TABLETOP ESP: EVERYDAY SENSING AND PERCEPTION IN THE CLASSROOM

Contributors

Richard Beckwith
Intel Corporation

Georgios Theocharous
Intel Corporation

Daniel Avrahami
Intel Corporation

Matthai Philipose
Intel Corporation

Index Words

Vision and Scene Understanding
Architectures for Educational Technology Systems
Interdisciplinary Projects
Pattern Recognition
Context Awareness

Abstract

Today's context-aware computers and computing devices can provide significant assistance in performing everyday tasks. Such devices "perceive" enough of your environment that they can see things the way you do—only better. Because they know who you are with, where you are, and more, context-aware computers can intelligently guide or teach you. With their greatly enhanced context awareness, these systems go far beyond "ease of use" and actually provide meaningful, real-time, relevant support to users.

Introduction

We are living in a time of mash-ups. You can listen to The Beatles's pop music mashed up with Jay-Z's rap on Dangermouse's infamous Gray Album. You can find Craig's List* apartments located on Google* Maps by surfing to *www.housingmaps.com*. Popurls'* social news aggregator at *www.popurls.com* looks across Twitter*, Delicious*, Reddit*, and others to see what is popular on the Web right now. You can blend together many things and get the benefits of each—plus synergies. In this article we discuss what happens when you mix context awareness with task assistance. Context awareness helps a machine understand where it is but not just its location. Within that location, context awareness can tell a machine who is there, what they are doing, and what resources (e.g., objects) are available. Context awareness, therefore, helps the machine by providing it with information about its context, but the trick is to make that information about context useful for users. Task assistance is all about helping users.

Within Future Technologies Research, researchers in the group project known as "Everyday Sensing and Perception," or ESP for short, focus on improving context awareness through various means, including parallelization, machine learning, and machine vision. We are also concerned with making context awareness useful. To this end, we have developed a performance support application, which we describe later in this article, that uses ESP-developed technologies to extend context awareness and task assistance to the tabletop in front of the user.

Context Awareness

Apple's iPhone* "face sensor" (or "proximity sensor") provides an example of context awareness. The face sensor is one of those things that makes an iPhone feel more advanced than a typical mobile phone. The iPhone uses its face sensor to determine whether a user's face is near the display. If a face is nearby, the phone can assume that the person is making a call and turn off the keypad so that the user's cheek doesn't make an errant keystroke, mute the call, or worse. The iPhone is not seeing a face, though. It makes an inference from a sensor event. The sensing is accomplished by use of an infrared LED and an infrared sensor, the combination of which creates a very simple but useful sensing mechanism. The LED pulses while the sensor measures reflectance. If the reflectance changes significantly, then the phone can make a series of inferences: (a) an object is very close, (b) that object is likely to be a face, and (c) the user wants the phone's display off, which will save power and save the user from random keystrokes or button presses. In this way, the iPhone uses sensors to make useful inferences.

"The iPhone uses its face sensor to determine whether a user's face is near the display."

Technology can support a wide array of inferences by using many types of sensors. Most mobile phones include a Global Positioning System (GPS), or they can use nearby cell towers or 802.11 access points for location determination [1]. The phone knows where you are and can show your location on a map or give you directions to the nearest coffee shop, for example. This is where some people think context awareness begins and ends. However, today's phones can sense location and much more. There is a large variety of sensors available for small, low-power applications. Mobile devices often include multi-axis accelerometers, gyroscopes, and magnetometers. Our group has shown that by using just a 3-axis accelerometer mounted on a TV remote control, it is possible to identify which member of a family is using that remote and personalize the TV watching experience for that person [2]. Clearly, context awareness can support many more facets of our lives than simply telling us where we are.

"As different kinds of sensors become ubiquitous in computing devices, the ability of those devices to sense context increases significantly, and the urge arises to fuse these data into a large data stream."

As different kinds of sensors become ubiquitous in computing devices, the ability of those devices to sense context increases significantly, and the urge arises to fuse these data into a large data stream. When data are fused, that is, when data are combined from multiple sources, devices can make even more powerful inferences than they can with unfused data. Luckily, the emerging ubiquity of sensors is occurring at the same time as substantial increases in the ability of today's processors to deal with large volumes of data.

"We are entering an era in which mobile computers will be able to use their video cameras for machine vision."

In fact, technology has become so powerful that we are entering an era in which mobile computers will be able to use their video cameras for machine vision. Video cameras are extraordinarily high data rate sensors, to be sure, but as devices become more powerful, machine vision becomes more tractable, and cameras become just another type of sensor that can be used for context awareness. What the camera sees can be analyzed by an application, and that analysis can be used to support a user (just like when the infrared sensor "sees" a face on an iPhone). In fact, our research group believes that computer vision, fused with data from other sources, will be key to making context awareness more useful for people across wider swaths of their lives.

Performance Support

One question that remains is what benefits these inferences might have for users. What reason would someone have for using a device loaded with sensors to make inferences about their context? One answer is that these devices will be used for context-aware task assistance (or performance support).

Electronic performance support tools have been around for decades. These tools are meant to provide what users need, when they need it. The tools are a source of "just-in-time" support. Bezanson offers a good description:

> A performance support system provides just-in-time, just enough training, information, tools, and help for users of a product or work environment, to enable optimum performance by those users when and where needed [3].

Performance support systems must collect contextual data and use those data to assist users. According to Bezanson, when a system tries to support its users but is not context sensitive, at best it can be "sympathetic." Such a system can be designed to be as easy to use as possible, but it will not be able to use contextual information to make things easier still. Alternatively, a system can be context sensitive but not support users. The best performance support is when a product is both context-aware and tries to help users, as with autopilot or some computer-based agents. For a relevant example, tutoring, unlike simple computer-aided instruction, takes contextual information and uses it to personalize a student's instruction.

"The best performance support is when a product is both context-aware and tries to help users, as with autopilot or some computer-based agents "

The kinds of performance that a device can support become more and more complex as context awareness improves. The face sensor could be considered a simple performance support tool for making phone calls. It uses contextual information, and it makes a phone system easier to use. An installation wizard for software is more complex than the face sensor, as it can sense what applications are on a machine and then ask about relevant preferences and other variables that would affect an installation. At the complex end of the continuum, LeafView* [4] is a mobile application that assists in botanical species identification. When a user aims a mobile device's camera at an example leaf, LeafView uses computer vision to select a small number of potential targets (about 5) from a large database of species (about 80,000 type specimens). A recent version of LeafView also uses the GPS on camera phones for geo-locating the sample. By using computer vision and location sensors, LeafView offers performance support that enables normal citizens to perform as scientists.

On The Tabletop

One context that ESP has been investigating is how technology can support users who are seated or standing at a desk or table. The tabletop is an important context if only because people spend so much time there. Elementary students spend much of their school day (about 1000 instructional hours a year) seated at a desk in school [5]. The 2006 American Time Use Survey [6] showed that US citizens 15 years of age and older spent an average of nearly two hours a day at surface-oriented activities. Clearly, this is a frequent context and, just as clearly, sensors and applications will need to know a lot about what people are doing in this context if our devices are to be most useful there.

Tabletop computing, as a paradigm, has garnered some interest in recent years from the research community, but not in quite the way that we are pursuing it. Tabletop computers, like Microsoft's Surface* or the SMART* Table, are really specialized computers with an integral horizontal display surface with which a user interacts. We would like to be able to support traditional activities on traditional tabletops, so we have been pursuing the use of a context-aware portable computer on a normal tabletop.

"We have been pursuing the use of a context-aware portable computer on a normal tabletop."

The applications and algorithms being developed within the ESP project use computer vision and various other sensors. All of this is done in an effort to better understand the context. Our goal is to enable devices to use this context data to improve a user's quality of life. In the case of the system described here, we are most interested in what is happening both on and around the tabletop in order to support education.

Tabletop Tutoring

Imagine if a low-cost laptop could be a world-class tutor for children, using everyday sensing and perception to understand their activities, moods, and knowledge: in other words, creating a personalized curriculum just for them. This is the vision of ClassmateAssist. Our goal is to use context-aware computing to support the current practices of teachers while, at the same time, providing them with additional capacity to instruct and evaluate their students.

"ClassmateAssist is designed to be a netbook-based application that assigns individual students problems that have been selected to be of the right difficulty for them."

ClassmateAssist is an application designed to assist students and teachers in the use of mathematics manipulatives in kindergarten through second-grade classrooms. Manipulatives are physical objects used to assist in the instruction of math concepts. They are employed every day in more than 60% of all-day kindergarten classes [7]. ClassmateAssist is designed to be a netbook-based application that (a) assigns individual students problems that have been selected to be of the right difficulty for them, (b) follows the students' progress during the process of problem solving, and (c) assists them as they work to solve these problems. The problems were developed in conjunction with kindergarten through second-grade teachers. With ClassmateAssist, teachers can continue to use physical manipulatives that are already in their classrooms and use them on desks that are already there. ClassmateAssist further supports teachers' current practices by autonomously giving tailored support to individual students so that all students in the class can work at their own pace while the teacher provides individual attention to students who need it most at that moment.

Our system has three basic components. First, a component follows the manipulatives as students use them. Second, an interaction planner uses the student's progress and history to select the problems, hints, and feedback the system will provide. Third, after selecting the content, the system uses various strategies to display the problems, hints, and feedback for the student.

Observing the Manipulatives

To follow the manipulatives, ClassmateAssist uses a computer-vision component that watches what students are doing. The application uses a standard camera (such as the one on laptop lids) to watch the manipulatives on the surface between the laptop and the student and to track these objects as the student moves them. However, the system does not need to watch everything. We worked with teachers to define pedagogically relevant, "observable" phenomena for the system to watch. For example, it watches the manipulatives to see where students are building clusters of objects selected from those on the desk and where students' movement of objects will not serve to create the appropriate clusters. It can see whether the students seem to be moving the manipulatives in such a way that they will reach a solution to the task or not. Furthermore, it can recognize common errors or non-goal directed movements.

"Taking various pieces of data, such as movement toward goal and time since assignment as input, ClassmateAssist gives the students new assignments hints on the current task, or feedback on a recently completed task."

Planning the Interaction

Taking various pieces of data, such as movement-toward-goal and time-since-assignment as input, ClassmateAssist gives the students new assignments, hints on the current task, or feedback on a recently completed task.

To determine assignments, we worked with teachers to define a range of tasks, which we verified as appropriate for kindergarten through second-grade classrooms, by deploying a version of the application that used a touch screen and virtual manipulatives. The order of assignments was heuristically determined, but a more principled approach will be used in our final version.

We want to deliver the same types of hints and feedback that teachers would deliver if they were watching the child working. To this end, we are currently analyzing videotapes of a teacher providing one-to-one instruction to students while using manipulatives. We are observing the teacher for the information she provides as well as the contextual variables that elicit communication from the teacher. That is, we are looking at what the teacher said and did as well as the aspects of the students' behavior to which the teacher was responding.

Providing Input to Participants

To reflect what a teacher might say or do, ClassmateAssist provides both auditory and visual information. It uses a speaker on the computer and a text-to-speech (TTS) engine for the auditory assignments and feedback. In practice, we have found that the teachers with whom we are working do not mind the sound in their classrooms. The application also uses the display on the laptop to show information to the student. On-screen video can be "marked up" by highlighting screen regions or superimposing information. Finally, we are pursuing a method to use an integrated pico-projector to actually project the hints or feedback directly onto the physical manipulatives.

Drilling Down on the Tabletop

How does ESP help us reach these goals just mentioned? Various research threads within our labs allow us to select among many capabilities. Among them, we use object recognition, activity detection, facial affect coding, interaction planning, camera/projection integration, and more.

"Because objects on a surface make up a considerable portion of the context, tabletop context awareness requires robust identification of the objects that are on the table."

On and Around the Table

The context of the tabletop includes many contextual variables with which we must concern ourselves. There are the objects on the table, the people around the table, the actions of those people, and their internal states. Each of these things, in turn, can impact how a computer-based tutor should interact with users.

Objects

Because objects on a surface make up a considerable portion of the context, tabletop context awareness requires robust identification of the objects that are on the table. Computer-vision object recognition used to work in restricted contexts, and projects were more "proofs of concept" than useful tools. As the science is advancing and new tools are developing, the field is moving beyond proofs of concept. The systems we have developed use various strategies in concert to deliver high-probability object identities.

One strategy developed inside the ESP research team involves a figure-ground segmentation algorithm [8] that distinguishes the tabletop and other surrounding contexts from the hands and the objects that the hands are manipulating. Making this distinction between figure and ground allows us to prioritize further analyses of those objects in the context that are immediately relevant.

However, we are also concerned with the objects that are not in use but are stable on the tabletop. The work we are building on looks for similarities between the objects in view of a camera and the objects known from a database [9]. ESP applications apply color matching to known exemplars, texture matches by using Scale Invariant Feature Transform (or SIFT) [10], and 2D outline matching. As an example of the benefits provided by having these algorithms working together, consider that our work has shown that with an initial figure-ground segmentation, the SIFT algorithm gives us a 91% recognition rate, while SIFT alone gives us only a 12% recognition rate [8].

Within ESP, we have also developed parallelization methods to reduce latency. Latency here is the time between video capture and the completion of analysis, and it can be an issue with interactive machine-vision-based applications such as ours. We do not want users to perceive a delay. Such latency can make an application unusable. To ensure that users will have a good experience, we have put considerable effort into making our analyses possible in "interactive time scales" [11].

In addition to knowing which objects are on the table, we also perform analyses to determine which objects are clustered together [12].

Activities

Another important part of context is the activities of the people at the table. To determine activities, a system can use the manipulated objects to infer what people are doing [13]. Objects on a table can be used to make inferences about what kinds of tasks are possible for the user or which behaviors a user may be likely to engage in. Detection and identification are made more tractable because the objects in the context constrain the set of possible activities. If a person is seen with a pen, a system can surmise that "writing" has a high probability of occurring. To this end, researchers in ESP have compiled sets of object names associated with particular activities [14].

"Facial recognition software has improved to the point where, especially when vision systems are combined with voice recognition, it is possible to determine with near-perfect accuracy who someone is."

People

Knowing who is in the context is valuable information for interaction planning. Knowing who the user is, which child, for example, allows an application to decide among different content. In addition, there may be more than one person and knowing whom the user is with—a parent, a teacher, a collaborating peer—can help the system decide what information to display. We use both faces and voices in determining who is in the context [15]. Facial recognition software has improved to the point where, especially when vision systems are combined with voice recognition, it is possible to determine with near-perfect accuracy who someone is.

Internal States

Faces can do more than just provide identity information. Faces can reveal something about how people feel. Internal states (ranging from confusion to interest and even distraction) are quite useful in determining how to support an individual trying to accomplish a task. A person's internal state—what activity they may find difficult or whether they are distracted—can be used to select a higher level of assistance or a simpler task.

"A person's internal state—what activity they may find difficult or whether they are distracted—can be used to select a higher level of assistance or a simpler task."

While a person's internal state is clearly something in the context, it may be less clear that we can use technology to "see" it. However, there is a set of emotions, sometimes called the basic emotions or Darwinian emotions, that not only are distinct but also are associated with specific muscle configurations. Paul Ekman's research over the past 40 years [16] has shown that contractions in specific sets of facial muscles, what Ekman calls "facial action units (FAUs)," are uniquely associated with particular internal states. For example, smiling involves a set of muscles around the mouth and the eyes. If a system thinks those muscles are in that particular configuration, it can infer two things: that the person is smiling and that they are happy.

The universality of facial configuration and internal state allows a vision system to make intelligent guesses as to a person's feelings. Machine-vision researchers have been building applications that recognize FAUs and categorize emotions [17]. We have built one of these applications into a version of our tutor.

Under the Hood

The perceptual system, as described, can be used to watch the objects, users, and actions. This is done to be context sensitive, make the interaction unobtrusive, and to help us actively move a user toward a goal, that is, for performance support. Thus far, the discussion has centered on sensing and context awareness much more than performance support. Performance support happens "under the hood." The performance support for our tutor is provided by an agent/model that receives the context data, makes inferences about both context and user status, and generates the actions of the system.

As noted, if a system is to go beyond "easy to use" and actively help the user, it must be context sensitive, but it also has to do more. For any performance support application to work, it must generate content (action) that is relevant to the user and the user's current context. However, the application cannot know for sure that the inferences it is making are correct. For example, the system cannot really know what the user knows. It can know with some probability that a user is confused, but there is a complementary probability that the system is wrong. Furthermore, it may not be able to observe all of the variables that are relevant in the context. For example, relevant things can happen outside the sensors' range. Therefore, the system's actions must be undertaken with a level of uncertainty. This raises a "planning under uncertainty" problem.

"The universality of facial configuration and internal state allows a vision system to make intelligent guesses as to a person's feelings."

The ESP group has used Partially Observable Markov Decision Processes (POMDPs) in its work on planning with faulty or incomplete information [18]. POMDPs are a special case of a Markov decision process. With normal Markov decision processes (MDPs), actions can be mapped with probabilities to future states; this means that system actions—the system's output to the user—have a certain probability of affecting the course of the user's future internal states, say. However, MDPs assume that every state variable can be observed without error [19], and we know we are dealing with probabilities, not absolutes. Fundamentally, we are saying that we can make a high probability inference as to where, for example, a student is in a problem-solving activity, and we are able to generate meaningful content to help that student, based on where she is. It may have a high probability of accuracy, but it is still an inference.

"The ESP group has used Partially Observable Markov Decision Processes (POMDPs) in its work on planning with faulty or incomplete information."

POMDPs are an alternative that allow for this uncertainty. A POMDP models the relationship between an agent and its environment with five variables: states, actions, observations, transition probabilities, and reward or cost functions. The system observables include the identities and relative positions of objects and the physical movements, including facial expressions, of users. Its actions are the teaching behaviors, including the hints and feedback that the system produces. ClassmateAssist's hidden states include the cognitive state of the user and the progress toward the goal. Its cost functions assume that the system should refrain from interrupting a user and assign a cost to any system-initiated vocalization or visualization.

ClassmateAssist uses sensor data to observe the environment and infer the user's internal states with respect to the task so that it might select actions to be used in its interactions. The system has a vocabulary of actions, and it makes assumptions about the probabilities with which those actions will affect the next state of the user.

The problem-solving activities and where a user is in a task jointly determine how the system chooses to interact. ClassmateAssist has to determine (with some degree of certainty) that a student is likely to solve a problem on her own and withhold hints. It also has to determine (with some degree of certainty) where she is in the task in order to know which hints or feedback might be appropriate, if necessary.

System Actions

We use the laptop to augment a student's interaction with objects on the tabletop. The system's actions are both auditory and visual. Sometimes it tells a user; sometimes it shows a user; and sometimes it does both. For telling a user, the system employs the by-now common method of a TTS engine. It has stored text that describes the tasks and that details the hints and feedback that provide the instruction. If the context is appropriate, for example, when the user has completed a task and needs a new assignment, the application chooses an assignment and sends the relevant text to the TTS.

Because hints often require a bit more than words alone can easily provide, the output of our tutor also involves visualizations of a type sometimes called mixed or augmented reality. That is, we combine graphic information generated by our system with the environment. Augmented reality uses context-awareness to supply the user with computer-generated imagery that visualizes further information about the context to the user.

We would use this, for example, because teachers in our observations frequently used gestures over the manipulatives to help students focus on particular coins, sets of coins, or features of coins. Rather than try to use text to describe the location or some feature of a coin, we have opted to use visualization techniques. The system either displays information on the screen or projects it directly onto the objects themselves.

"Augmented reality uses context-awareness to supply the user with computer-generated imagery that visualizes further information about the context to the user."

Screen-based Augmented Reality

One version of the system uses the screen of the computer to present the visualization. It shows the live video feed from the camera, which is facing the objects, and it then superimposes information about the objects onto the screen for the user. This information can be used to single out a particular object or set of objects. The on-screen augmentation can also highlight a region of the tabletop that a user is being asked to place objects within. In doing these things, the system can help in the performance of the task or it may help the user to learn something new.

Figure 1 shows a coin-sorting task where the user has been asked to move the dollar coin into the "box" displayed on the screen. We have also included a teacher avatar that watches what the user is doing and walks over to inspect and approve when the user does the correct thing.

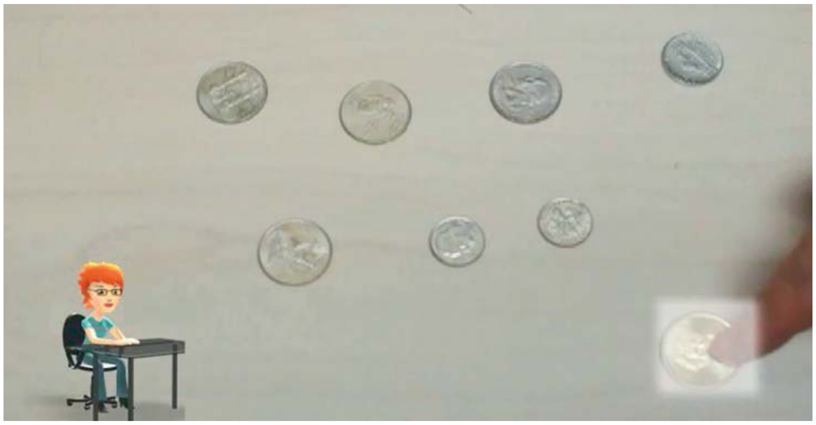

Figure 1: Screen-based Augmented Reality Interface (Simulated)
Source: Intel Corporation, 2010

This screen-based augmented reality is relatively transparent, but does involve looking at the screen rather than the objects in front of the user.

“In the typical system, a camera is used to watch objects in an area, and a projector is used to respond to movements of those objects.”

Projector-based Augmented Reality

Another way to mark up the world is to project feedback and hints onto the objects themselves. Projection plus camera systems (ProCams) have been around for some time. Their utility is described by IEEE International Workshop on Projector-Camera Systems: “Systems that utilize controllable lighting systems with light-sensing devices facilitate a wide range of applications” [20]. In most cases, “controllable lighting systems” are projectors and “light sensing devices” are cameras. In the typical system, a camera is used to watch objects in an area, and a projector is used to respond to movements of those objects. The Situated Multimedia Art Learning Lab (SMALLab) [21] is an example. SMALLab uses a room-sized area and mounts projectors and cameras many feet above the floor. The camera watches the users and the objects that the users hold, while the projector changes the images projected toward the floor, based on the movements that the camera sees. Teachers design the interactions so that the projected images and actions of the users work together to teach curricular content.

With the advent of low-power, short-throw, inexpensive pico-projectors, similar applications have become feasible in a portable unit. Sixth Sense [22] demonstrates the use of a pico-projector and camera in a wearable system that can watch the user’s hands and nearby objects and then project relevant content onto local surfaces.

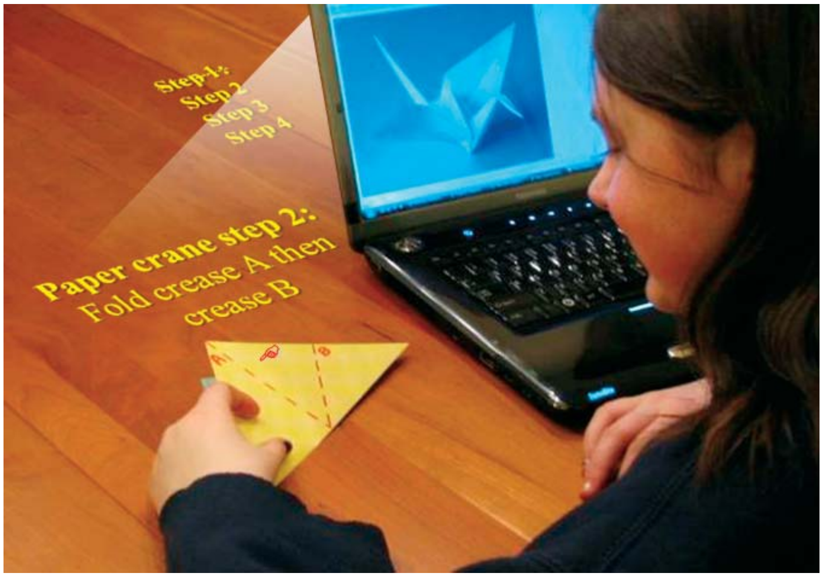

Figure 2: Projector-based Augmented Reality Interface (Simulated)
Source: Intel Corporation, 2010

Figure 2 is a simulated photo of a user who wants to do some origami. The vision system watches the paper and the user’s hands and then projects instructions onto the folded paper and the table around it.

Within our group, we have developed Bonfire, a system that augments everyday laptops with projectors, cameras, and perception software [23]. This system projects onto surfaces around a laptop and watches those surfaces for gestures or objects that are then taken as commands to the system. With Bonfire, the tutor can watch as a student moves manipulatives and then it can project content onto them. Instead of focusing on the screen as they manipulate objects, students need only look at the objects themselves.

The Future of Context

The future of context is easy to see: more, faster, better. The data rate of sensor streams will no longer present problems. Computational power will make computer vision much more commonplace. In addition, sensor fusion with its multi-modal data will also be common. Technology will be able to do more analyses at interaction speed, and the quality of the models that technology can create will improve substantially.

As our technology builds more detailed models from our experience, the depth of personalization that can be delivered will increase. A system can know what tasks users have accomplished in the past and what they might want to accomplish next. It can know what level of difficulty is suitable for them as well as what kind of support they would be best with. That is, user models will not only aggregate our past experiences but they will be able to use those data to augment our current experience. Personalized technology will be able to augment our everyday experience in a way that is uniquely appropriate just for us, because the augmentation is based on the history we share with our devices—a history made available to the technology by context awareness.

"Personalized technology will be able to augment our everyday experience in a way that is uniquely appropriate just for us, because the augmentation is based on the history we share with our devices."

Summary

This article has described the current state-of-the-art in context awareness. The capabilities of today's processors allow systems to use high data rate sensors, such as video, to provide higher levels of perception. Accurate sensing of where you are, who you are with, and what you are doing opens up the capacity to deliver new kinds of services. The Everyday Sensing and Perception project will make context awareness 90% accurate over 90% of the day. Our work has focused on the identification of objects in the environment, activities of people in the environment, personal identification of those people, as well as their affective states. By providing this level of context awareness, a system can support complex tasks. Our work on tabletop tutoring shows how this emerging capability can be used to teach users new concepts or skills.

References

[1] LaMarca, A., Chawathe, Y., Consolvo, S., Hightower, J., Smith, I., Scott, J., Sohn, T., Howard, J., Hughes, J., Potter, F., Tabert, J., Powledge, P., Borriello, G., and Schilit, B. "Place Lab: Device Positioning Using Radio Beacons in the Wild." *IEEE Pervasive*, 2005.

[2] Chang, K., Hightower, J., and Kveton, B. (2009). "Inferring Identity using Accelerometers in Television Remote Controls." *International Conference on Pervasive Computing*, 2009.

[3] Bezanson, W. *Performance Support Solutions: Achieving Goals Through Enabling User Performance.* Trafford Publishing, Victoria, B.C, Canada, 2002.

[4] White, S., Marino, D., and Feiner, S. "Designing a Mobile User Interface for Automated Species Identification." *ACM CHI*, 2007.

[5] Silva, E. "On the clock: Rethinking the way schools use time." *Education Sector*, 2007.

[6] "American Time Use Study." U.S. Bureau of Labor Statistics (2006). Available at ***http//www.bls.gov/tus/home.htm***

[7] "Student Work and Teacher Practices in Mathematics." *National Center for Educational Statistics: US Department of Education,* 1999. Available at ***http://nces.ed.gov/pubsearch/pubsinfo.asp?pubid=1999453***

[8] Ren, X. and Gu, C. "Figure-Ground Segmentation Improves Handled Object Recognition in Egocentric Video." To appear in *CVPR* 2010.

[9] Rahimi, A. and Lee, S. "Visual Object Instance Recognition." Available at ***http://people.csail.mit.edu/rahimi/projects/objrec/***

[10] Lowe, D. "Object recognition from local scale-invariant features." *International Conference on Computer Vision,* 1999.

[11] Pillai, P., Mummert, L., Schlosser, S., Sukthankar, R., and Helfrich, C. "SLIPstream: Scalable Low-latency Interactive Perception on Streaming Data." *NOSSDAV,* 2009.

[12] Shi, J. and Malik, J. "Normalized cuts and image segmentation." *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 22(8), 2000.

[13] Philipose, M., Fishkin, K.P., Perkowitz, M., Patterson, D.J., Fox, D., Kautz, H., and Hähnel, D. "Inferring activities from interactions with objects." *IEEE Pervasive Computing,* pages 50–57, October 2004.

[14] Wu, J., Osuntogun, A., Choudhury, T., Philipose, M., and Rehg, J. "A Scalable Approach to Activity Recognition Based on Object Use." *ICCV,* 2007.

[15] Choudhury, T., Clarkson, B., Jebara, T., and Pentland, A. "Multimodal Person Recognition using Unconstrained Audio and Video." *Audio- and Video-based Biometric Person Authentication,* March 1999.

[16] Ekman, P. and Friesen, W. V. *Unmasking the face. A guide to recognizing emotions from facial clues.* Prentice-Hall, Englewood Cliffs, New Jersey, 1975.

[17] Bartlett, M., Littlewort, G., Lainscsek, C., Fasel, I., Frank, M., and Movellan, J. "Fully automatic facial action recognition in spontaneous behavior." *International Conference on Automatic Face and Gesture Recognition,* 2006.

[18] Theocharous, G., Beckwith, R., Butko, N., and Philipose, M. "Tractable POMDP Planning Algorithms for Optimal Teaching in 'SPAIS'." *International Joint Conference on Artificial Intelligence* (IJCAI) workshop on Plan Activity, and Intent Recognition (PAIR), July 2009.

[19] Blythe, J. "An Overview of Planning Under Uncertainty." *Lecture Notes in Computer Science,* 1600:85-110, 1999.

[20] *ProCams 2010.* Retrieved from ***http://www.cs.cmu.edu/~ILIM/ProCams2010/***

[21] Birchfield, D., Megowan-Romanowicz, C., and Johnson-Glenberg, M. "Next Gen Interfaces: Embodied Learning Using Motion, Sound, and Visuals–SMALLab." *American Educational Research Association Annual Conference* (AERA); *Applied Research in Virtual Environments for Learning* (ARVEL), 2009.

[22] Mistry P. and Maes, P. "SixthSense – A Wearable Gestural Interface." *SIGGRAPH Asia, Sketches,* 2009.

[23] Kane, S., Avrahami, D., Wobbrock, J., Harrison, B., Rea, A., Philipose, M., and LaMarca, A. "Bonfire: a nomadic system for hybrid laptop-tabletop interaction." *ACM Symposium on User Interface Software and Technology* (UIST), 2009.

Acknowledgments

We thank the entire ESP team for all of their tremendous work and supplying some of the coolest toys. We thank Nick Butko, Vimal Mathew, Nicol Schmied, and Dan Teibel for their help with the tutoring system, Tye Rattenbury for running the analyses on the American Time Use Survey, Kieran Del Pasqua for the screen-based augmented reality images, and Olivia Laing and Anthony Lamarca for helpful comments on earlier drafts.

Author Biographies

Richard Beckwith is a Research Psychologist with FTR's People and Practices Research group. He focuses primarily on education, agriculture, and privacy. He received his PhD from Teachers College, Columbia University in 1986. From 1986 to 1991 he was a research scientist at Princeton University's Cognitive Science Lab, working on WordNet. From 1991 until coming to Intel, he was a Research Faculty member at Northwestern University in the Institute for the Learning Sciences (ILS). His e-mail is richard.beckwith at intel.com.

Georgios Theocharous received his PhD degree in Computer Science in 2002 from Michigan State University. From 2002 to 2004 he was a post-doctoral associate at the Computer Science and Artificial Intelligence Lab at MIT, and in October 2004 he joined Intel as a Research Scientist. His research interests include computational models of learning and planning under uncertainty and their applications to the real world. Specific models include reinforcement learning, completely and partially observable Markov decision processes (POMDPs), semi-Markov decision processes, hierarchical POMDPs, and dynamic Bayesian nets. His e-mail is georgios.theocharous at intel.com.

Daniel Avrahami is a Research Scientist at Intel Labs Seattle. He received PhD and MSc degrees in Human-Computer Interaction from the HCI Institute at the School of Computer Science at Carnegie Mellon University and a BSc degree in Computer Science from the Hebrew University in Jerusalem, Israel. His current research looks at context-aware applications for everyday life. His other research interests include the design and implementation of new intelligent communication tools. His e-mail is daniel.avrahami at intel.com.

Matthai Philipose co-leads the Everyday Sensing and Perception project out of Intel Labs, Seattle. He builds systems that observe and reason about human activity, emphasizing detailed but tractable models of activity, very high-density sensing, and automatically acquired common sense. He is broadly interested in statistical reasoning, logic, and programming languages. He has a BS degree from Cornell University and a PhD degree from the University of Washington, both in Computer Science. His e-mail is matthai.philipose at intel.com.

Copyright