

# Interacting with Proactive Community Displays

Nicolas Villar, Albrecht Schmidt, Gerd Kortuem, Hans-Werner Gellersen

*Computing Department*

*Lancaster University*

*Lancaster LA1 4YR*

*United Kingdom*

*{villar, albrecht, kortuem, hwg}@comp.lancs.ac.uk*

## Abstract

*In this paper we describe the design and architecture of an adaptive proactive environment in which information, which reflects the communal interests of current inhabitants, is proactively displayed on large-scale public displays. Adaptation is achieved through implicit communication between the environment and personal sensor devices worn by users. These devices, called Pendle, serve two purposes: they store and make available to the environment user preferences, and they allow users to override the environment's proactive behavior by means of simple gestures. The result is a smooth integration of environment-controlled interaction (experienced by the user as implicit interaction, triggered by their presence) and user-controlled explicit interaction. Initial results show that user-controlled adaptation leads to an engaging user experience that is unobtrusive and not distracting.*

## 1. Introduction

Communal environments offer great opportunities for informal interaction and communication. However, the informative aspects of the environment - in the shape of posters, screen displays or anything able to display information - do not always make best use of their display real-estate. Traditional static media such as posters quickly become outdated, their informative purpose outlived as they become commonplace. Screen savers often occupy computer screens that are left on, leaving unused a space that could be useful in displaying information unobtrusively in the periphery of the environment. We looked at some of the communal spaces in our department and found that although some of them were equipped with valuable display resources, such as projection equipment and large plasma screens, this equipment remained unused as gaining access to information on these displays was complicated.

As a result, we became interested in looking at how we could facilitate interaction between mobile users and display resources in their ambient environment. Our goal

is to enable users to take better advantage of display resources that are available in their environment. To that effect, we developed a proactive environment where individual users and groups of users can interact implicitly and explicitly with display services in their vicinity by means of small, personalized, wearable devices, called *Pendle* (Figure 1). A *Pendle* mediates between user and proactive environment. It serves two purposes: it holds personal information to implicitly trigger proactive behavior in the environment, and it provides a gesture-based interface for the user to explicitly modify the environment's.

The environment contains several large-scale public displays that proactively display information from the web that is relevant to the users near them. This is done in a spontaneous fashion, in a manner similar to MIT's Community Portals [3], although our design relies on the internet as a source of information and the content to display is chosen based on keywords that represent individual users' interests. Similarly also to the Teleporting system developed at the Olivetti Research Laboratory [1], our system allows users to quickly gain access to personal information throughout different displays and environments on demand. They are able to interact with the displays and get fast access to a small number of pre-selected internet. The result of our design is a smooth integration of environment-controlled interaction (experienced by the user as implicit interaction, triggered by their presence) and user-controlled explicit interaction.

## 2. Concept and System Architecture

The starting point of our design is the idea to utilize existing screen real-estate of public displays and desktop computers as proactive display space.

Our main design goals are:

- Displays show content which reflects the communal interests of people in their vicinity
- The content on the displays changes without users intervention



**2.2.3. Finding content.** At the centre of the system is a component that transforms the information that is continuously broadcasted from the wearable devices into meaningful content, which can then be shown on the display services.

The basic function of this component is to relate messages from the wearable device to meaningful URLs which can be displayed. Dependent on the messages received this may only be forwarding a URL received at a particular location to a selected screen. If, instead, keywords are received from the wearable devices, the system uses available information sources - such as a search engine - and finds appropriate content. This content relates to the set of search terms received in a particular spatial area (e.g. a room) in the building.

**2.2.4. Receivers, Connectors and Gateways.** To realize such a system, the information that is wirelessly transmitted from the wearable devices has to be handed to a module which finds appropriate content, and this content description has to be given to the appropriate display service.

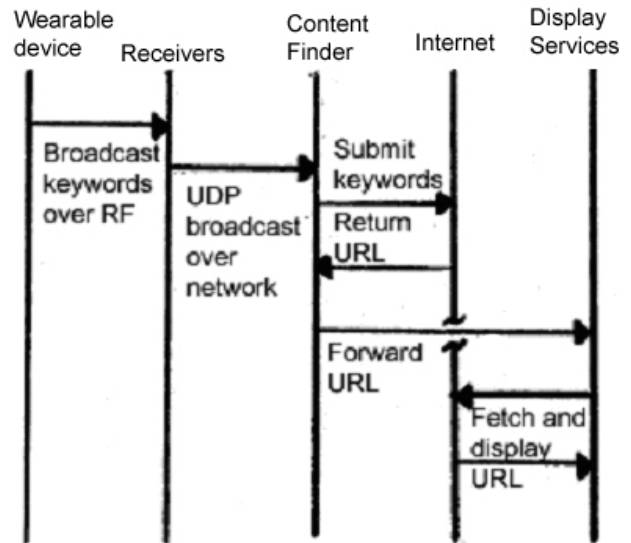
The physical location where the information from the wearable device is received, as well as the physical location of the display service is of great importance. In contrast, the module which relates received messages to meaningful content can be situated anywhere.

To facilitate communication, receivers for the wireless communication are connected to the local network. Messages received are then extended by the physical location of where they were received and broadcasted into the local network as UDP packets.

The module that relates information to content receives these messages resolves them into an URL and sends it to the most appropriate display service. The content pointed to by this URL is then fetched directly by the display service using its available network connection.

## 2.2 Data Flow

The data flow between *Pendles* and infrastructure is shown in Figure 3. Users wearing the device broadcast chosen keywords or URLs. These messages are picked up by receivers in the area which is covered by the wireless transmission. Each receiver broadcasts the messages received, extended by an ID and the physical location to the local Ethernet. In the network one or more modules receive the messages collected by the receivers and can determine whether or not to use them and to transform them into valid URLs. After creating a valid URL, a display service (or a set of them) is selected and a request to display the URL is sent. The valid URL is then received by a display service and - depending on the state of the service at that time - the URL content is fetched and displayed.



**Figure 3. Component interaction**

In the following sections we describe in more detail the design and implementation of the wearable device, the content finder, and the display service.

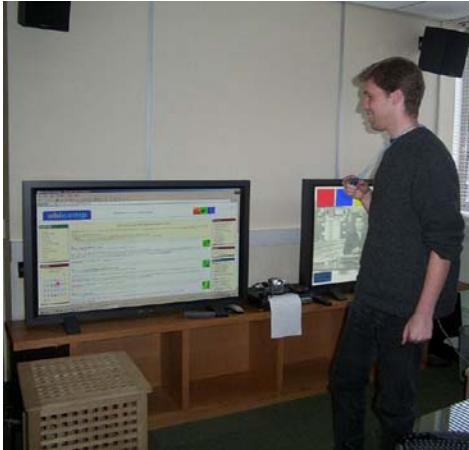
## 3. The *Pendle* Wearable Device

We designed a wearable device - called a *Pendle* - to serve as mediator between users and proactive environments. By wearing it the user influences the environment's proactive behavior. It also serves as a point of interaction by providing a simple user interface to services in the environment. The name *Pendle* allures to its realization in the form-factor of a pendant, as well as the fact that environment services are dependant on this device in order to operate.

### 3.1 Concept and Basic Operation

The *Pendle* is a personal wearable device. It stores a user's predefined information which it can wirelessly transmit to the environment when it is being worn. This enables services to 'sense' the user's presence and preferences. An interface for explicit interaction with these services is provided through gesture recognition feature built into the device. Gestures can be associated with commands that can be performed on specific services.

We chose to realize the *Pendle* as a badge on a ribbon that can comfortably be worn around the neck. The *Pendle* is a self-contained device with a small size and weight in order to make it a relatively unobtrusive item to wear, and the ribbon allows the user to comfortably hold and manipulate the object while wearing it.



**Figure 4. User interacting with Proactive Displays**

Users interact implicitly with the public displays services in the environment simply by wearing their *Pendles*. They can manipulate it by performing gestures to explicitly control the displays and to access the content of a predefined set of information.

Although similar to the Gesture Pendant [2] in concept and form-factor, we wanted to use the *Pendle* as more than an artifact that provides interaction with our system. We wanted to use the fact that users would wear the *Pendle* as a way to measure some of their implicit actions and therefore allow the system to proactively respond.

### 3.2 Interaction Modes

The *Pendle* is a device for interacting with services provided by an augmented environment. It supports three interaction modes:

**Inactive:** If the device is not being worn it is inactive. In this mode no interaction takes place between the device and the environment.

**Implicit:** In implicit mode the *Pendle* wirelessly transmits information chosen by the user to the environment. This mode is in effect as long as the device is being worn by the user (unless the explicit mode becomes active). The environment services use the information contained in the profile to adapt their behavior to suit the user.

**Explicit:** The explicit mode is in effect whenever the user performs gestures with the device. Each gesture represents a specific command. As soon as the *Pendle* recognizes a gesture, it transmits the corresponding command to the environment.

## 4. The *webPendle* Application

We developed an application called *webPendle* where the *Pendle* device and related infrastructure is used to create interesting proactive/interactive informative environments. In our application display resources – such as large plasma screens or PC monitors - in the environment proactively display information from the web that is relevant to the users near them. This has to be done in a way that the system implicitly displays websites that might be interesting, while also allowing the user to explicitly request a particular website of their choice.

### 4.1 Keywords to enable implicit behavior

In order to provide a satisfactory user experience while the system behaves proactively, it is important that the system has a good idea of a user's likes and dislikes. In order to do this, the user describes herself as a set of keywords to be stored on the *Pendle*. These keywords, along with the keywords of other users in the environment, will be used to find content that should reflect the communal interests.

### 4.2 Bookmarks for explicit interaction

The users are also able to use the *Pendle* to call up predefined URLs to be displayed, similar to accessing websites through a web-browser's 'favorites' or bookmark list. In theory this list could be linked and updated from the user's web browser on their PC, allowing them to carry their bookmarks with them.

### 4.3 The meaning of gestures

To provide a vehicle for explicit interaction with the system, we assigned the following meanings to simple gestures that can be performed with the *Pendle*:

- Wearing the *Pendle* marks the beginning of the implicit interaction.
- Picking up the *Pendle* marks the beginning of the explicit interaction.
- Holding up the *Pendle* causes display services in the vicinity to display the next favorite URL stored in the its memory
- Shaking the *Pendle* removes URLs that are currently on display.
- Letting go of the *Pendle* marks the end of the explicit interaction.
- Taking off the *Pendle* marks the end of the implicit interaction.

By limiting the number of gestures to a small number, we are able to recognize these with almost perfect accuracy while requiring a minimal device implementation. This is further described in section 5.1.

## 4.4 Displaying content

The display service moderates what content is shown, and for how long it is displayed. It displays both explicitly requested and implicitly (keyword-based) URLs. By default, the display service acts in a proactive manner, while explicit requests for URLs cause interactive behavior.

URLs that have been marked by the content-finder as being the result of a keyword search are fetched from the internet by the display service and displayed for at least two minutes. After this period, the content is replaced by new incoming URLs. This is done in order to maintain a dynamic informative environment that is highly relevant to users in the vicinity, while at the same time ensuring that the information is not changed so frequently as to prove distracting.

URLs that are marked as being the result of a user making an explicit gesture have priority over implicit-URLs, and will display immediately upon reception of a request. Also, a “remove URL” command (shaking the *Pendle*) or a user leaving the environment causes any content explicitly requested by that user to disappear and the service goes back to displaying keyword-based URLs.

In the case when the URL points to a media clip, the media clip will be played in its entirety and then the service will go back to displaying the last static URL received.

## 5. Implementation

This section describes our implementation of the *webPendle* application as was deployed in a communal recreation area of our department.

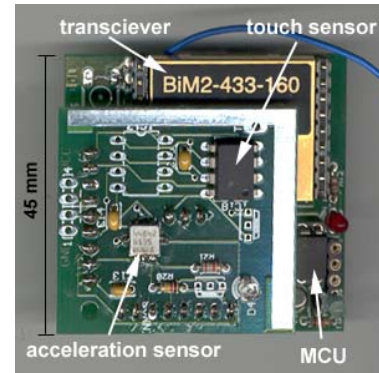
### 5.1 Pendle Implementation

We implemented a prototype of the *Pendle* using a mini Smart-Its device coupled with a sensor board [6] (Figure 5, Figure 6). The Smart-It core provides a PIC microcontroller capable of storing and running small programs. It also has 8K of non-volatile RAM through an I2C connection, a wireless transceiver and a 3V lithium cell.

The smart-its core board is augmented through an add-on board with two sensors: a touch sensor (QT110) and an ADXL202 that provides two axis of acceleration. The combined components are 45mm x 45mm x 20mm, weighing about 40g.

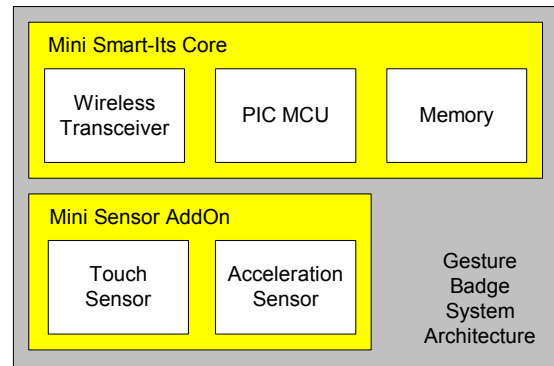
We use the acceleration sensor to measure both static and dynamic acceleration. Dynamic acceleration is caused by the user or *Pendle* moving, and is useful for determining the implicit context (sitting, moving) and also some explicit gestures, like shaking the *Pendle*. Static acceleration refers to the reading that gravity

produces on the sensor. The touch sensor is used to determine when the user is holding the *Pendle*, important for establishing where the explicit interaction begins and ends.



**Figure 5. *Pendle* implementation using a Smart-Its Device with Sensor add-on.**

As the *Pendle* is designed to be able to work independently from its environment, algorithms cannot rely on off-board processing and therefore must be kept minimal. A peak-based feature extraction method similar to the one described in [21] is used for this task. By supporting only a small number of possible gestures it is possible to detect these with very high accuracy.



**Figure 6. *Pendle* Device Architecture.**

The non-volatile memory on the smart-its core is used to store the keywords and URLs as defined by the user. Although we considered the possibility of using the *Pendle* as a pointer to a place on the network or internet where this information could be stored - in a manner similar to the Active Badge [4] - we chose to store the data directly on the *Pendle*. We liked the idea of users carrying around the data in their physical devices as it allowed for an easier use of the *Pendle* over different environments. It also facilitates some of the future improvements we have considered for the system, such as the ability to exchange URLs or keywords with other

users without the need for an internet connection to be present, as exchange of information is possible between *Pendles* without the need for external infrastructure.

Our system also differs from the Active Badge in that it operates anonymously. Particularly in proactive operation, there is no need to differentiate between individual users as the information displayed is the result of the shared interests of the people in the environment. Pseudonyms or IDs are assigned to every *Pendle* to allow for explicit control, but they are not necessarily linked to a user's actual identity. The anonymity option was introduced to deal with the privacy concerns that users of a system like this are likely to have.

## 5.2 Receivers

Receivers are the link between the wireless communication used by the *Pendles* and the local Ethernet network used by the rest of the components. As such, they are deployed throughout the environment where the display services reside.

The receivers in our implementation are Smart-Its devices connected to PCs already present in the environment that provides the Ethernet connectivity. We are also experimenting with devices that can directly interface with the LAN, removing the need for a PC. The software running in the MCU of the device simply receives wireless transmissions, augments them with the ID of the receiver and broadcasts it onto the network to be picked up by the Content-Finder module.

## 5.3 Content-Finders

We implemented the content-finder as a program that resides on a machine connected to the same local network as the receivers and display services. Its physical location is not important.

The program listens on a defined network port for any broadcasts from the receivers. We chose to use UDP as the communications protocol as it easily enables us to broadcast the information to several content-finders on the same network. As it is intended for use over a local area network, we expect no problems in terms of packet loss.

The content-finder uses a table that maps receiver ID's to available display services. It is used to determine which display services to send particular commands to. This is initially a static mapping, but a possible improvement would be to examine ways in which this table could be dynamically updated as new devices become available on the network. In our implementation, the table simply maps receivers to display services in their same physical location.

In the case of URLs or commands being received, they are simply forwarded on to the display services detailed on the available services table. These are marked

as being the result of an explicit interaction, to differentiate them from implicit-URLs to allow the display services to recognize them as such and deal with them accordingly.

However, any keywords that are received are placed into an array. They are maintained in the array as long as they are periodically resubmitted by the *Pendle*, otherwise their validity expires five minutes after the last submission, and they are removed. This is done to maintain a dynamic and highly relevant keyword pool, reflective of the chosen interests of people in the environment or ones that have recently been in the environment.

We then faced the task of converting these keywords into valid URLs that could be retrieved and rendered by the display services. Our first approach was to randomly submit a few keywords from the pool to the Google search engine (using the Google Web API). The most relevant URL to be found – according to Google's ranking system – is chosen to be displayed. This resulted in varied if somewhat random and sometimes unrelated content being displayed. Other approaches were also examined, for example the algorithm outlined in Table 1 which yielded more relevant results.

1. Randomly select 10 keywords from the pool
2. Submit keywords to search engine
3. If no URL is retrieved, go to step 5.
4. If URL has low relevancy score, go to step 5. Else, go to step 6.
5. Remove one keyword from submission list. Go to step 2.
6. Fetch and display URL.

**Table 1. Content-finding Algorithm**

By implementing different keyword-to-URL translation methodologies, the system can be changed either to implicitly display highly relevant content for one or a few of the users in the environment, or more vaguely relevant content for a wider number of users.

This program is intended to operate autonomously and require no front-end or administration. However the location handling and search strategies can be modified to fine-tune the behavior of the system.

## 5.4 Display Services

We developed a program to encapsulate the services offered by both the Microsoft Internet Explorer Control

and the Microsoft Media Player Control in a single application. Our program is able to differentiate between implicitly and explicitly requested URLs/commands, maintain a list of recently-accessed URLs, and display different types of content using the most appropriate of the two applications. We found that by doing this, we were able to display any of the content-types we were interested in accessing. Adobe Acrobat documents are accessed through the Internet Explorer control. Also, within our institution major television channels are available as streaming video, so there was no need to treat this type of media different from a video stream which is dealt with the Media Player control.

## 6. Discussion

Proactive environments currently receive considerable interest, as the research field moves forward from early demonstrations of smart rooms [7] to large initiatives investigating ambient intelligence [8]. One prominent example of proactive environments is the Reactive Room project, which took a thorough HCI perspective on augmented environment concerning itself with issues of predictability and controllability [17,18]. In the Reactive Room project, a room used for video conferencing was made aware of the context of both users and objects in the room for the purpose of relieving the user of the burden of controlling the objects.

Yet pure proactive environments, in which users have no influence over the environment's behavior, are problematic for two reasons. First, pure proactive environments are not likely to facilitate any significant personalization of the user experience. Secondly, many scenarios of proactive environments appear to be unrealistic and even undesirable from a user experience perspective. They tend to assume users will want their environments to act on their behalf and that users will agree with the proactive behaviours they exhibit. However, HCI studies of adaptive interactive systems have consistently emphasized controllability as fundamental usability issue [9].

We address the problem of pure proactive environments by introducing the *Pendle* personalized wearable device. A core aspect of the presented research is to integrate the distinct advantages of personal wearable devices with those that proactive environments offer. The benefits of such integration have previously been discussed by Rhodes et al. who also sketched a variety of application scenarios [10]. Their work highlights the distinct advantages of wearable vs. ubiquitous facilities and their combination (e.g. personalization and localization). Our work takes this forward with the focus on the interactions afforded by a combination of personal interaction devices and proactive environments. There has been further work that investigates interactions between personal devices and

environment-based facilities, however generally with a focus on explicit interaction (e.g. [11] on use of personal devices to control shared displays, and [12] on interactive applications migrating across personal and public devices).

A different emphasis in combination of wearable and environment-based technologies for interactive services is largely explored in many ubiquitous computing projects (e.g. [13]). Here, the focus is generally on environment-based services that integrate wearable components such as Active Badges [14] for identification and localization of users, for example to allow users to summon their remote desktops to nearby displays [15].

Key to our approach is to foresee a wearable device that provides for casual interaction on the basis of an easy-to-use repertoire of hand gestures. In related work, Starner et al have proposed a wearable gesture interface, like ours in the form-factor of a pendant [16]. Their Gesture Pendant is designed for explicit environment control with user-definable gestures performed in front of the pendant. Gesture recognition is based on computer vision, requiring significantly more computational resources than provided in our compact device design. Rekimoto proposed a simple gesture input technique that is based on a wrist-mounted device with acceleration sensor and sensor electrodes [17]. GestureWrist can recognize several variations of gestures. However as it is designed to be always on, i.e. not foreseeing an explicit trigger mechanism, it can yield unintended recognitions. We believe our approach affords significantly more casual interaction ('fingering a device worn around the neck') and lower cognitive load (no hands-ear coordination).

Many projects in the area of proactive environments employ computer vision infrastructure for external observation of users. This raises concerns with respect to intrusion on privacy that we believe need to be carefully weighed. Examples are the EasyLiving project visually tracking users [19] and work of Darrel et al proposing face detection in augmented environments [20]. We address the privacy problem by basing environment adaptation and personalization on information willingly and explicitly provided by users. The use of personal device that are under control of the user at all times give the user control over what kind of information is disclosed at which time.

## 7. Conclusion

In this paper we have proposed a proactive environment controlled by personal wearable devices, The *Pendle* device serves as mediator between user and proactive environment. It provides the user with control and influence over their environment's proactive behaviour on the basis of a simple user interface that lends itself to casual interaction and a smooth integration

of environment-controlled interaction (experienced by the user as implicit interaction, triggered by their presence) and user-controlled interaction (i.e. explicit interaction to directly manipulate the behaviour of the environment).

The result is an unobtrusive adaptation of information displays according to the interests of nearby. At the same time, the *Pendle* device enables users to override proactive behaviors to quickly and easily access content on the internet. As such, the system improves the interaction between the users and the valuable displays services in their vicinity without making it obtrusive or distracting to do so, while at the same time creating a more interesting space to inhabit.

As it stands, the system allows for exploration into several interesting ways in which it can be improved. In particular, it would be interesting to examine different methodologies for locating the most appropriate displays services or more advanced algorithms for finding relevant content to display.

## References

- [1] F. Bennett, T. Richardson, and A. Harter. "Teleporting – making applications mobile." *Proc. Of the IEEE Workshop on Mobile Computing Systems and Applications*, pages 82-84, Santa Cruz, California, Dec. 1994. IEEE Computer Society Press.
- [2] T. Starner, J. Auxier, D. Ashbrook and M. Gandy. "The Gesture Pendant: A Self-illuminating, Wearable, Infrared Computer Vision system for Home Automation Control and Medical Monitoring." *International Symposium on Wearable Computers*, Atlanta, GA: IEEE Computer Society, pp. 87-94 (2000).
- [3] N. Sawhney, S. Wheeler, and C. Schmandt. "Aware Community Portals: Shared Information Appliances for Transitional Spaces." *CHI 2000, April, 2000*.
- [4] R. Want, "The Active Badge Location System," Olivetti Research Laboratory, *ACM Trans. Information Sys.* Vol. 10, no 1, 1992, pp.91-102
- [5] O. Cakmakci, J. Coutaz, K. Van Laerhoven, and H.-W. Gellersen. "Context Awareness in Systems with Limited Resources". In Proc. of the third workshop on Artificial Intelligence in Mobile Systems (AIMS), ECAI 2002, Lyon, France. 2002. pp. 21-29.
- [6] The Smart-Its Project. <http://www.smart-its.org>
- [7] A. Pentland. Smart Rooms, *Sci Amer.*, April 1996.
- [8] European Research Consortium for Informatics and Mathematics. *Ambient Intelligence*, ERCIM News No.47, Oct. 2001. [http://www.ercim.org/publication/Ercim\\_News/enw47/](http://www.ercim.org/publication/Ercim_News/enw47/)
- [9] A. Jameson. Adaptive Interfaces and Agents. In J.A. Jacko, A. Sears (Eds.) *The Human-Computer Interaction Handbook: Fundamentals, Evolving Technologies and Emerging Applications*. Lawrence Erlbaum Assoc, 2002
- [10] B.J. Rhodes, N. Minar and J. Weaver. Wearable Computing Meets Ubiquitous Computing: Reaping the best of both worlds. In *Proc. of 3rd Intl. Symp. on Wearable Computers (ISWC '99)*, San Francisco, CA, Oct. 1999.
- [11] B.A. Myers, H. Stiel, and R. Gargiulo. Collaboration Using Multiple PDAs Connected to a PC. In *Proc. CSCW'98: ACM Conference on Computer-Supported Cooperative Work*, Nov 1998, Seattle, pp. 285-294.
- [12] Greenberg, S., Boyle, M. and LaBerge, J. PDAs and Shared Public Displays: Making Personal Information Public, and Public Information Personal. *Personal Technologies*, Vol.3, No.1, March 1999, pp. 54-64, Springer-Verlag.
- [13] M. Addlesee, R. Curwen, S. Hodges, J. Newman, P. Steggles, A. Ward, and A. Hopper. Implementing a Sentient Computing System. *IEEE Computer*, Vol 34, No 8, pp. 50-56, August 2001.
- [14] A. Harter and A. Hopper, A Distributed Location System for the Active Office. *IEEE Network*, Vol. 8, No. 1, January 1994.
- [15] F. Bennett, T. Richardson and A. Harter. Teleporting - Making Applications Mobile. *Proc. WMCSA '94, Workshop on Mobile Computing Systems and Applications*, Santa Cruz, Dec. 1994.
- [17] J. Rekimoto. GestureWrist and GesturePad: Unobtrusive Wearable Interaction Devices. *Proc. ISWC'01, Fifth Intl Symposium on Wearable Computers*, October 2001, Zurich, Switzerland.
- [16] T. Starner, J. Auxier, D. Ashbrook and M. Gandy. The Gesture Pendant: A Self-illuminating, Wearable, Infrared Computer Vision system for Home Automation Control and Medical Monitoring. *Proc. ISWC 2000, Intl. Symp. on Wearable Computers*, Oct 2000, Atlanta, GA, pp. 87-94 (2000).
- [17] J.R. Cooperstock, K. Tanikoshi, G. Beirne, T. Narine and W. Buxton. Evolution of a Reactive Environment. *Proc. of CHI'95, Conference on Human Factors in Computing Systems*, Denver, May 1995.
- [18] Cooperstock, J., Fels, S., Buxton, W. & Smith, K. Reactive environments: Throwing away your keyboard and mouse, *CACM* 40(9), 1997, pp. 65-73.
- [19] J. Krumm, S. Harris, B. Meyers, B. Brumitt, M. Hale, S. Shafer. Multi-Camera Multi-Person Tracking for EasyLiving. *IEEE Workshop on Visual Surveillance*, July 2000.
- [20] T. Darrell, K. Tollmar, F. Bentley, N. Checka, L.-P. Morency, A. Rahimi, and A. Oh. Face-responsive interfaces: from direct manipulation to perceptive presence, *Proc. Ubicomp 2002, Intl. Conf. on Ubiquitous Computing*, Gothenburg, Oct 2002.
- [21] A.Y. Benbasat and J. A. Paradiso. An Inertial Measurement Framework for Gesture Recognition and Applications. *Proc. GW 2001, International Gesture Workshop*, London, UK, 2001.