about cyborgs and gargoyles - state of the art in wearable computing

Kai Kunze
Embedded Systems Lab, University Passau
http://wearable-computing.org/
http://wearcomp.eu/
http://twitter.com/kai_ser/
{first.lastname@uni-passau.de}

This paper gives you an introduction to wearable computing. It starts of with a small introduction concerning the terms, then moving over to some concrete application scenarios, from a maintenance to firefighters. Subsequently, we discuss some tools used to build wearable, context-aware systems. Finally, we take a look at some problems concerning the current systems and privacy implications.

Track [**Science**]: Language—*English*

1. INTRODUCTION

With the widespread integration of embedded processors and the proliferation of mobile devices, the vision of pervasive/ubiquitous/wearable computing becomes more and more feasible: computing that helps us transparently in our everyday life, focusing on the our needs without disturbing us. In the following pages, I try to give a brief overview of the field and then go into details about some of our efforts towards this vision using concrete application examples.

Personally, I do not see too much differences between the visions outlined in ubiquitous, pervasive, wearable computing and ambient intelligence. Each discipline just focuses more or less on particular aspects of the technology.

Wearable computing is a nice term as it describes the ideas in a simple way. Computing as useful and unobtrusive as clothes.

The pervasive/wearable computing field is as such interdisciplinary. It combines aspects of embedded systems, signal processing, artificial intelligence, machine learning and human computer interaction.

Subsequently, I will discuss some of the basic assumptions of the field, giving some application scenarios we work on at the Embedded Systems lab. Afterwards, I give details on how current pervasive systems are prototyped and implemented. Finally, I go into some of the technical challenges we still face in this field.

Of course, all of the technologies introduced in the paper have some pretty strong privacy implications. On the one hand, computing that knows what the user is doing can be a bless, as it is able to help him to accomplish the tasks he is working on. On the other hand, it can track and record every move the user does and therefore implement an Orwellian Big Brother. I will shortly rough out some of the ethical issues in the last section. The content of this paper is a summary of papers given in the bibliography.

Regarding the title 'About cyborgs and gargoyles', sadly, I have not much background in implantable sensors, therefore I won't discuss them here. Although implantables are extremely interesting, so far, the user acceptance is very low except in the medical domain. The title is, I think, still justified as having wearable technology to support us is the first step to integrate computing seamlessly in our life, as extensions to our sensory inputs.

2. CONTEXT RECOGNITION IS THE KEY

We basically want use the already existing substrate of microprocessors and sensors embedded in our environment to infer relevant information about the users current activities and state. This "relevant information" is often referred to as context. Detecting this relevant information is then defined as 'context recognition'. There are some debates in the research community on what context actually is. For the remainder of this paper, I will use the very practical, broad definition given above. We have the infrastructure in place given from today's embedded systems that helps us gather information about our current activities and state. There are plenty of papers([5,9,12]) describing context recognition systems. The sensors used for such systems vary substantially, from video cameras used for computer vision, over microphones to inertial motion sensors. One can classify them tough into two distinct types:

—infrastructure sensing: Stationary sensors like a microphone array, a stationary camera, a radio frequency localization system like Sputnik etc.

—onbody sensing: sensors worn on the user's body, for example: inertial motion, (accelerometers, gyroscopes etc.), galvanic skin response sensors etc.

Using a combination of these sensors as information sources, we should be able to support any high level user acitvity, from work and collaboration over everyday living to healthcare and game/fitness applications.

2.1 application scenarios

The idea to use context recognition systems, that help the user in everyday activities seems promising. Yet, how do such system examples look like? In this section, I describe some concrete application scenarios from research projects implemented by the Wearable Computing Lab, ETH Zuerich [1] and the Embedded Systems Lab [2], Uni Passau. Most of them are taken from the WearIT@Work project, an EU Project with over 50 partners in academia and industry (see [10] and http://www.wearitatwork.com/) and RELATE, a smaller EU project focusing on baseline research (see http://eis.comp.lancs.ac.uk/relate/).

2.2 search and rescue

The first application scenario we look at is very challenging, supporting firefighters during rescue operations. Today, firefighters use ropes in cases of impaired visibility to mark paths taken into buildings to help them in finding the exit again. They call these ropes 'lifelines'. For many situations, this technique works reliably well. Yet,

[1] http://www.wearable.ethz.ch/
[2] http://esl.fim.uni-passau.de

Figure 1: A first generation RELATE sensor node placement on the floor of the training facility and a firefighter equipped with several sensor nodes

there are also serveral limitations to the rope 'lifeline'. The lifeline can get stuck or be cut under doors or other objects and it limits the effective range of the firefighters. In the WearIT and RELATE projects, we developed the use sensor networks for a virtual lifeline. Such that, firefighters automatically deploy sensor nodes along their paths, using a dispenser mechanism, establishing an ad-hoc infrastructure for positioning, sensing and communication. The firefighters can interact with this sensor network by way of wearable computing equipment and receive navigational information on e.g. a head-mounted display, over a headset or tactile feedback. The technology is to enable automatic tracking of a building's floor plan, tagging dangerous areas, checking rooms that have been searched for unconscious persons etc. Although we showed a first prototype implementation at Ubicomp 2007 [6], we are still far away from a system that is able to support firefighters in a real emergency.

2.3 healthcare and hospitals

The next application area, we take look at is in the hosptial and pervasive healthcare area. As you know, hospitals are a highly complex environment. Although there is already a quite some IT infrastructure at place, traditional interaction with a computer system is not really possible during most medical procedures. One of the problem areas my colleagues tackled over a period of 3 years, is the daily ward rounds of doctor and nurse teams. Currently, most of the interaction is still on paper.

In the system my colleagues proposed, implemented and tested, the doctor and nurse teams move around without paper,laptop or filing card. The doctor wears a wrist band (inertial motion sensor + rfid reader) and a qbic belt intergrated linux pc, depicted in Figure 2. The doctor identifies the patient by means of his RFID reader on his wrist. The relevant patient files appear on the screen, attached to the bed. The doctor immediately gets an overview of all the important data about the patient, With the inertial sensor attached to his wrist, he can navigate through the application without a mouse or keyboard. This is keeping his hands sterile and

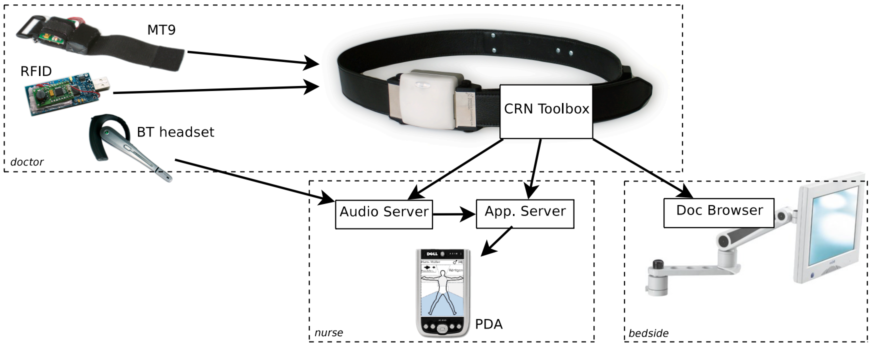

Figure 2: System architecture for ward rounds

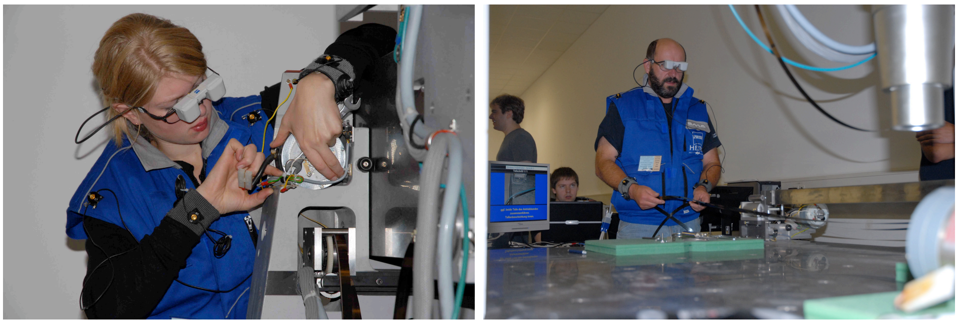

Figure 3: Pictures from the experimental sessions for the maintenance scenario. From left to right: Installing the xmotor and the ygears.

free to examine the patient at any time.

The basic system architecture is given in Figure 2. The system was evaluated over several weeks in a hospital in Steyr, Austria, using real doctor/nurse teams and patients (see [1]).

2.4 maintenance

Another application focus of our work is the domain of wearable maintenance systems.There are many such systems proposed and implemented since the early days of wearable computing. These systems aim to provide maintenance personnel with access to complex electronic information with as little interference as possible to the primary task at hand. Typically, they rely on head mounted displays (often with augmented reality), input modalities that minimize hand use (e.g. speech, special gloves) and interfaces that aim to reduce the cognitive load on the user. It is widely believed that wearable maintenance systems can benefit from automatic work progress tracking. Main uses for such tracking are just in time automatic delivery of information (see the manual page that you need without having to explicitly demand it), error detection (e.g. you forgot to fasten the last screw), and warnings (e.g. do not touch this surface).

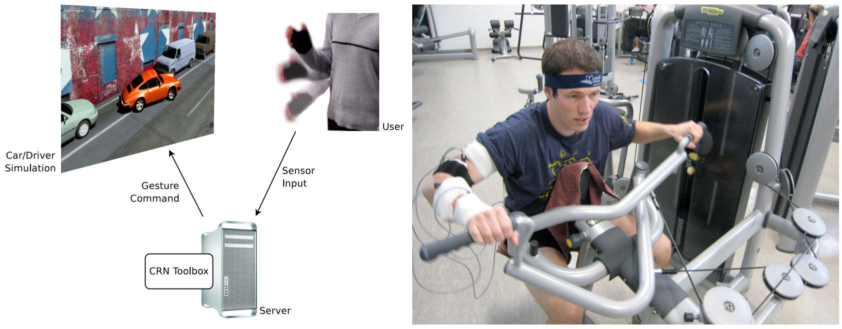

Figure 4: Car parking game, You are the person helping a virtual driver to park a car using hand gestures and a picture from data recordings to recognize gym exercises

In a paper accepted for pervasive 2009, we show a quantitative study on the usefullness of context recognition in a real world maintenance environment [8].

2.5 games and sports

Of course, wearable systems can also support gaming and sports applications. Although there are already several gaming platforms on the market that follow this apporach (Sony's eyetoy, Nintendo's Wii). Their 'recognition' capabilities are very limited and users cannot use their expertise gathered by doing the real sport in game. We have shown how to recognize difficult gestures online to enable car parking game [2]. Some of our efforts in this area focus on the recognition of martial arts movements and gym exercises.

3. HOW DOES IT WORK?

Today, the development of context recognition systems is mostly done in two phases. First the recognition method (sensor setup, feature set, classifiers, classifier parameters, fusion methods etc.) is designed. Researchers apply algorithms from the fields of signal processing, machine learning, artficial intelligence. In this first phase experimental data is mostly fed offline into conventional rapid prototyping tools such as Matlab. These tools provide a rich reservoir of 'off the shelve', parameterizable algorithms and visualization methods. Thus, different system variants can be tested quickly without the need for time consuming implementation work.

Unfortunately, most such simulation environments are not suitable for actually running applications, especially in mobile and pervasive environments. In general, they depend on custom 'engines' or libraries requiring large memory footprints and high computing power. Consequently, the implementation of activity recognition applications is mostly done in a separate, second phase. The selected algorithms are implemented in an appropriate programming language and then distributed to specific devices.

We actively develop the Context Recognition Network (CRN) Toolbox ([4] and [3]) (available under LGPL from http://crnt.sf.net) to combine the two phases and to permit quick construction of complex multi-modal context recognition systems, that can be immediately deployed in the targeted environment. The CRN

Toolbox has been used to develop the protoypes for the application scenarios discussed above.

3.1 practical tips to build context recognition systems

First of all, you can obtain the source code and examples of some of the systems presented above by checking out the toolbox source from http://crnt.sf.net . Here you can also find some more information about the supported platforms. You should get it working on all systems that have POSIX Threads. However, please understand that the CRN toolbox is in the very first stages of development, this means that documentation and examples are still lagging.

If you are in genereal interested in signal processing and machine learning algorithms, you might want to take a look at the tutorials of the 'Intelligent Systems' course I teach this semester at the Unversity Passau (http://esl.fim.uni-passau.de/wiki/index.php/Intelligent_Systems_Winter_2008).

Unfortunately, most of the tutorials still use Matlab, which is quite expensive and therefore not so easy to obtain for a hobby researcher. Yet, I started to include the equivalent scripts in python using scipy. We are in process of switching to it for the tutorials, although it might take a while until all of the scripts are ported. For people interested in free data exploration tools, I can recommend the following:

—Scientific Python, great package with nearly everything you need to get going http://scipy.org/

—Matplotlib, nice plotting library. http://matplotlib.sourceforge.net/

—Sage uses scipy, matplotlib and other packages good alternative to maple, matlab etc. http://www.sagemath.org/

The weka machine learning package in Java is also a good toolkit for alogrithms.

4. BECOMING OPPORTUNISTIC

Another issue away from rapid prototyping and easy testing, we are currently struggling with, is the augmentation problem. Most of the systems introduced above work well if you use dedicated sensors with a fixed placement and known orientation. Most inference algorithms cannot cope with failing or displaced sensors. This is actually my personal field of interest and the focus of my phd. work. There are a couple of interesting approaches to get over these limitations (for example [11]). In [7] we show how one can recognize the surface on which a mobile phone is placed using the virbration motor and frequency beeps. Unfortunately, these algorithms are not yet very practical, as they still run offline using scipy and matlab and cannot be implemented on commodity embedded hardware.

5. PRIVACY IMPLICATIONS

As this is a technical paper, I cannot go much into the ethical issues and problems associated with this technology. Also, this is not really my expertise. Yet, anybody will agree that wearable computing raises some severe ethical problems. On the one hand, the system that can help a user avoid mistakes and errors is able, on the other hand, to report error statistics to his employee etc. I want to raise awareness to these issues, as a lot of people are uneasy with the installment of video cameras or similar. I believe that camera surveillance is not so much a danger to our privacy, than much simpler sensors and technology that can be easily embedded in mobile phones or other household devices. This is partly due to the fact, that computer vision has severe limitations (light, angle etc.) compared to the fusion of several, more simple "sensor" signals. For example, to track a person's purchasing habits in the city, it is way easier to use the credit card record combined with the cell log of the mobile phone than cameras distributed downtown.

REFERENCES

[1] K. Adamer, D. Bannach, and P. Lukowicz. Developing a wearable assistant for hospital ward rounds: An experience report. In *Proceedings of the International Conference for Industry and Academia on Internet of Things*, 2008. To appear.

[2] D. Bannach, O. Amft, K. Kunze, E. Heinz, G. Tröster, and P. Lukowicz. Waving real hand gestures recorded by wearable motion sensors to a virtual car and driver in a mixed-reality parking game. In *Proceedings of the IEEE Symposium on Computational Intelligence and Games*, pages 32–39, 2007.

[3] D. Bannach, O. Amft, and P. Lukowicz. Rapid prototyping of activity recognition applications. *IEEE Pervasive Computing (to appear)*, April-June, 2008. to appear.

[4] D. Bannach, K. Kunze, P. Lukowicz, and O. Amft. Distributed modular toolbox for multimodal context recognition. In *Proceedings of the 19th International Conference on Architecture of Computing Systems*, volume 3894 of *LNCS*, pages 99–113. Springer, 2006.

[5] N. Kern, B. Schiele, H. Junker, P. Lukowicz, and G. Tröster. Wearable sensing to annotate meeting recordings. *Personal and Ubiquitous Computing*, 7(5):263–274, 2003.

[6] M. Klann, T. Riedel, H. Gellersen, C. Fischer, M. Oppenheim, P. Lukowicz, G. Pirkl, K. Kunze, M. Beuster, M. Beigl, O. Visser, and M. Gerling. Lifenet: an ad-hoc sensor network and wearable system to provide firefighters with navigation support. In *Adjunct Proc. Ubicomp 2007*, pages 124–127, September 2007.

[7] K. Kunze and P. Lukowicz. Symbolic object localization through active sampling of acceleration and sound signatures. In *Proc. 9th Int. Conf. on Ubiquitous Computing - Ubicomp 2007*, Innsbruck, Austria, 2007.

[8] K. Kunze, F. Wagner, E. Kartal, and P. Lukowicz. Does context matter ? - a quantitative evaluation in a real world maintenance scenario. In *Pervasive Proceedings*, 2009. To appear.

[9] J. Lester, T. Choudhury, and G. Borriello. A practical approach to recognizing physical activities. *Proceedings of Pervasive*, Jan 2006.

[10] P. Lukowicz, A. Timm-Giel, M. Lawo, and O. Herzog. Wearit@work: Toward real-world industrial wearable computing. *IEEE Pervasive Computing*, 6(4):8–13, Oct-Dec 2007.

[11] D. Minnen, C. Isbell, I. Essa, and T. Starner. Discovering multivariate motifs using subsequence density estimation and greedy mixture learning. *Artificial Intelligence Proceedings*, Jan 2007.

[12] C. Randell and H. Muller. Context awareness by analysing accelerometer data. *The Fourth International Symposium on Wearable Computers*, 1(1):175–176, 2000.