information and adapt the answers to user's context over interaction cycles. Two crucial subtasks to this end are: (i) modeling user's context-depended preferences by considering not only hard constrains (e.g., the price should not be more expensive than 50 euros), but also soft constrains (e.g., I prefer French food, otherwise Italian food is also fine); and (ii) improving the accuracy of conversational search according to user's critiques made on the current recommended items.

Modeling user preferences plays a major role in the design of adaptive recommendation systems which provide information or services personalized for user's needs. Typically, user preferences, such as "prefers A over B", could be soft constraints which are represented by a list of alternatives, and thus enable to achieve the optimal answers regarding to the available objects, even if there are no exact match. To adapt the answers to user's current context in mobile environments, it is necessary for dialogue systems to exploit user preferences in a given contextual situation. Since user preferences can be expressed via item ratings, especially for those recommender systems based on collaborative filtering, much attention has been focused on assessing and modeling the relationship between contextual factors and item ratings [1]. In this paper, we aim to model user preferences by taking into account the contextual information and assess the preference weights given multiple attributes.

Given that users are likely to make their feedback on the recommended items during conversational search, this style of user-system interaction requires the dialogue systems to provide better answers adapted to user's critiques. Therefore, the mechanism of preference adjustment formed in critiques-based recommender systems is critical for improving the accuracy of recommendations, so as to ensure a natural and intelligent interaction. Based on Multi-Attribute Utility Theory (MAUT) [9], a critique generation method has been presented to assist users in making critiques according to their stated and potentially hidden preferences [2]. Relying on the above work, we focus on the design of a mechanism that can adjust user preferences and accordingly refine the queries based on the critiques.

Most of the current search engines and recommender systems perform well if a user has a single search criterion and does not have multiple trade-offs to explore. However, few of them provides efficient solutions for finding the user's target choice relying on multiple service properties and their values. To tackle this challenge, we have been involved in the European PARLANCE project[1], which aims to design and develop mobile, interactive, "hyper-local" search through speech in a number of languages [5]. In the context of PARLANCE, our research aims to realize preference-enabled querying for mobile conversational search. Our contributions in this paper are two-folds: (i) we present an ontological architecture of preference-enable querying for smart dialogue systems, which allows to represent user preferences by taking into account the contextual information; and (ii) we propose a query refinement mechanism which can incrementally adapt the retrieved items to user's context given the critiques.

IUI 2013 Workshop: Interacting with Smart Objects, March 18, 2013, Santa Monica,CA, USA Copyright is held by the author/owner(s)

[1] `http://sites.google.com/site/parlanceprojectofficial/`

In this paper, we discuss the related work on context awareness, user preference modeling and interaction strategies of dialogue systems. By presenting a motivating scenario, we highlight the requirements that led us to design the architecture. After presenting the overall architecture, we elaborate our preference model and query refinement mechanism. Finally, we demonstrate our preliminary evaluation results and outline the future research.

RELATED WORK

This section overviews some important issues concerned with the design and implementation of context-aware dialogue systems in smart environments.

Context awareness. In the field of Ambient Intelligence (AmI), context-awareness is exploited as a technique for developing applications which are flexible, adaptable, and capable of acting autonomously on behalf of users. The most acknowledged definition of context provided by [4] is: "any information that can be used to characterize the situation of an entity (...) relevant to the interaction between a user and an application, including the user and application themselves". As can be observed from this definition, besides the external context such as location and temporal context, it is necessary to take into account the internal context, include current user state, inferences on user behavior and long-term user properties (such as preferences in interaction style) that are relevant to the interaction between a user and a system [15]. Ontology-based context modeling has advantages in terms of sharing a common understanding of the structure of context information among users, devices as well as services, and reusing the domain knowledge, and also describing contexts at a semantic level.

User preference modeling. Given that the user can be considered to be part of the contextual information, it is essential to employ user models to differentiate user types and adapt the performance of dialogue systems accordingly [11]. The General User Modeling Ontology (GUMO) provides a uniform interpretation of distributed user profiles in intelligent semantic web enriched environments [6]. From the quantitative perspective, preferences are rating and defined as a function μ that captures the satisfaction or appealingness of an item $i \in I$ to user $\mu \in U$ within a scale of numerical values [13], usually the real interval [1, 1], i.e., $\mu : U \times I \rightarrow [-1, 1]$. On the other hand, preferences are also viewed as qualitative descriptions of a set of properties that specify user interests, which can be added to the queries as constraints [8]. A preference model based on MAUT proposed in [3] is represented as: a pair ($\{V_1, ..., V_n\}, \{w_1, , w_n\}$), where V_i is the value function for each attribute A_i, and w_i is the relative importance of A_i. The utility of each product ($\langle a_1, a_2, ..., a_n\rangle$) can be hence calculated as:

$$U(\langle a_1, a_2, ..., a_n\rangle) = \sum_{i=1}^{n} w_i V_i(a_i)$$

Based on the above methods, we are targeted at assessing the preference weights within utility functions and updating the weight values in the user profile.

Interaction strategies of dialogue systems. Dialogue systems in smart environments consist of a series of interconnected computing and sensing devices which surround the user pervasively in his environment and are invisible to him, providing a service that is dynamically adapted to the interaction context [11]. Based on user feedback, dialogue systems conduct preference adjustments in order to modify actual user behavior [10]. For example, a mixed interaction strategy is explored by a context-aware dialogue system named SmartKom [16], so that both the user and the system can take the initiative in the dialogue. In addition, critiquing-based recommendations [12], which elicit critiques that users made on the recommended items, allow users to adjust their preferences during the user-system interaction. However, regardless of the importance of user preference adjustments, it is not a trivial task, particularly in the dynamic environment and mobile applications. This paper adopts the concepts of the above characteristics, and designs a query refinement mechanism, aiming at incrementally enhancing the accuracy of conversational search.

MOTIVATING SCENARIO AND REQUIREMENTS

Consider "Bob Ferris", who is a male student at Ecole Centrale Paris. He is looking for a restaurant that is good for having his birthday dinner. An example of interaction between Bob (U) and his dialogue system (S) can be shown as follow:

(1) U: Where can I have my birthday dinner with 5 to 6 friends tonight?
(2) S: Well, I suggest the medium size restaurants near to your place. I have found some French, Chinese and also Italian restaurants. Which cuisine do you prefer?
(3) U: I prefer French food, otherwise Italian food is also fine.
(4) S: I have found 10 French restaurants. The closest one is "Le Bon Plat", next to the Secaux station. The price is medium, costing 35 euro per person. Do you like it?
(5) U: Sounds good, but is there any cheaper one?
(6) S: Yes, a cheaper French restaurant "Le Gourmet" costing 25 euro per person is at the Antony center. But you need to take the bus 379 for 3 stops.
(7) U: Oh, no. A bit far, I want to a closer one.
(8) S: In this case, there is no suitable French restaurant. How about an Italian restaurant? "Bel Cibo" is on St. Velpeau, two kilometers far from your place. It costs 28 euro per person.
(9) U: Nice, l will choose this one.

As the above interactions show, a smart dialogue system should have the following features:

1. The systems can select interesting objects for users by considering their context-dependent preferences. For example, since "Bob" has a social activity (i.e., birthday dinner), the system infers that fast food is not his current preference. Moreover, user preferences are interpreted as both hard constraints (e.g., price less than 35 euro) and soft constraints (e.g., preferring French food to Italian food).

2. The systems is able to explore multi-attribute tradeoffs for assisting users in locating their target objects. In case that there is no answer can meet user's all desired attributes

(e.g., cheaper and nearer), the system can provide a list of suboptimal alternatives and further adjust their ranking.

3. The systems should have a refinement mechanism, allowing to improve the accuracy of recommended items according to user's critiques. As the scenario shows, the preferences are adjusted and queries are accordingly refined over interactive dialogue turns.

ARCHITECTURE OF PREFERENCE-ENABLED QUERYING FOR SMART DIALOGUE SYSTEMS

Based on the above derived requirements, we designed an architecture of preference-enabled querying for smart dialogue systems (PEQSDS). As shown in Figure 1. The main components and their interactions are elaborated below:

Ontological knowledge base. An ontology-based knowledge base is constructed to represent the background knowledge. It consists of the geographic knowledge exploited from DBpedia for representing location information and also the domain-specific ontologies, e.g., a tourism ontology that can capture the concepts of point of interests. These background knowledge is exploited by the query refinement mechanism to enrich the user preferences.

Context-dependent preference model. In order to represent preferences in a given contextual situation, go beyond the physical context like current location that can be measured by hardware sensors, we also consider the logical context, such as user's activities. For example, a user typically prefers to drive highways for commuting (*activities & time*), but he wants to drive through country roads to go to a supermarket during the weekend. The relations between those context dimensions are represented by the properties in the Resource Description Framework (RDF) schema[2], such as a user *performs* an activity, so that the system can decide the appropriate information (e.g., navigation information) that is adapted to the user's current context. We rely on RDF as data model to formalize information as it facilitates the integration of multiple data sources and the representation of information in a flexible way.

Query refinement mechanism. The user's requests are firstly processed by the *dialogue management* component relying on the technologies of speech recognition, while designing this component is out of the scope of this paper. After receiving the processed requests, the query refinement mechanism is triggered to generate corresponding queries that encodes both hard and soft preference constraints. The initial query set is further enriched and refined according to the user's critiques until the retrieved items can meet the information needs.

User preference model

Semantic Web vocabularies such as the Friend-Of-A-Friend (FOAF)[3] and the Weighted Interests Vocabulary[4], facilitate usage and integration of user profiles in different applications.

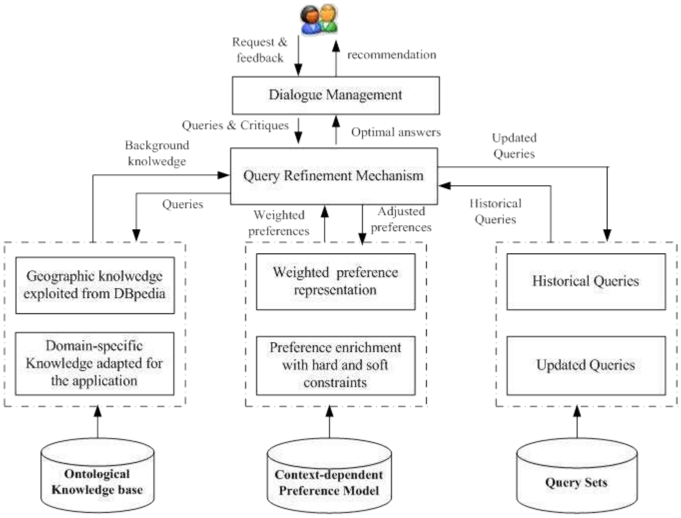


Figure 1. Architecture of PEQSDS

We make use of the standard ontologies and further describe the weighted preferences within utility functions.

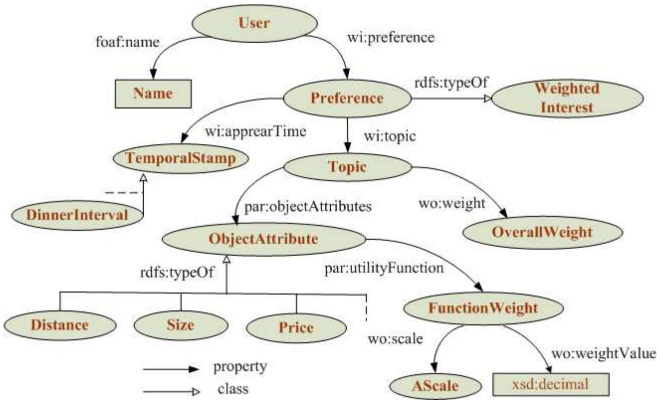


Figure 2. A User Preference Model

As Figure 2 shows, the concept preference has the subject topic and the temporal stamp. The former concept specifies preferred topic in a given domain, while the latter one describes the temporal dynamics of preferences. In particular, the object Topic has two properties: the property *overallWeight* is re-used from the Weighted Interests Vocabulary, while the property *objectAttribute* is defined by us to represent multiple attributes with utility functions. Further, the *objectAttribute* has a property *utilityFunction*, which allows to assign a function weight for each attribute. Thus, a quantitative model of preferences is constructed to compare all alternatives within a topic.

Query Refinement Mechanism

Our algorithm of query refinement mechanism can be described in three main steps: generating initial queries according to user preferences that are enriched by exploiting the knowledge base; adjusting user preferences according to the critiques and further assessing the preference weights based on the MAUT; and refining the queries with respect to the

[2] `www.w3.org/TR/rdf-schema/`

[3] `http://xmlns.com/foaf/spec/`

[4] `http://smiy.sourceforge.net/wi/spec/weightedinterests.html`

adjusted preferences until the retrieved items meet user's information needs. In the following, we explain the sub-steps of our algorithm regarding how it generates a list of possible relevant items and uses the top candidate to learn user's criticisms, and how it adjusts the preference weights and further adapts the retrieved items to user preferences.

1. Query generation. To translate a user's request to an appropriate query, the preferences are enriched by exploiting our knowledge base, and an initial query set can be generated by reusing the SPARQL[5] *query generator* we developed for our Relevance Assessment Rule Engine (RARE) [7]. For example, an initial request in our scenario can be expressed as: *selecting near restaurants at medium prices for Bob having his birthday diner*. By taking into account the contextual information, including user's current location, target event and also preferred cuisine recorded in the user profile, a SPARQL query can be generated as:

```
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix wi:   <http://purl.org/ontology/wi/core#> .
@prefix par: <http://mas.ecp.fr/parlance/rdf/> .

SELECT ?user ?item
WHERE {
          ?user foaf:name "Bob" .
          ?user par:locatedIn ?curLoc .
          ?user par:isInvolvedIn   ?event  .
          ?event par:socialEvent  par:BirthdayDinner .
          ?item   par:location  ?loc.
          ?loc     par:Nearby ?curLoc .
          ?item   wi:topic  par:Restaurant .
          ?item   par:Size  par:Medium .
          ?item   par:Price  par:Moderately .
          ?item   par:Name ?name .
          {?item  par:foodType par:ItalianFood} UNION
          {?item  par:foodType par:FrenchFood} UNION
          {?item  par:foodType par:ChineseFood} .
          }
               ORDER BY ?name
```

2. Preference adjustment. According to the preference-based organization (Pref-ORG) algorithm proposed in [2], each alternative item will be turned into a tradeoff vector (i.e., critique pattern) comprising a set of (*attribute, tradeoff*) pairs. The tradeoff indicates whether the attribute of an item is improved or compromised compared to the same attribute of the top candidate. Enlighten by the above critique generation approach based on the *Apriori* algorithm, we explore the user's stated critiques not only to adjust the preference order, but also to determine the tradeoffs between attributes, so that the user preferences about the relative importance of each attribute can be reflected. For example, in dialogue turn (3), a preference order regarding the cuisine is adjusted as: $Preference_{foodType} = French \succ Italian$; and in turn (5), the price constraint is modified as: $Preference_{price} = price < Medium$; and also in turn (8), the relative importance of attributes are adjusted as: $priceRange_{weight} \succ distance_{weight} \succ foodType_{weight}$.

3. Query refinement. According to the adjusted preferences, the hard as well as soft preference constraints encoded in the queries are refined. The query with negation expressions can be encoded to filter out irrelevant results. For example, in dialogue turn (3), the expression $MINUS\{?cuisine = res : ChineseFood\}$ is to meet to the constraint of cuisine preference, and in turn (5) the expression $FILTER(?price < 35)$ is to meet the constraint of price preference. We also implemented our algorithm of *preference weights updating* that uses the CONSTRUCT query to infer new triples for updating the weight values. In order to collect the items with higher preference weights, an *ORDER BY DESC (?overallWeight)* clause is specified to sort the results in descending order given the overall weight values.

IMPLEMENTATION

This section explains how the preference weights can be assessed and further updated in the user profile.

Weight assessment for user preferences

We present the user preferences over all preferable attributes within utility functions relying on the MAUT. MAUT uses utility functions to convert numerical attribute scales to utility unit scales, thus allowing direct comparison of diverse measures. Applying this approach involves the following steps, which are described below: 1) normalizing attribute scores in terms of each attribute's Best-to-Worst range, and defining the utility function of the ith attribute ($U_i(x_i)$); 2) specifying the tradeoffs between attributes to reflect user preferences about the relative importance of each attribute and defining k_i as the weight of the ith attribute; and 3) for a consequence set that has values $x_1, x_2, ..., x_m$ on the attributes of m objectives, its overall utility is computed as: $U(x_1, x_2, ..., x_m) = k_1U_1(x_1) + k_2U_2(x_2) + ... + k_mU_m(x_m) = \sum_{i=1}^{m} k_iU_i(x_i)$, where $(k_1 + k_2 + ... + k_m = 1)$, and $0 \leq U_i(x_i) \leq 1$, and $0 \leq U(x_1, x_2, ..., x_m) \leq 1$.

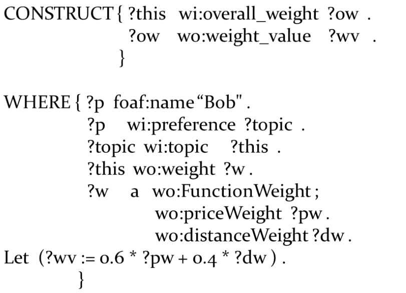
We illustrate how the preference weights can be assessed by applying the above approach in our scenario. Regarding to the three alternatives and their attributes selected in a specific interaction cycle, i.e., a French restaurant named "Le Bon-Plat" (35 euro per person; 1 kilometer distance), a French restaurant named "Le Gourmet" (25 euro per person; 3 kilometer distance) , and an Italian one "Bel Cibo" (28 euro per person; 2 kilometer distance), we firstly set the best value (1) and the worst value (0) by comparing their property values as: $U_{price}(BonPlat) = U_{price}(35) = 0, U_{price}(Gourmet) = U_{price}(25) = 1; U_{distance}(Gourmet) = U_{distance}(3) = 0, U_{distance}(BonPlat) = U_{distance}(1) = 1$. Then, according to the following formula: $U(x) = \frac{x - x_i^-}{x_i^+ - x_i^-}$, where x_i^- denotes the worst value of X_i, and x_i^+ denotes the best value of X_i, utility functions can be defined as: $U_{price}(BelCibo) = 0.6, U_{distance}(BelCibo) = 0.5$. Further, the ratio of the weights is specified according to user's critiques, e.g., $K_{distance} = 2/3K_{price}$. Finally, the overall weight can be assessed as: $U(Bon) = 3/5 \times 0 + 2/5 \times 1 = 0.4; U(Gourmet) = 3/5 \times 1 + 2/5 \times 0 = 0.6; U(Cibo) = 3/5 \times 3/5 + 2/5 \times 1/2 = 0.56$. It can be seen that in this

[5] www.w3.org/TR/rdf-sparql-query/

interaction cycle “Le Gourmet” with the highest weight can be recommended.

SPIN for calculations

We defined SPARQL rules in order to calculate the overall weights and also update the values in the preference model. The SPARQL Inferencing Notation (SPIN)[6] provides the syntax to attach SPARQL queries to resources in an RDF-compliant way using RDF properties spin:rule and spin:constraint. The spin:rule property accepts SPARQL CONSTRUCT queries as value and can be used to infer new triples on the basis of the statements in the query’s WHERE clause [14]. SPIN adoption is supported by tools as TopBraid composer[7]. SPIN allows us to define a function: *calOverallWeight (value, functionWeight, overallWeight):float*, for calculating the preference weight of a given item. As Figure 3 shows, the CONSTRUCT clause infers new triples that represent the updated value of the preference weight, and the LET clause specifies how the value is computed by assigning the ratio of weight.



```
CONSTRUCT { ?this   wi:overall_weight  ?ow  .
             ?ow    wo:weight_value    ?wv  .
            }

WHERE { ?p  foaf:name“Bob" .
        ?p    wi:preference  ?topic  .
        ?topic  wi:topic    ?this  .
        ?this  wo:weight  ?w .
        ?w    a   wo:FunctionWeight ;
                  wo:priceWeight  ?pw .
                  wo:distanceWeight ?dw .
Let  (?wv := 0.6 * ?pw + 0.4 * ?dw ) .
        }
```

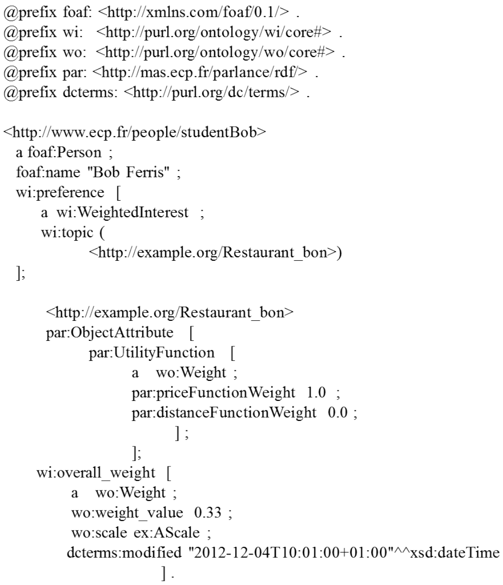
Figure 3. A SPIN Example

PRELIMINARY EVALUATION

We preliminarily evaluated our architecture PEQSDS, in particular the query refinement mechanism based on the data constructed from our scenario.

Dataset construction. To collect a list of items that the user “Bob” may be interested in, we exploited a web search engine, namely Paris Digest[8], which is acted as a city guide for tourism and can recommend points of interests such as restaurants and hotels. By specifying the constraints of location, price, date and cuisine, 22 restaurants were recommended by the search engine. In our evaluation setting, those items were used for re-ranking relying on the query refinement mechanism. We also established a preference model that can specify how much a user is interested into a certain topic. The profile snippet shown in Figure 4 expresses that the user “Bob” is interested into an restaurant named “Bon”. For this interest, the utility functions of price and distance are 1.0 and 0.0, separately; and the preference weight is 0.33.

Evaluation measures. We measured the precision of retrieved results in each interaction cycle, in order to assess how likely user’s target choice could have been located in the recommended items once the query has been refined. The precision denotes the ratio of correctly retrieved items over all retrieved items and can be defined as the following formula: $Precision = \frac{|\{retrieved\ items\} \cap \{user\ target\ items\}|}{|\{retrieved\ items\}|}$. Further, the interaction effort reduction, which is used to assess how effectively the system could potential reduce users’ objective effort in locating their target choice [2], is defined as: $EffortReduction = \frac{1}{NumUsers}(\sum_{i=1}^{NumUsers} \frac{actualCycle-targetCycle}{actualCycle})$, where $actualCycle$ denotes the number of cycles a user actually experienced and $targetCycle$ denotes the number of cycle until user’s target choice first appeared in the recommended items. In addition, a baseline is defined as: choosing the candidates by specifying a single search criterion (i.e., choosing the restaurants for the user within 2 kilometers). We compared the results of the refined queries to those of the baseline.



```
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix wi:  <http://purl.org/ontology/wi/core#> .
@prefix wo:  <http://purl.org/ontology/wo/core#> .
@prefix par: <http://mas.ecp.fr/parlance/rdf/> .
@prefix dcterms: <http://purl.org/dc/terms/> .

<http://www.ecp.fr/people/studentBob>
  a foaf:Person ;
  foaf:name "Bob Ferris" ;
  wi:preference  [
      a  wi:WeightedInterest  ;
      wi:topic (
            <http://example.org/Restaurant_bon>)
  ];

       <http://example.org/Restaurant_bon>
       par:ObjectAttribute   [
             par:UtilityFunction   [
                   a   wo:Weight ;
                   par:priceFunctionWeight  1.0  ;
                   par:distanceFunctionWeight  0.0 ;
                         ] ;
                   ];
     wi:overall_weight  [
          a   wo:Weight ;
          wo:weight_value  0.33 ;
          wo:scale ex:AScale ;
         dcterms:modified "2012-12-04T10:01:00+01:00"^^xsd:dateTime
                      ] .
```

Figure 4. A Part of Preference Model in the Scenario

Results. In the beginning of the experiment, among a set of n potential interesting items ($n = 22$), after refining the preference order on the cuisine, two items were filtered out and the remained 20 items were sorted by the distance. Then, the top candidate was used to collect user’s critiques (e.g., cheaper). In the second interaction cycle, 3 alternatives were retrieved after modifying the preference constraint on the price range. Accordingly, the precision is enhanced to 0.33, compared to the precision achieved by the baseline ($Precision_{baslineline} = 0.25$). By executing SPARQL rules to assess the preference weight, the overall utility was recalculated for each alternative, and the corresponding queries were executed to rank the alternatives by the updated weight values. In the third cycle, the target choice was ranked as the second candidate. This result requests another interaction cycle to re-compute the preference weights by adjusting the

[6] http://spinrdf.org/

[7] http://www.topquadrant.com

[8] http://www.parisdigest.com

ratio of the weight. Finally, the target choice appears in the fourth cycle ($EffortReduction = 0.25$). The above preliminary results show that our approach can satisfy multiple search criteria and enable end-users to more efficiently target their best choices. However, the limitation is that the target choice was not ranked as the top candidate after re-computing the tradeoff values between attributes. In the next step, our approach should be improved by computing the preference weights as accuracy as user's critiques.

CONCLUSIONS AND FUTURE WORK

In this paper, we presented an ontological architecture of preference-enabled querying for smart dialogue systems (PEQSDS). Our architecture is able to enrich user preferences by exploiting an ontological knowledge base and represent the weighted interests within utility functions relying on a preference model. Also, it provides a query refinement mechanism for adapting the retrieved results to user's context. Contrast to the most of existing approaches that are too restricted to a single search criterion, our approach allows for selecting relevant objects by considering both hard and soft preference constraints. Moreover, it is able to adjust user preferences and further refine the queries by learning the user's critiques. Our preliminary evaluation based on a scenario of mobile conversational search shows that the query refinement mechanism offered by PEQSDS can incrementally improve the accuracy of recommendations.

We currently investigate the enhancements to our approach by applying critique generation and association rule mining techniques. We will show how the tradeoffs between desirable attributes can be determined according to the critiques. Relying on our ontological knowledge base, we will further demonstrate how our system is adaptable for dynamic contextual situations. We will also implement SPARQL rules and improve our algorithm to enhance the ability of computing the preference weights. Our future work mainly focuses on conducting evaluation studies based on large and realistic datasets to show the feasibility of our approach in a pervasive environment. We plan to exploit user's profile information and the search history provided by YAHOO! Local, and further evaluate the system behavior in terms of ranked-biased precision and also the interaction effort reduction.

ACKNOWLEDGMENTS

This work has been supported by the European FP7 PARLANCE project. We gratefully acknowledge valuable comments from Dr. Pearl Pu.

REFERENCES

1. Baltrunas, L., Ludwig, B., Peer, S., and Ricci, F. Context relevance assessment and exploitation in mobile recommender systems. *Personal and Ubiquitous Computing 16*, 5 (2012), 507–526.
2. Chen, L., and Pu, P. Experiments on the preference-based organization interface in recommender systems. *ACM Transactions on Computer-Human Interaction (TOCHI) 17*, 1 (2010), 5:1–5:33.
3. Chen, L., and Pu, P. Critiquing-based recommenders: survey and emerging trends. *User Modeling and User-Adapted Interaction 22* (2012), 125–150.
4. Dey, A. K. Understanding and using context. *Personal Ubiquitous Computing 5*, 1 (2001), 4–7.
5. Hastie, H., Lemon, O., and Dethlefs, N. Incremental spoken dialogue systems: Tools and data. *SDCTD 2012* (2012), 15–16.
6. Heckmann, D., Schwartz, T., Brandherm, B., Schmitz, M., and von Wilamowitz-Moellendorff, M. Gumo–the general user model ontology. *User modeling 2005 3538* (2005), 428–432.
7. Hu, B. *Towards Contextualized Information Delivery: A Rule-based Architecture for the Domain of Mobile Police Work*. PhD thesis, Delft University of Technology, Delft, The Netherlands, 2011.
8. Karger, P., Olmedilla, D., Abel, F., Herder, E., and Siberski, W. What do you prefer? using preferences to enhance learning technology. *Learning Technologies, IEEE Transactions on 1*, 1 (2008), 20–33.
9. Keeney, R., and Raiffa, H. *Decisions with multiple objectives: preferences and value trade-offs*. Cambridge University Press, 1993.
10. Kuo, M., Chen, L., and Liang, C. Building and evaluating a location-based service recommendation system with a preference adjustment mechanism. *Expert Systems with Applications 36*, 2 (2009), 3543–3554.
11. López-Cózar, R., and Callejas, Z. Multimodal dialogue for ambient intelligence and smart environments. *Handbook of ambient intelligence and smart environments* (2010), 559–579.
12. McCarthy, K., Reilly, J., McGinty, L., and Smyth, B. Experiments in dynamic critiquing. In *Proceedings of the 10th international conference on Intelligent user interfaces*, ACM (2005), 175–182.
13. Polo, L., Mínguez, I., Berrueta, D., Ruiz, C., and Gómez, J. User preferences in the web of data. *Semantic Web*, 0 (2012), 1–9.
14. Spohr, D., Cimiano, P., McCrae, J., and O'Riain, S. Using spin to formalise accounting regulations on the semantic web. In *International Workshop on Finance and Economics on the Semantic Web (FEOSW 2012)* (2012), 1–15.
15. Streefkerk, J., van Esch-Bussemakers, M., and Neerincx, M. Designing personal attentive user interfaces in the mobile public safety domain. *Computers in Human Behavior 22*, 4 (2006), 749–770.
16. Wahlster, W. *SmartKom: Foundations of Multimodal Dialogue Systems (Cognitive Technologies)*. Springer, 2006.

Patterns for HMI design of multi-modal, real-time, proactive systems

Nádia Ferreira
Namahn
Grensstraat 21,
B-1210 Brussels, Belgium
nfe@namahn.com

Sabine Geldof
Namahn
Grensstraat 21,
B-1210 Brussels, Belgium
sg@namahn.com

Tom Stevens
Namahn
Grensstraat 21,
B-1210 Brussels, Belgium
ts@namahn.com

Tom Tourwé
Sirris – ICT & Software
Engineering group
A. Reyerslaan 80, B-1030
Brussels, Belgium
Tom.Tourwe@sirris.be

Elena Tsiporkova
Sirris – ICT & Software
Engineering group
A. Reyerslaan 80, B-1030
Brussels, Belgium
Elena.Tsiporkova@sirris.be

ABSTRACT

The paper reports on the findings of our quest for existing multi-modal design patterns. Further, it describes our approach to elicit new ones from a diversity of real-world applications and our work on organizing them into a meaningful pattern repository using a set of pre-defined parameters, so that they can be described in a uniform and unambiguous way easing their identification, comprehensibility and applicability. These patterns enable designers to optimize the interaction between human operators and systems that reason about and proactively react on information captured e.g. via sensors. Therefore we think that research on interaction with smart objects could benefit of this work.

Author Keywords

Human-centred Interface Design; Interface Design Methodology; Multi-modal Design Patterns; Adaptive Interfaces; Intelligent Support for Complex Tasks; Pro-active Paradigms for User Interaction.

ACM Classification Keywords

H.5.2: User interfaces, H.1.2 User/Machine Systems, D.2.2 Design Tools and Techniques- User Interfaces.

General Terms

Human Factors; Design.

IUI 2013 Workshop: Interacting with Smart Objects, March 18, 2013, Santa Monica,CA, USA Copyright is held by the author/owner(s)

INTRODUCTION

Rationale

The design of multi-modal, adaptive and pro-active interfaces for complex real-time applications requires a specific approach in order to guarantee that the interaction between human and computer remains natural. In order for the interface to adapt to the user and the context, the system needs to reason about her needs and proactively adapt to these while keeping the user in control. The HMI (human-machine interface) design should accommodate varying forms of interaction, depending on what is most appropriate for that particular user at that particular time. HMI design patterns are a powerful means of documenting design know-how, so that it can be re-used. We propose a formal framework to organize and annotate this know-how so that the designer (or at runtime, the system) is supported in the selection (and instantiation) of a pattern, fit to the situation at hand.

Project context

The contribution described in this paper has been developed in the context of ASTUTE[1] a large EU research project. The project focuses on the design of intelligent user interfaces, providing pro-active decision support to the user. The ultimate goal is to develop a platform for building embedded products that capture, reason and act upon user intentions thereby taking into account the user's context (i.e. user environment and all the factors which will influence the user performance) and state (i.e. aspects determining the ability of the user to perform in a given

[1] ASTUTE Pro-active decision support for data-intensive environments; http://astute-project.eu/

situation, such as stress level, fatigue, …). The project approach will be validated with various demonstrators proposed by leading industrial partners in the following domains: automotive (infotainment decision support system aiming at increased driver's safety and comfort), avionics (anticipation support for the pilots to plan their tasks for the descent, under a huge workload), emergency dispatching (distributed and highly dynamic decision support for fire-fighters, their commander and a crisis team facilitating the management of emergency events), building management (context-tailored support for the daily activities of the staff of a rest home) and manufacturing process management (distributed control room solution supporting the different roles on the production floor via distributed devices, both fixed and mobile). Our goal in this project is to provide an appropriate methodology for user interface design, based on design patterns. For this purpose, we have developed a generic approach for collecting and organizing HMI design patterns, so as to facilitate the design of human-centred intelligent interfaces.

Human-centred interaction design and design patterns

In recent years, the need for human-centred design in the development of embedded systems has been recognised ([1], [2], [3] [4]). In need of a systematic approach to their activities, human-machine interaction (HMI) designers are developing human-centred design (HCD) methodologies ([5] [6], [7]). Throughout these methodologies HMI design patterns play an important role in exploring and specifying the interaction between the human user and the computer- in that they inspire design and enable to reuse concrete solutions through appropriate descriptions [8]. The idea of a pattern as a piece of proven and reusable design knowledge has already been applied to interaction design by a number of organizations, resulting in pattern libraries [9], [10], [11], [12], [13], [14] and a key reference publication [15]. More specifically, design patterns for multi-modal interaction have also been investigated in [16], [17], [18], [19]. That collection, far from complete, triggered us to find new patterns and to develop a method to refine their description.

Section 2 describes how we have been collecting patterns, both from the literature and from the demonstrator designs. Section 3 explains our methodology work on organizing the collection of patterns. Section 4 provides an outlook to future activities to continue this research.

COLLECTING PATTERNS

In this section, we describe how a collection of patterns was assembled as input to our methodology. Via a focused literature review, all existing patterns relevant for our project were listed, i.e. patterns for systems that support the users' decision making through multi-modal and pro-active interfaces. Furthermore, new patterns were elicited in a bottom-up fashion, from the designs of the demonstrators discussed above.

Relevant patterns from literature

As a basis for our collection of patterns to be offered to the project consortium we explored the literature on multi-modal interaction design patterns. We selected, consulting [19], [17], [15], [13], [16], [18] and [20], 24 patterns deemed applicable to the type of systems envisaged in our project. To validate their applicability to our project, we organised a workshop among the designers of the different application domains. During this workshop, each of the demonstrator designers indicated which ones of the 24 identified patterns were relevant for their design. Each demonstrator identified about 13 relevant patterns. 10 patterns from the list were relevant for at least 4 of the 5 demonstrator designs.

Identifying new multi-modal patterns from designs

Method

As a method to identify new patterns, we used the 6 steps worked out by [21] on the basis of [8]'s motivation: observation, comparison, enrichment by imagination, prototyping, forces identification and applicability context analysis. During a workshop, the demonstrator designers were made aware of the concept and usefulness of patterns.

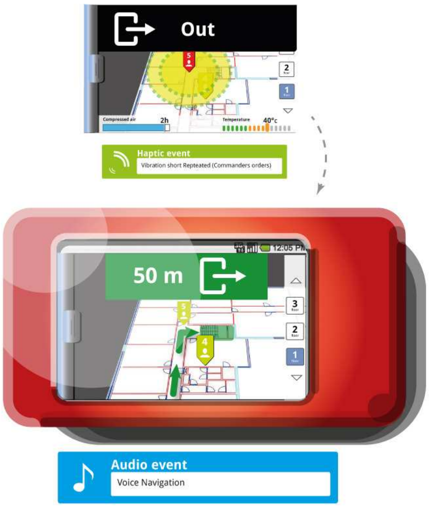


Figure 1: 'Context based information articulation' pattern in Emergency response

New patterns were identified in a combined fashion: HMI designers proposed a set of new patterns (top-down), demonstrator designers scrutinized their designs and proposed a set of patterns (bottom-up). Again in a very collaborative and iterative manner among the demonstrator designers of the 5 application domains, the proposed patterns were discussed, filtered and refined. However, not all proposed patterns were included in this final list since typically some system functions were confused with patterns. For example, route calculation or task management are functionalities, for which particular interaction patterns might be defined. The candidate list was filtered in terms of applicability to the different application domains, yielding a final list of 8 patterns:

- Multiple warnings
- Meaningful sounds
- Goal progression-based information detail and adaptation
- Mode switch
- Interactive instructions
- Modality combination for urgent messages
- Spatial point representation
- Context-based information articulation

Example

The following example illustrates the type of patterns that were newly identified:

Label	Description
Pattern name	Goal progression-based information detail and adaptation
Problem	The user is performing a task, while the goal and execution progression are known by the system. How to reduce information overload and provide the user with the essential and specifically tailored information?
Solution	At the appropriate time, the system adapts the level of information detail and perspective conveyed to the user, based on the user's proximity (in space and/or time) to her goal. In visual modality, this can be considered as a zoom into important information by refining the level of information detail and by providing complementary information. Consider using an extra modality in order to notify the user of a change during the task and display of information.

Table 1. Example of newly identified patterns

To present the patterns at a high level, we cite the label, problem and solution, and an example that gives a concrete idea of what the pattern intends to support. The full description of patterns will include worked out prototypes from the demonstrator designs.

Figure 1 illustrates the application of a new pattern in the emergency dispatching domain.

Pattern themes

Space limitation does not permit to list and define all patterns identified so far for our project. We have grouped the patterns into themes. The description of these themes gives an idea of the range and typology of all patterns relevant to our project:

System capacities - related to system features e.g. task management, commands…

Input - patterns that are either multi-modal, or specific to one modality (e.g. speech-enabled form) or two (e.g. multi-modal interactive maps)

Criticality support - applying to critical situations e.g. alerts, simulation…

Context-aware systems - requiring input about user state and context from sensing devices in order to adapt the interaction

Pro-active system behaviour - depending on the capacity of the system to anticipate/predict the needs of the user and react accordingly

Output - presenting information to the user in the most appropriate form

Although we were able to identify the patterns from the literature that are applicable to the designs and to describe new patterns in the same textual format, we observed that this level of description is not optimal in view of reusability in other domains.

Section 3 describes our endeavour to refine and formalize the description of patterns in a step towards a genuine design methodology.

PATTERN-BASED DESIGN METHODOLOGY

Goals

Re-use

The main motivation for describing existing solutions to known problems is to make this design knowledge available to others, both designers and non-designers. However, in order to be usable, patterns need to be described in a uniform way at the right level of detail so that they become applicable to other design contexts. Apart from a certain level of ambiguity associated with the textual description of the pattern problem and solution, the questions 'when?' and 'how?' also need to be answered. This information is missing from most multi-modal pattern descriptions we found in the literature, those found in [15] and [13] being noticeable exceptions.

Selection

The designer needs to be able to select the pattern that is applicable, based on the characteristics of a particular situation: context, information type and user type. Most pattern collections use a hierarchical classification scheme to support this process. Patterns belong to categories, possibly to subcategories, as in [15]. For instance, within the pattern category 'tables' one will find subcategories such as 'basic table', 'table with grouped rows', etc. The user's selection process is guided by the labels of the (sub)categories. This principle is also used in on-line pattern libraries, such as Welie.com [11] or Yahoo.com [10]. However, the limits of hierarchical classification are well known: different persons may classify elements in different classes, resulting in the difficulty to find an item. Moreover, a strictly hierarchical organization does not suffice to enable a motivated selection because it approaches the patterns from only one perspective. We need a richer organization where patterns can be selected from multiple perspectives and all their characteristics or attributes are taken into account.

The above described selection process applies to the selection at design time (i.e. when the interaction is designed, before implementation). However, in the applications that we envisage, some decisions need to be made at runtime (i.e. while the user is interacting with the system). For instance, depending on the context or the user state, a particular modality is not available. In this case, part of the pattern may be instantiated slightly differently. We might also envisage that an adaptive interface applies one or another pattern at runtime, depending on the user's context.

In the following sections we describe the methods incorporated in our methodology to match the above goals.

Methods

Structural description

In order to document patterns in a consistent way, we propose a structure that contains the necessary information to allow for re-use of this pattern in new designs. It is based on a proposal by [21] compiled on the basis of pattern structures by different authors, building further on the first attempts by [8]. This structure still needs to be validated by applying it to all our newly identified patterns. It describes the following pattern characteristics: name, task and problem, context, applicability, solution, consequences, examples and relationships.

Modelling framework

To meet all our goals however, we need a more formal level of analysis of the characteristics that specify a pattern. [22] and [23] have argued that, in order to be usable in practice, HMI design theory (guidelines) and expert knowledge (expertise and patterns) need to be formalised. They propose an ontology-driven semantic modelling framework and distinguish 3 levels of modelling abstraction: domain, expert and application-specific knowledge. Domain knowledge captures the generic HMI design knowledge in the form of concepts that are typical in multiple smart environments. For instance, the concepts 'user', 'device', 'primary activity', 'modality' are connected to each other by means of relationships ('uses', 'performs', 'supports_modality'). Expert knowledge incorporates the specific knowledge on multi-modal interaction residing in the community of expert designers (e.g. in the form of design patterns). Finally, application-specific knowledge formalises what needs to be known about a particular application area (in our case: the 5 demonstrator domains). For instance, it will be specified that the activity 'measuring_air_quality' in the emergency response domain requires the use of 'both_hands' and that the location 'emergency_site' 'has_noise_level' 'loud'. By separating these layers, a maximum reusability of design knowledge (i.e. design patterns) is guaranteed.

Moreover, [22] illustrates with an example that the formalisation of the features that determine the situation at hand (as captured e.g. by sensors) and the characteristics of the interaction between the user and the system allows for making decisions at runtime on the appropriate modality in a particular situation.

In line with this framework, we have derived a set of parameters to specify design patterns. Our hypothesis is that these parameters will facilitate the description of patterns, the selection of the appropriate one (at design time) and the selection of the right modality and other features of the interaction (at runtime).

Parameters

A set of parameters to be used for HMI pattern description and specification has been derived through multiple interactive and iterative sessions between a mixed team of HMI designers and ontology engineers. The goal of these sessions was to develop a uniform HMI pattern model reflecting the need for a formalized description and an increased level of detail, in order to be able to decide on pattern applicability in a certain design context. In the spirit of ontology-based modelling, the resulting model is a hierarchical class structure, describing pattern parameters and their attributes. Two major types of parameters have been identified: 1) parameters that characterize the pattern (see Figure 2) and 2) parameters that characterize the situation in which the pattern will be applied. The pattern parameters specify the essence of the pattern and impose constraints on the situation in order to be applicable. For instance, if a pattern is specified to have as an interaction mode 'visual output', the visual output interaction channel of the user needs to be available. On the other hand, some features of the situation might determine the instantiation of particular pattern variables (e.g. whether or not to use a particular modality).

Our study showed that the specificities of patterns can be described in a sufficient detail by means of the following 3 parameters:

Information: features of the information conveyed via the interaction e.g. type, volume to be presented, complexity

Interaction mode: characteristics and constraints of the interaction channel to be used e.g. modality and direction.

Interaction events: features of the interaction events triggered by the pattern e.g. what is its priority and importance for the user, whether the interaction will interrupt the task that the user is currently performing, does it require coordination with other events or tasks?

It also emerged from our study that the following 3 parameters are sufficiently expressive to fully characterise the situation in which patterns can be applied:

User: profile of the user, both intrinsic (seniority w.r.t. task, familiarity with the system) and context related (focus w.r.t. device, alertness, availability of senses, role in the interaction).

Context: the situation in which the user finds herself e.g. safety, stability, privacy, visibility and noise. Both physical and social context are modelled.

Device: features of the device(s) with which the user is interacting e.g. mobility constraints, size, previous interaction mode, modality, direction (in or output).

Various relationships have been derived between the parameter classes and subclasses as for instance: Device supports Interaction Mode, User executes Task, Task is PartOf Workflow, User has Context, Device sends Information, User interactsVia Interaction Channel, etc.

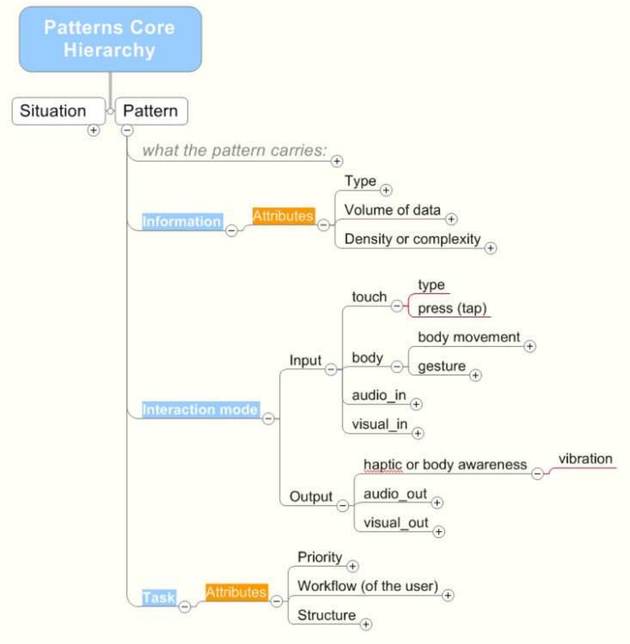


Figure 2: Semantic parameters that specify the pattern

CONCLUSIONS AND FUTURE WORK

The methodology set out in this paper lays the foundation for future work on several fronts.

First, the proposed patterns will be validated both at design time and at runtime. Most of the patterns will be instantiated in at least 2 of the 5 demonstrators of our project, hence we will find out which patterns are valid across domains. Feedback of the designers about the use of the patterns in the designs will allow us to refine, tune and enhance the patterns. Also, as the demonstrator prototypes have been developed, extensive user testing will yield additional feedback on the utility of the pattern to enhance the user experience.

Second, we will further elaborate the structural description formalism. Starting with the existing structure, we will refine it for multi-modal patterns. We will further investigate what are the best means to describe and visualize the specificities of multi-modal patterns.

The proposed hierarchical parameter model and the relationships defined between the different parameter classes are presently being implemented as a full-fledged ontology enabling further model refinement and reasoning.

Within the consortium, a runtime adaptive HMI engine is being developed on the basis of the parameters proposed in this paper. As described above, context values will be derived from sensor data and reasoning about these context values will enable to determine the applicability of a pattern in a particular situation or the selection of a particular variant of the pattern. Similarly, smart objects could behave differently according to the context by applying one or another interaction pattern at runtime.

Through this work, we aim at demonstrating that the modelling of appropriate data about user state and context linked to the specification of interaction patterns constitute a powerful means to realise a proactive, adaptive multi-modal interaction.

The benefits of this approach have already been demonstrated within the project consortium, as the patterns have fostered multiple communication exchanges about the similarities and the differences across the different demonstrator domains, in view of better grasping the essence of the envisaged type of systems. We believe that the results of this project, including the release of a well-documented collection of interaction design patterns, could benefit a larger community, encompassing the smart objects community, in particular but not limited to those involving interaction with users. This can be achieved by setting up a collaborative environment where different stakeholders in the design domain could interact, exchange ideas, improve and annotate patterns proposed by others and contribute new patterns.

ACKNOWLEDGEMENTS

This research is supported by the ARTEMIS JU (http://www.artemis-ju.eu/) and by the Brussels institute for research and innovation Innoviris (www.innoviris.be) through the ASTUTE project (www.astute-project.eu).

REFERENCES

1. Hudson, W. „Adopting User-Centered Design within an Agile Process: A conversation," *Cutter IT Journal,* vol. 16, nr. Software Usabillity, 2003.

2. Schaffer, E. Institutionalization of Usability: A Step-By-Step Guide, Addison-Wesley Professional, 2004.

3. I. International Organization for Standardization, „ISO 9241-16:1999 Ergonomics of human-system interaction," TC 159/SC 4 Ergonomics of human-system interaction, 1999.

4. Plass-Oude Bos, D., Poel, M., Nijholt, A. „A Study in User-Centered Design and Evaluation of Mental Tasks for BCI," *Lecture Notes in Computer Science,* vol. Volume 6524/2011, pp. 122-134, 2011.

5. Cooper, A. About Face: The Essentials of Interaction Design, IDG Books, 1995.

6. Goodwin, K., Cooper, A. Designing for the Digital Ag How to Create Human-Centered Products and Servic Wiley, 2009.

7. Browne, D. P. Studio: structured user-interface design for interaction optimisation., Prentice-hall, 1994.

8. Alexander, C., Ishikawa, S., Silverstein, S. A Pattern Language, NY: Oxford University Press, 1977.

9. Oy, P. F. "Patterny," Pattern Factory Oy (Ltd.), [Online]. Available: http://patternry.com/. [Accessed 10 05 2012].

10. "Yahoo! Design Pattern Library," [Online]. Available: http://developer.yahoo.com/ypatterns/. [Accessed 10 05 2012].

11. Van Welie, M. "Patterns in interaction design - Patterns library," 2008. [Online]. Available: http://www.welie.com/patterns/. [Accessed 10 05 2012].

12. Morville, J., Callender, P. Search Patterns: Design for Discovery, O'Reilly Media, 2010.

13. Saffer, D. Designing Gestural interfaces, Sebastopol: O'Reilly Media Inc, 2009.

14. Scott, T., Neil, B. Designing Web Interfaces, Principles and Patterns for Rich Interaction., O'Reilly Media, 2009.

15. Tidwell, J. Designing interfaces, Patterns for effective interaction design, O'Reilly Media, 2010.

16. Godet-Bar, G., Dupuy-Chessa, S., Nigay, L. "Towards a system of Patterns for the design of Multimodal Interfaces," in *Computer-Aided Design of User Interfaces V, Proceedings of the Sixth International Conference on Computer-Aided Design of User Interfaces CADUI '06*, Bucharest, 2006.

17. Wolff, C., Ratzka, A. "A Pattern-Based Methodology for Multimodal Interaction Design," in *Lecture Notes in Computer Science, 2006, Volume 4188/2006*, Berlin, Springer-Verlag Berlin Heidelberg, 2006, pp. 677-686.

18. Ratzka, A. "Design Patterns in the Context of Multi-modal Interaction," in *Proceedings of the Sixth Nordic Conference on Pattern Languages of Programs*, Bergen, 2007.

19. Ratzka, A. "Identifying User Interface Patterns from Pertinent Multimodal Interaction Use Cases," in *Mensch & Computer 200*, München: Oldenbourg Verlag, 2008.

20. Mahemoff, M. „Design Reuse in HCI and Software Engineering, Phd Thesis," University of Melbourne, Depatrment of Computer Science and Software Engineering, Melbourne, 2001.

21. Kriwaczek, F. "Software Engineering Design II," 09 03 2009. [Online]. Available: http://www.doc.ic.ac.uk/~frk/frank/da/hci/pattern%20handout.pdf. [Accessed 10 05 2012].

22. Tourwé, T., Tsiporkova, E., González-Deleito, N., Hristoskova, A. „Ontology-driven Elicitation of Multimodal User Interface Design Recommendation," in *Proc. of The 23rd Benelux Conference on Artificial Intelligence, BNAIC 2011*, Ghent, Belgium, 2011.

23. Tsiporkova, E., Tourwé, T., González-Deleito, N. Hristoskova, A. „Ontology-driven Multimodal Interface Design for an Emergency Response Application," in *Proc. of the 9th International Conference on Information Systems for Crisis Response and Management, ISCRAM 2012*, Vancouver, Canada, 2012.

A Mobile User Interface for Semi-automatic Extraction of Food Product Ingredient Lists

Tobias Leidinger, Lübomira Spassova, Andreas Arens, Norbert Rösch
CRP Henri Tudor, CR SANTEC, 29, avenue J.F. Kennedy, L-1855 Luxembourg
{tobias.leidinger, luebomira.spassova, andreas.arens, norbert.roesch}@tudor.lu

ABSTRACT

The availability of food ingredient information in digital form is a major factor in modern information systems related to diet management and health issues. Although ingredient information is printed on food product labels, corresponding digital data is rarely available for the public. In this article, we present the Mobile Food Information Scanner (MoFIS), a mobile user interface that has been designed to enable users to semi-automatically extract ingredient lists from food product packaging. The interface provides the possibility to photograph parts of the product label with a mobile phone camera. These are subsequently analyzed combining OCR approaches with domain-specific post-processing in order to automatically extract relevant information with a high degree of accuracy. To ensure the quality of the data intended to be used in health-related applications, the interface provides methods for user-assisted cross-checking and correction of the automatically recognized results. As we aim at enhancing both the data quantity and quality of digitally available food product information, we placed special emphasis on fast handling, flexibility and simplicity of the user interface.

Author Keywords

Mobile user interface; information extraction; food product information; optical character recognition; data acquisition.

ACM Classification Keywords

H.5.2. [Information Interfaces and Presentation]: User Interfaces – Graphical user interfaces, Interaction styles

INTRODUCTION & RELATED WORK

Ingredient information about food products can be interesting for different groups of people due to either ethical or health reasons, such as finding organic or vegan ingredients, or filtering out products unsuitable for allergy sufferers or diabetes patients. Although ingredient information is printed on food product labels, corresponding digital data is rarely available for the public. Various online food databases provide ingredient data; most of them are user-maintained, such as Codecheck.info, Barcoo.com, Fddb.info, Das-ist-drin.de or Wikifood.eu, which has been developed and is maintained by the CRP Henri Tudor. Many platforms also provide interfaces for mobile devices, so that food information becomes more easily accessible. Most databases rely on the participation of volunteers and therefore offer interfaces for users to add new products or edit product information. However, manual entry of food information is tedious and time consuming, which restricts the growth of the corresponding databases.

According to a 2005 WHO report [6] and Robertson et al. [14], the attitude of European consumers is changing towards the intensified consumption of healthy food. In order to enable users to take informed decisions concerning the healthiness of food, extensive food information platforms are needed. Research on how to encourage users to voluntarily provide digital data have resulted in game-like approaches, for example the mobile game Product Empire by Budde and Michahelles [2], where users can build virtual empires by uploading product information. Still, to the best of our knowledge, none of the approaches that involve user input of digital food data offer methods for automated image-based data extraction of ingredient lists, although the data is available, printed on the product labels. In addition, food product labels change frequently as indicated by Arens et al. [1], so that it is important to keep the data up-to-date.

Image processing and optical character recognition (OCR) have been in the focus of research for many years, and corresponding commercial tools as well as open-source solutions are available for desktop computers. There are mobile OCR tools using server-based methods to process images captured by mobile devices, such as Google Goggles[1] or the ABBYY Business Card Reader[2]. For translation purposes, mobile apps for instant recognition and translation of written text are available, e.g., Word Lens[3] or TranslatAR [5]. Laine and Nevalainen describe a theoretical approach to how OCR can be performed directly on mobile devices [9], and there is furthermore at least one prototype implementing the Tesseract OCR engine[4] for Android devices[5]. Most applications of OCR tools require scanned text documents that are of high quality and have a clear contrast between text and background to produce good recognition results. Ingredient lists,

IUI 2013 Workshop: Interacting with Smart Objects, March 18, 2013, Santa Monica,CA, USA

[1] http://www.google.com/mobile/goggles/
[2] http://www.abbyy.com/bcr/
[3] http://questvisual.com/us/
[4] http://code.google.com/p/tesseract-ocr/
[5] https://github.com/rmtheis/android-ocr

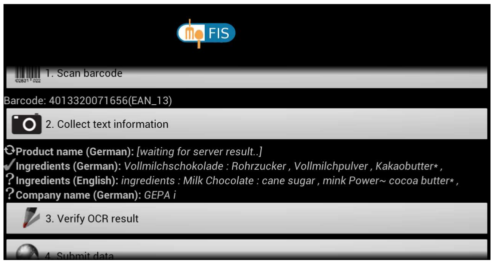
however, have various appearances like different shapes of product packaging, varying foreground and background colors, glossy surfaces or irregular background patterns. In our research setting, the pictures are assumed to be taken with a low resolution mobile phone camera. Taghva and Stofsky point out that OCR errors vary from device to device, from document to document and from font to font [17]. This can induce problems with word boundaries and typographical mistakes. In their article, Taghva and Stofsky present a spelling correction system for the Emacs platform on desktop computers. However, OCR input and error correction on a mobile phone require rather multi-modal approaches. The advantages of multi-modal systems for error correction over uni-modal ones are discussed in a user study by Suhm et al. [16]. Dumas et al. describe the differences between classical graphical user interfaces (GUI) and multi-modal ones (MUI) [3]. The semi-structured nature of ingredient lists makes full automation extremely challenging, although the domain vocabulary of food product data is rather restricted. To compensate for weaknesses of the OCR and post-processing systems, the user has to input unknown words or unrecognized parts using other text input methods, like the mobile phone's keyboard. In this context, Kristensson discusses challenges for intelligent text entry methods [8]. She states that "text entry methods need to be easy to learn and provide effective means of correcting mistakes".

Most references concerning multi-modal interfaces focus on speech recognition together with correction methods. In the last decades, several user interfaces for speech recognition error correction have been presented. The systems interpret spoken language and the interfaces allow the user to correct the result, for example using pens, touch screens or keyboards at mobile phones or 3D gestures at Kinect-based game consoles. Some of the techniques encompass word confusion networks, which show different candidates for each recognized word that can be replaced [12, 18], some show interfaces that allow the user to select sentence or word alternatives [4], and others use the dasher interface, which allows users to navigate through nested graphical boxes in order to select subsequent characters [7]. In the food-related domain, Puri et al. propose an approach to refining the results of an image-based food recognizer by allowing the user to list out each of the food items present in a picture using spoken utterances [13]. Although the project uses disambiguation of recognition results through incorporation of input from different modalities (image and speech), the system by Puri et al. does not offer any further opportunity for error handling through the user.

The present paper introduces the Mobile Food Information Scanner MoFIS – a user interface for mobile devices that enables semi-automatic OCR-based information extraction from food product packaging.

MOBILE USER INTERFACE

The MoFIS interface aims at enabling users to add product information to a food product database in an effortless way with their mobile phones, in order to enhance both the data quantity and quality of digitally available food product information. We have implemented a prototype that provides possibilities for scanning product barcodes, taking pictures of product packaging, marking and tagging regions of interest (ROIs) and cross-checking and correcting product information with a special emphasis on fast handling, flexibility and simplicity of the user interface.



Barcode: 4013320071656(EAN_13)

Product name (German): *[waiting for server result..]*
✓Ingredients (German): *Vollmilchschokolade : Rohrzucker , Vollmilchpulver , Kakaobutter* ,*
?Ingredients (English): *ingredients : Milk Chocolate : cane sugar , mink Power~ cocoa butter* ,*
?Company name (German): *GEPA i*

Figure 1: Main menu of the MoFIS application.

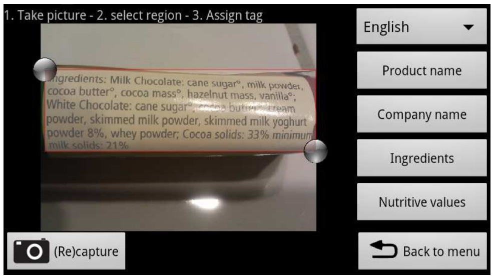

1. Take picture - 2. select region - 3. Assign tag
English
Product name
Company name
Ingredients
Nutritive values
(Re)capture
Back to menu
Ingredients: Milk Chocolate: cane sugar°, milk powder, cocoa butter°, cocoa mass°, hazelnut mass, vanilla°; White Chocolate: cane sugar°, cocoa butter°, cream powder, skimmed milk powder, skimmed milk yoghurt powder 8%, whey powder; Cocoa solids: 33% minimum milk solids: 21%

Figure 2: Image capturing interface with selected ROI.

Main menu

Using four buttons, the user can start the different tasks of the data acquisition process (1. barcode scanning, 2. taking pictures to collect text information, 3. correcting and verifying the OCR result and 4. submitting data). Products are identified by their EAN codes using the ZXING barcode scanner library[6].

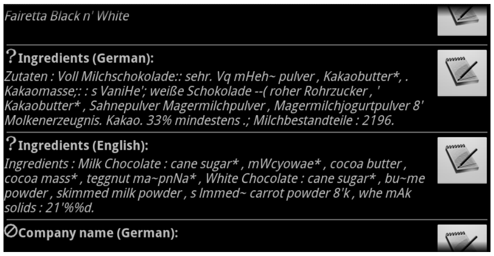
Collecting information

For providing product information, the MoFIS app offers the user the possibility to take several pictures of the food product packaging in order to capture all areas containing relevant information, which is automatically extracted using OCR. The data preview in the main menu adapts whenever the user provides or edits food-related information or when a provided image has been successfully interpreted by the system. Small status icons to the left of the respective preview line visualize the current state of each data item (Figure 1). In each of the pictures, the user can mark several regions of interest (ROIs). Furthermore, the language and the type (product name, company name, ingredients, nutritive values) of each fragment can be specified (Figure 2).

Result confirmation

The user is presented an overview of all information that has been extracted and interpreted by the OCR engine so far. It

[6] http://code.google.com/p/zxing



Fairetta Black n' White

?Ingredients (German):

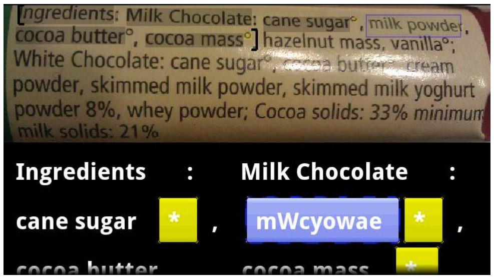
Zutaten : Voll Milchschokolade:: sehr. Vq mHeh~ pulver , Kakaobutter*, . Kakaomasse;: : s VaniHe'; weiße Schokolade --(roher Rohrzucker , ' Kakaobutter* , Sahnepulver Magermilchpulver , Magermilchjogurtpulver 8' Molkenerzeugnis. Kakao. 33% mindestens .; Milchbestandteile : 2196.

?Ingredients (English):

Ingredients : Milk Chocolate : cane sugar* , mWcyowae* , cocoa butter , cocoa mass* , teggnut ma~pnNa* , White Chocolate : cane sugar* , bu~me powder , skimmed milk powder , s lmmed~ carrot powder 8'k , whe mAk solids : 21'%%d.

⊘Company name (German):

Figure 3: Overview of OCR results.

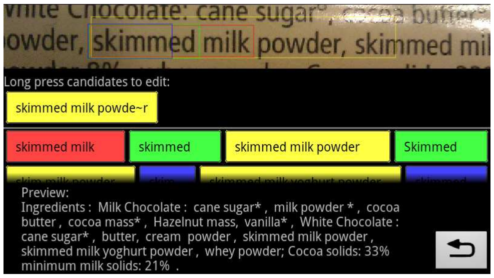


[Ingredients: Milk Chocolate: cane sugar°, milk powder, cocoa butter°, cocoa mass°] hazelnut mass, vanilla°; White Chocolate: cane sugar°, cocoa butter°, cream powder, skimmed milk powder, skimmed milk yoghurt powder 8%, whey powder; Cocoa solids: 33% minimum milk solids: 21%

Figure 4: Ingredient list overview.

shows the status of the result for each ROI, so that the user can identify possible errors at a glance (Figure 3). The edit button opens a view for further analysis and error correction. In order to facilitate the user's interaction during the confirmation task, the corresponding original picture has been integrated in different views of the user interface. The aim of this approach is to keep the user focus on the screen and, in this way, to reduce the probability of errors.

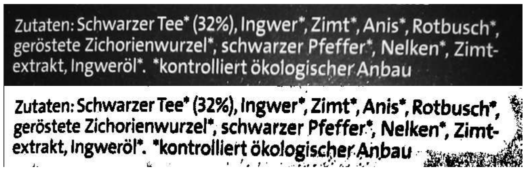
Due to the complex semi-structured nature of ingredient lists and the significance of their content, the interface provides particular user support in the task of finding possible errors in the corresponding OCR results and offers different possibilities for error correction. An *ingredient list overview* enables cross-checking by showing a section of the original image containing the ingredient list in the upper half and the corresponding text candidates in the bottom half of the screen (Figure 4). The two parts of the view are synchronized in such a way that the image shows only a section corresponding to the visible candidates, and scrolling of the candidates shifts the image accordingly. Furthermore, the exact section of candidates currently visible in the lower view is marked with brackets in the original picture in the upper view. The overview initially shows the best automatically retrieved candidates according to a comparison with an ingredient dictionary. Candidates with a low confidence score are highlighted with a yellow or a red background, based on the error distance between the best dictionary match and the OCR result. The overview provides the opportunities to *remove* wrong candidates, *edit* minor errors or manually *insert* parts where the OCR entirely failed. A long press on a candidate opens a context menu with the corresponding options. The user can additionally double tap on every candidate to see a detailed *error correction view* presenting several candidates for each position in the ingredient list provided by the OCR post-processing engine. In order



White Chocolate: cane sugar°, cocoa butter°
powder, skimmed milk powder, skimmed mil
Long press candidates to edit:
skimmed milk powde~r
skimmed milk
skimmed
skimmed milk powder
Skimmed
skim milk powder
skim
skimmed milk yoghurt powder
skimmed

Preview:
Ingredients : Milk Chocolate : cane sugar* , milk powder * , cocoa butter , cocoa mass* , Hazelnut mass, vanilla* , White Chocolate : cane sugar* , butter, cream powder , skimmed milk powder , skimmed milk yoghurt powder , whey powder; Cocoa solids: 33% minimum milk solids: 21% .

Figure 5: Detailed error correction view.



Zutaten: Schwarzer Tee* (32%), Ingwer*, Zimt*, Anis*, Rotbusch*, geröstete Zichorienwurzel*, schwarzer Pfeffer*, Nelken*, Zimtextrakt, Ingweröl*. *kontrolliert ökologischer Anbau

Zutaten: Schwarzer Tee* (32%), Ingwer*, Zimt*, Anis*, Rotbusch*, geröstete Zichorienwurzel*, schwarzer Pfeffer*, Nelken*, Zimtextrakt, Ingweröl*. *kontrolliert ökologischer Anbau

Figure 6: Sample of the pre-processing: original image and binarized version.

to compensate for text segmentation problems and to enable the matching of composed ingredient terms, candidates with different text lengths are considered, which are presented in different colors in the correction view. The user can decide which candidate fits best to the original text marked with a box of a corresponding color (Figure 5). If the correct candidate is not suggested but a close match, the user can long press on a candidate in the correction view to change the proposed text. In this way, the user can benefit from the OCR results and thus speed up the data acquisition process even if the recognition results might not be perfectly correct. After the user has made a selection, the subsequent word candidates are automatically adapted by the selection engine.

IMPLEMENTATION

Although modern smartphones become more and more powerful, we followed the approach of the majority of mobile OCR clients and send the captured pictures to a server for pre-processing, OCR and post-processing. The implementation of the user interface is so far limited to Android devices.

Text extraction

The open-source OCR tool Cuneiform[7] is used to extract the ingredient information from the source image. Along with the text data, Cuneiform provides the bounding boxes of recognized letters, which are used to layout the preview components in the user interface. The OCR software is used as a black box, so that improvements are limited to appropriate pre- and post-processing methods.

Pre-processing

In order to achieve reasonable results, the OCR software requires binarized images. A good binarization algorithm for text images amplifies the contrast between text and background. Especially in the case of poor quality pictures of product packaging, pre-processing is an essential step. For

[7] `https://launchpad.net/cuneiform-linux`

this project, we apply a modified version of Sauvola's binarization method [15, 10]. Figure 6 shows an example of the result of the pre-processing algorithm.

Dictionary matching

Our post-processing method can compensate for OCR mistakes while trying to preserve the structure of ingredient lists. Some ambiguities can not be resolved completely automatically, and thus, the user interface offers alternatives, i.e., the possibility to manually change the preselected candidates.

Structure of ingredient lists

Generally, ingredients are free-text phrases, separated by commas or semicolons. Brackets enclose nested sub-lists of ingredients, and special patterns, like declarations of weight or percentages, can be appended to an ingredient. As Taghva and Stofsky state in [17], OCR systems can misinterpret letters, numbers, dividers and spaces, so that detecting word boundaries is not always feasible. Therefore, reconstruction of the correct ingredient list is only possible with a semi-automated approach involving human assistance.

Entity matching

In this work, the existing WikiFood ingredient list database was used in order to extract a corpus of food ingredient terms and corresponding term frequencies. We use the concept of bi-grams to enable access to potential candidates in constant time. For this purpose, the OCR result is parsed and split into individual words. Subsequently, all word entities are collected and matched against the ingredient corpus. In addition, a sliding window with a certain window length is used to combine adjacent ingredients in order to improve the matching of corrupt words and to be able to propose composed ingredient terms in the user interface. On the one hand, this creates overheads as the matching algorithm is executed more often, but on the other hand this approach significantly improves the result of the post-processing. In the matching phase, all candidate alternatives are ordered by their matching quality as compared to the OCR result, taking into account text edit distance, term frequency and term length. The edit distance is calculated using the Levenshtein distance [11] between the OCR result and the candidates. The output of this algorithm is a sorted list of candidates for the start position (offset) of every entity of the OCR result. Candidates at the same offset can span different words or letters and may overlap with neighboring candidates. Depending on the suggestions, it is up to the user to select composed ingredient terms or several individual words one after the other. The algorithm outputs a maximum of 10 best-ranked candidates for every offset position in order to limit network traffic and to filter out inappropriate candidates. The preliminary ingredient list is initially automatically composed of the best-ranked consecutive, non-overlapping candidates.

PERFORMANCE

To test the recognition rate of the MoFIS engine, we chose 20 random food products covering all different packing characteristics in shape (cylindrical, rectangular, uneven) and color (various text colors, background colors, transparent plastic foil). We took pictures of the ingredient lists with a standard mobile phone camera (5MP, auto-focus) indoors under normal daylight conditions, using the automatically triggered flash light. We counted (a) the number of ingredients that were correctly selected initially, without user interaction, (b) the number of candidates that could be found, but had to be selected from the list of alternatives and (c) the number of candidates that had to be input manually[8]. In average, (a) 90.38% of the candidates were recognised correctly, (b) 4.08% could be chosen from suggested alternatives and (c) only 5.54% had to be inserted manually.

CONCLUSION

In this work, we have presented the MoFIS system, consisting of an OCR server and a mobile user interface that can be used to capture pictures with the mobile device, process the pictures on the server and let the user validate and correct the results directly on the phone. Using the MoFIS interface, usually only little effort is necessary to accomplish the correct results. The system can be used to automatically extract and validate ingredient information from food product packaging using a mobile phone, which is the first such attempt to the best of our knowledge. Compared to basic OCR approaches, this leads to more complete and accurate data with only small additional effort.

We claim that this method provides an enormous advantage for user-maintained food databases compared to traditional text input. In the MoFIS system, most of the text is extracted automatically, so that only little user interaction is necessary. Finding errors – and especially correcting them – is supported by the user interface by presenting both original preview and user input simultaneously. As most modern mobile phones have a camera of sufficient quality and as it is possible to run the OCR and post-processing on a server in reasonable time, this mechanism can provide an adequate alternative to current food-related data acquisition approaches, e.g., through web platforms.

In our future research, we plan to evaluate the MoFIS system by conducting user-centered studies and to adapt and extend the system based on the results and the user feedback gathered in the course of these studies.

REFERENCES

1. Arens-Volland, A., Rosch, N., Feidert, F., Herbst, R., and Mosges, R. Change frequency of ingredient descriptions and free-of labels of food items concern food allergy sufferers. *Special Issue: Abstracts of the XXIX EAACI Congress of the European Academy of Allergy and Clinical Immunology, London, UK 65*, s92 (2010), 394.

2. Budde, A., and Michahelles, F. Product Empire - Serious play with barcodes. *Internet of Things (IOT), 2010* (2010), 1–7.

[8] We only considered textual ingredient phrases for this evaluation step, ignoring dividers, special signs and numbers when counting the number of words.

3. Dumas, B., Lalanne, D., and Oviatt, S. Human Machine Interaction. Springer-Verlag, Berlin, Heidelberg, 2009, ch. Multimodal, 3–26.

4. Feld, M., Momtazi, S., Freigang, F., Klakow, D., and Müller, C. Mobile texting: can post-ASR correction solve the issues? an experimental study on gain vs. costs. In *Proceedings of the 2012 ACM international conference on Intelligent User Interfaces*, IUI '12, ACM (New York, NY, USA, 2012), 37–40.

5. Fragoso, V., Gauglitz, S., Zamora, S., Kleban, J., and Turk, M. TranslatAR: A mobile augmented reality translator. In *IEEE Workshop on Applications of Computer Vision (WACV), 2011* (2011), 497–502.

6. Hayn, D. *Ernährungswende: Trends und Entwicklungen von Ernährung im Alltag (Food Change: Trends and developments in nutrition in everyday life)*, vol. 2. Inst. für sozial-ökologische Forschung (ISOE), 2005.

7. Hoste, L., Dumas, B., and Signer, B. SpeeG: a multimodal speech- and gesture-based text input solution. In *Proceedings of the International Working Conference on Advanced Visual Interfaces*, AVI '12, ACM (New York, NY, USA, 2012), 156–163.

8. Kristensson, P. O. Five Challenges for Intelligent Text Entry Methods. *AI Magazine 30*, 4 (2009), 85–94.

9. Laine, M., and Nevalainen, O. A standalone ocr system for mobile cameraphones. In *PIMRC*, IEEE (2006), 1–5.

10. Leidinger, T., Arens-Volland, A., Krüger, A., and Rösch, N. Enabling optical character recognition (OCR) for multi-coloured pictures. In *Proceedings of the ImageJ User and Developer Conference, Edition 1*, ISBN: 2-919941-18-6 (2012).

11. Levenshtein, V. I. Binary Codes Capable of Correcting Deletions, Insertions and Reversals. *Soviet Physics Doklady 10* (Feb. 1966), 707+.

12. Ogata, J., and Goto, M. Speech Repair: Quick Error Correction Just by Using Selection Operation for Speech Input Interface. In *Proc. Eurospeech05* (2005), 133–136.

13. Puri, M., Zhu, Z., Yu, Q., Divakaran, A., and Sawhney, H. Recognition and volume estimation of food intake using a mobile device. In *Workshop on Applications of Computer Vision (WACV), 2009* (2009), 1–8.

14. Robertson, A., Tirado, C., Lobstein, T., Jermini, M., Knai, C., Jensen, J. H., Luzzi, A. F., and James, W. P. T. Food and health in Europe: a new basis for action. Tech. rep., pp 7-90, 2004.

15. Sauvola, J., and Pietikinen, M. Adaptive document image binarization. *PATTERN RECOGNITION 33* (2000), 225–236.

16. Suhm, B., Myers, B., and Waibel, A. Model-based and empirical evaluation of multimodal interactive error correction. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, CHI '99, ACM (New York, NY, USA, 1999), 584–591.

17. Taghva, K., and Stofsky, E. OCRSpell: an interactive spelling correction system for OCR errors in text. *International Journal of Document Analysis and Recognition 3* (2001), 2001.

18. Vertanen, K., and Kristensson, P. O. Parakeet: a continuous speech recognition system for mobile touch-screen devices. In *Proceedings of the 14th international conference on Intelligent user interfaces*, IUI '09, ACM (New York, NY, USA, 2009), 237–246.

Digital Pens as Smart Objects in Multimodal Medical Application Frameworks

Markus Weber, Christian H. Schulz
German Research Center for AI
{firstname.surname}@dfki.de

Daniel Sonntag, Takumi Toyama
German Research Center for AI
{firstname.surname}@dfki.de

ABSTRACT

In this paper, we present a novel mobile interaction system which combines a pen-based interface with a head-mounted display (HMD) for clinical radiology reports in the field of mammography. We consider a digital pen as an anthropocentric smart object, one that allows for a physical, tangible and embodied interaction to enhance data input in a mobile on-body HMD environment. Our system provides an intuitive way for a radiologist to write a structured report with a special pen on normal paper and receive real-time feedback using HMD technology. We will focus on the combination of new interaction possibilities with smart digital pens in this multimodal scenario due to a new real-time visualisation possibility.

ACM Classification Keywords

H.5.2 User Interfaces: Input Devices and Strategies, Graphical HCIs, Prototyping

Author Keywords

Augmented Reality, Medical Healthcare, Real-time Interaction

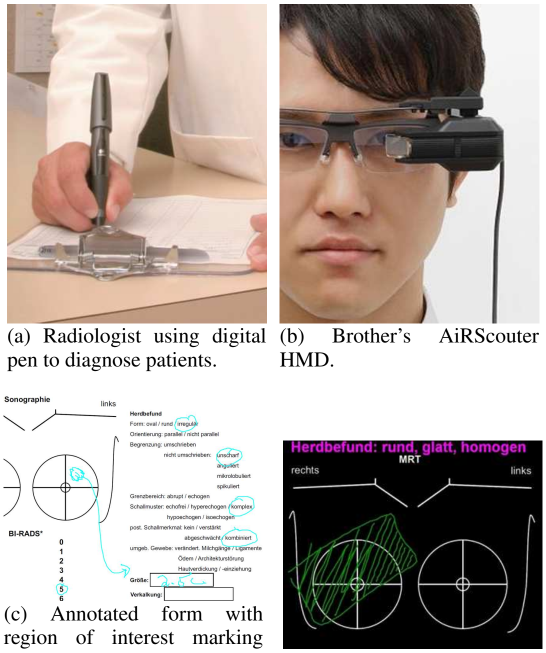
INTRODUCTION

A standard reaction of computer-affine people to this question is that they are much faster with a keyboard. And it is true, handwriting as an input metaphor is a very slow interaction process when you are trying to input information into the computer, especially for those people who learned typing rapidly with the keyboard.

However, people still use a pen for putting down information on paper and a lot of processes are still based on paper documents. In radiology practices paper reporting have been established over the last 20 years. However, this situation is not optimal in the digital world of database patient records. Digital records have many advantages over current filling systems when it comes to search and navigation in complete patient repositories called radiology information systems. In fact, modern hospital processes require digital patient reports. The current practice in hospitals is that dictated or written patient reports are transcribed by hospital staff and sent back to the radiologist for approval. The turn-over time is 2-30 hours and the process inefficient and also prone to error.

THE MOTIVATION

In [4], we presented an interaction method that shows how a radiologist can use our special mammography paper writing system to conduct a full mammography patient finding process. This digital pen based interface enables radiologists to create high-quality patient reports more efficiently and in parallel to their patient examination task. Thereby, he or she uses a digital pen-based interface where the user can write on normal paper which is printed with a light-grey dot pattern in order to allow for the recognition of the writing. Nevertheless, this solution still requires a workstation to provide feedback of the real-time recognition results; hence it limits the mobility of the pen-paper approach. Moreover the doctor has to constantly change his or her sight from the paper to the screen and back, thus the doctor cannot focus on the patient.


Sonographie
links
Herdbefund
Form: oval / rund / irregulär
Orientierung: parallel / nicht parallel
Begrenzung: umschrieben
nicht umschrieben: unscharf
anguliert
mikrolobuliert
spikuliert
Grenzbereich: abrupt / echogen
Schallmuster: echofrei / hyperechogen / komplex
hypoechogen / isoechogen
post. Schallmerkmal: kein / verstärkt
abgeschwächt / kombiniert
umgeb. Gewebe: verändert. Milchgänge / Ligamente
Ödem / Architekturstörung
Hautverdickung / -einziehung
BI-RADS*
0
1
2
3
4
5
6
Größe:
2.5
Verkalkung:
Herdbefund: rund, glatt, homogen
MRT
rechts
links

(a) Radiologist using digital pen to diagnose patients.

(b) Brother's AiRScouter HMD.

(c) Annotated form with region of interest marking the selection of associated anatomical concepts.

(d) Visualisation of recognised results in the HMD display.

Figure 1: Mobile pen-based system.

In [5] we have developed a mobile working station for the radiologist that is called RadSpeech. In the following we have extended the system for the use of multiple mobile stations combined with one stationary screen installation (see Figure 3a). The motivation of the setting here, was to provide hands-free interaction with a medical system using the natural language. Basically, the scenario describes how the medical expert retrieves medical images of one specific patient and then continues to make his finding by attaching semantic information, i. e. Radlex terms [1], to the affected areas within the images. Eventually, the relevant bits of information are then processed by the backend and made persistent for later reference.

We used a client-server-based approach, where each part that involves more computing is runnable on a separate platform. Note that regardless of the introduced modalities, all these approaches share the same underlying goal, namely to increase usability of the doctor's working routine environment, i. e., making knowledge acquisition fast and easy by providing state-of-art user interfaces to human medical expertise [7].

In this work we combine our novel interaction designs with current achievements concerning novel input devices, thus allowing for experiments with new interaction paradigms. For instance we have integrated an innovative mobile display systems in a form of a head-mounted display (Brother's AiRScouter WD-100G) which provide new ubiquitous possibilities for real-time interaction. We applied our technology to the HMD to provide mobile augmented reality interaction system for doctors that can be used during patient examination in the medical routine. The augmented reality system comprises of a digital smart pen (see Figure 1a) and a speech recognizer as input device. On the other side we have applied a see-through HMD (see Figure 1b) for visual feedback and a speech synthesis for audio feedback. See-through HMDs are of special interest as the doctors can focus on the patient during the examination process as the information is augmented in the view field.

In the following sections, we will describe the technical aspects with respect of the pen-based annotation framework, then we will also illuminate the relevant parts of our dialog system. Finally, we will highlight the interesting effects we gain when observing the interplay of both input channels combined with our new augmented reality approach.

MOBILE FEEDBACK PEN-BASED SYSTEM

Possible annotations on the printed paper may include terms to classify diseases which are formalised by using the International Classification of Diseases (ICD-10), annotations of regions-of-interest (ROI) in images, or pen gestures to choose predefined terms (e.g., anatomical concepts). In any case, the system maps the handwriting recognition (HWR) output to one or more medical concepts. Each of these ROIs can be annotated with anatomical concepts (e.g., *lymph node*), with information about the visual manifestation of the anatomical concept (e.g., *enlarged*), and/or with a disease category using ICD-10 classes (e.g., *Nodular lymphoma* or *lymphoblastic*). However, any combination of anatomical, visual, and disease annotations is allowed and multiple annotations of the same region are possible. Whenever an annotation is recognised and interpreted, the result is instantly augmented in the HMD (see Figure 1d) to give immediate feedback. Furthermore, the paper sheets contain special action areas to trigger additional functions of the system, such as a text-to-speech engine to provide optional audio feedback of the recognition results.

The architecture illustrated in Figure 2 is based on the pen component of the Touch&Write framework [1] and provides an overview of the proposed system. First, an ink collector component collects the online stroke data from the digital smart pen via Bluetooth. Now, the mode detection component [6] analyses the stroke information and classifies the annotation into different modes of the handwritten input.

In our scenario the system has to decide whether it deals with handwritten text information, image annotations, or pen gestures. The handwriting is analysed by using the MyScript engine of Vision Objects [2] and gesture recognition is performed by the iGesture framework [3]. Depending on the classification result, either the handwriting recognition or the pen gesture analysis is triggered. The recognition results are passed on to the interpretation layer via the event manager component. As the interpretation layer has a semantic mapping of the paper sheet layout, pen gestures and recognised texts are interpreted in the context of the diagnostic field where they occur.

Finally, a visualisation layer is responsible for providing an appropriate visualisation of the recognised diagnostic findings which depends on the hardware capabilities of the used HMD technology (see Figure 1d). The visualisation in the HMD mainly depends on the resolution of the HMD. In our demo prototype, Brother's AiRScouter (WD-100G) has been used with a screen resolution of 800x600 pixels. Due to the limited screen space, only selected parts of the complete finding form can be presented to the doctor (e.g., only the results of the MRT, see Figure 1d). Making a virtue out of necessity, we display only the diagnostic area of the actual form filling process, which does not overload the screen. For all areas, we present the schematic image with the marked ROIs combined with the selected anatomical concepts, as well as recognised handwritten annotations.

In summary, the mode detection of our smart pen automatically chooses the next step of analysis. The next analysis step will be, either:

- Pen gestures - for triggering predefined functionalities in our multi-modal system,
- Handwriting (common language combined with medical terms) - capturing diagnostic findings,
- Visual annotation - for marking ROIs in medical images.

Finally, as the smart pen *"knows"* the content of the medical forms and provide further information on the HMD, trigger actions, or provide a digitialized report of the finding.

[1] `http://www.radlex.org/` : Last seen 02/09/2013

[2] `http://www.visionobjects.com/en/myscript/about-myscript/` : Last seen 02/09/2013

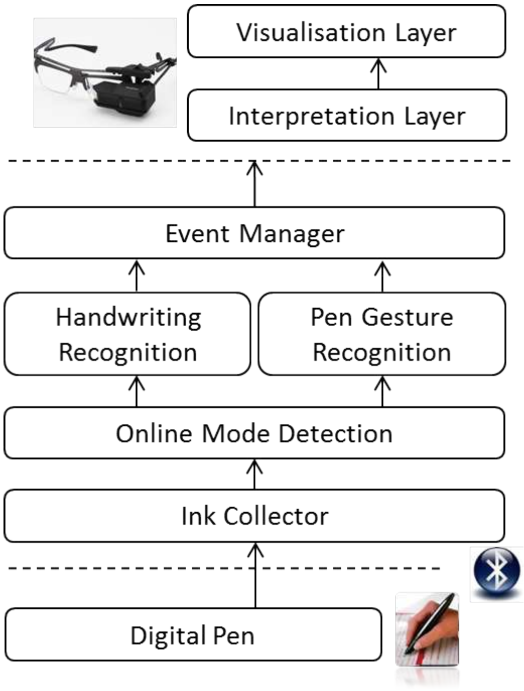


Figure 2: Proposed recognition and feedback data flow. The pen data is collected and further analysed in a three-tier architecture.

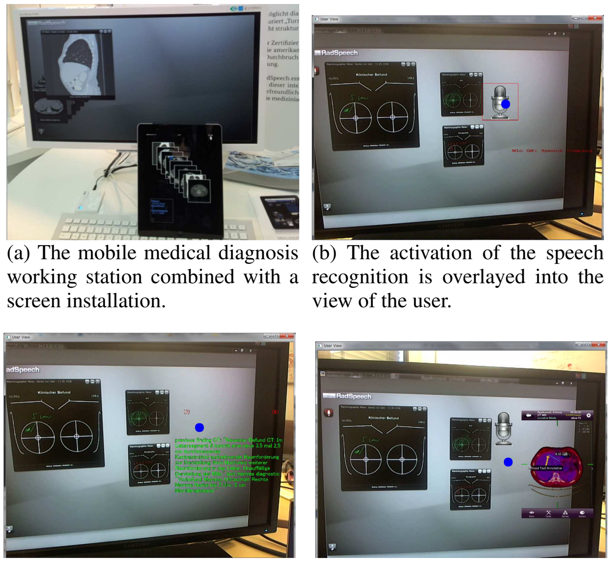
THE SPEECH INTERFACE

We use the microphone array that is integrated into the Microsoft Kinect [3] to transmit audio signals to the speech recognition (Nuance Recognizer 9.0) [4]. The recognizer runs on a speech server that integrates also a speech synthesizer (Nuance SVOX). Our dialogue platform is based on ontological concepts that during runtime models the interaction process inside a production rule engine [2].

We have adopted a design direction that allows the activation of the dialog system using different type of devices. As a result, the user is able to choose the modality which is most convenient in a specific situation. Figure 3b shows the graphical output in the sight of the user when opening the microphone by the movement of the eye. The blue dot within the frame that contains the microphone icon represents the visual focus. The highlighted frame and a notification signalize the activation of the speech recognition to the user.

AN INTERACTION EXAMPLE

The following dialog demonstrates a real-world example; while the radiologist is analysing medical images on the screen application, he or she is requesting for additional information about the patient:



(a) The mobile medical diagnosis working station combined with a screen installation.

(b) The activation of the speech recognition is overlayed into the view of the user.

(c) Overlay of the patients file information.

(d) A video is streamed into the users sight.

Figure 3: The combination of a speech-based interface and a see-through interface.

1 The doctor opens microphone using either eye gaze or pen gestures.
2 **Doctor says:** "Show me the previous finding in the HMD."
3 **HMD:** The sight of the doctor is augmented with the corresponding patient file.
4 **TTS:** "Previous Finding:. . . "
5 The doctor continues with the form-filling process.
6 **Doctor uses pen:** The Radlex terms *round, smooth, homogeneous* are marked.
7 **TTS:** "The annotation *round, smooth, homogeneous* has been recognized"

Figure 4 visualizes the interplay of the components given the dialog example in the context of the multi-device infrastructure. In the centre of the infrastructure we have a proxy, that is responsible to route and forward method invocation to any target recipient [5], i. e., I/O device. The eye tracker first interprets the gaze gesture as an open microphone command (see 1). The invocation of the actual method on the target is passed on through the network by means of the proxy. The calculation of the intended meaning of the speech input is done by the dialog system and will in turn result in the invocation of a remote call on the HMD part. Figure 3c shows the effect in the HMD of a `displayText` call that is triggered by the utterance (see 2). Simultaneously, the dialog system processes the audio format conveying the corresponding information (see 4). Further, during the form-filling process using the digital pen (see 5), user feedback is realized through multimodal output that involves the dialog system and the HMD. In particular, annotating ROI with text information is accompanied by audio feedback, note the sentence produced by the speech synthesizer (see 7) as the Touch&Write Framework recognizes the Radlex terms (see 6). The pen interface accesses the synthesis mechanism via the call `updateFindings` that is

[3] **http://www.microsoft.com/en-us/kinectforwindows/** : Last seen 02/09/2013

[4] **http://www.nuance.com/** : Last seen 02/09/2013

[5] **http://delight.opendfki.de/**: Last seen 02/09/2013

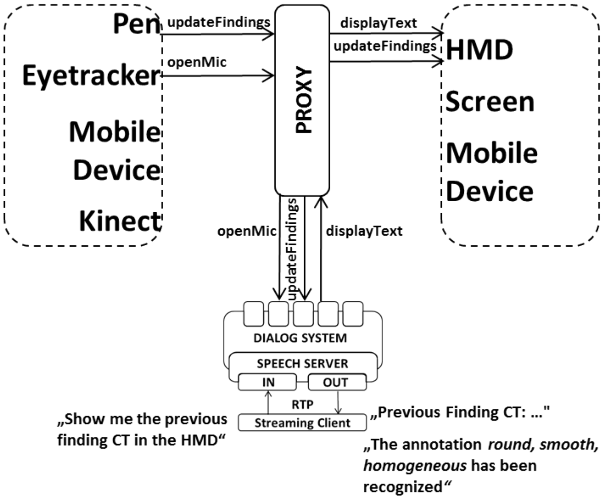


Figure 4: Any of the input devices on the left may serve to activate the speech modality, while on the right basically all output devices are available to give feedback to the user.

forwarded by the proxy to the dialog system. At the same time, the identical call effectuates real-time visual feedback augmenting the view of the doctor by displaying the selected terms in the see-through device (see the red coloured terms in Figure 1d). Besides the purpose of generating adequate multimodal feedback on the basis of the "digital pen print", we can also use the pen input to trigger explicit commands that goes beyond the context of the form-filling process. In Figure 3d a video is shown inside the augmented view. A `videoPlayback` call is triggered when underlining predefined terms within a designated area on the form. Based on our infrastructure we can easily make the latter functionality accessible also to other input modalities, such as speech. Finally we are able also to route the video stream to other output devices, such as the screen installation.

CONCLUSION

We presented a novel mobile real-time smart pen feedback environment which directly projects the results of the digital form-filling process into the eyes of the doctors. Radiologists can perform diagnostic reporting tasks by using their standardised form sheet and handwritten comments as well as simple pen annotations. To access additional information about the patient, we integrate a state of the art dialogue system called RadSpeech.

Our prototype employs modern HMD technology for displaying the real-time recognition results; as a result, our interaction system is mobile and the doctor does not need any additional devices, such as a tablet or smartphone, to check the digital version of the diagnosis. Moreover, to improve the robustness of the speech recognition in a real world scenario we used either pen or eye-gaze gestures to control the speech recognition instead of using continuous speech recognition.

Finally, after controlling the results, the digital version of the diagnostic finding can be transmitted to the radiology information system. This improves the quality and consistency of reports as well as the user interaction. Radiologists are also not forced to dictate information in the order in which it appears in the report. Most importantly, complete reports are available in seconds due to the mobile data acquisition and real-time feedback functionality.

ACKNOWLEDGMENTS

This work is part of THESEUS-RadSpeech (see **`www.dfki.de/RadSpeech/`**), supported by the German Federal Ministry of Economics and Technology (01MQ07016) and Nuance Foundation (see **`www.nuancefoundation.org/`**).

REFERENCES

1. M. Liwicki, M. Weber, T. Zimmermann, and A. Dengel. *Seamless Integration of Handwriting Recognition into Pen-Enabled Displays for Fast User Interaction*, pages 364 – 368. 2012.
2. N. Pfleger and J. Schehl. Development of advanced dialog systems with PATE. In *Proceedings of Interspeech 2006—ICSLP: 9th International Conference on Spoken Language Processing, Pittsburgh, PA, USA*, pages 1778–1781, 2006.
3. B. Signer, U. Kurmann, and M. Norrie. iGesture: A General Gesture Recognition Framework. In *9th International Conference on Document Analysis and Recognition 2007.*, volume 2, pages 954 –958, sept. 2007.
4. D. Sonntag, M. Liwicki, and M. Weber. Digital pen in mammography patient forms. In *Proceedings of the 13th international conference on multimodal interfaces (ICMI '11).*, pages 303–306, Alicante, Spain, 2011. ACM, New York, NY, USA.
5. D. Sonntag and C. H. Schulz. Multimodal multi-device discourse and dialogue infrastructure for collaborative decision making in medicine. In *Proceedings of the International Workshop on Spoken Dialogue Systems Technology*, IWSDS12. Springer, 2012.
6. M. Weber, M. Liwicki, Y. T. H. Schelske, C. Schoelzel, F. Strauss, and A. Dengel. MCS for Online Mode Detection: Evaluation on Pen-Enabled Multi-Touch Interfaces. In *11th International Conference on Document Analysis and Recognition*, 2011.
7. D. L. Weiss and C. Langlotz. Structured reporting: Patient care enhancement or productivity nightmare? *Radiology*, 249(3):739–747, 2008.