wear-UCAM: A Toolkit for Mobile User Interactions in Smart Environments*

Dongpyo Hong, Youngjung Suh, Ahyoung Choi, Umar Rashid, and Woontack Woo

GIST U-VR Lab.
Gwangju 500-712, Korea
{dhong, ysuh, achoi, urashid, wwoo}@gist.ac.kr

Abstract. In this paper, we propose a toolkit, wear-UCAM, which can support mobile user interactions in smart environments through utilizing user's context. With the rapid developments of ubiquitous computing and its relevant technologies, the interest in context-aware applications for mobile/wearable computing also becomes popular in both academic and industrial fields. In such smart environments, furthermore, it is crucial for a user to manage personal information (health, preferences, activities, etc) for the personalized services without his or her explicit inputs. Regarding reflection of user's context to context-aware applications, however, there are only a few research activities on such frameworks or toolkits for mobile/wearable computers. In the proposed wear-UCAM, therefore, we focus on a software framework for context-aware applications by taking account of how to acquire contextual information relevant to a user from sensors, how to integrate and manage it, and how to control its disclosure in smart environments.

1 Introduction

In recent years, with the development of ubiquitous computing enabled technologies and their rapid deployment, there has been an increased interest in context-aware applications on mobile/wearable computing [1,2]. In ubiquitous computing, it is generally required to install various sensors in environments in order to serve users [2]. Consequently, it is not easy for a user to manage personal information from the environmental monitoring [3].

In mobile/wearable computing, however, personal information can be retrieved from various wearable/mobile sensors, analyzed and used in services through explicit controls of a user rather than that of environments [3,4]. In this regard, a user interface for mobile/wearable computers should support users' personal information to be controlled by users themselves as well as manage their preferences to have quality of services. This is the reason that mobile/wearable computing has become a focus of attention in the era of ubiquitous/pervasive computing. In addition, mobile/wearable computing complements the inherent limitation of ubiquitous computing such as cost for infrastructure construction, privacy from environmental monitoring and so on.

* This was supported by Seondo project of MIC, Korea.

E. Sha et al. (Eds.): EUC 2006, LNCS 4096, pp. 1047 – 1057, 2006.

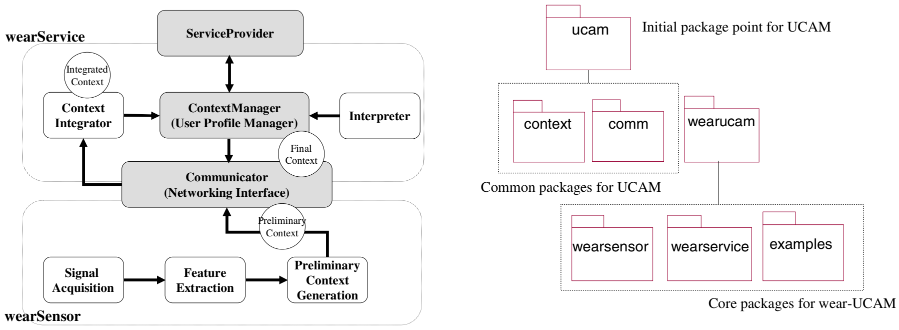
Nonetheless, there have been only a few research activities on context-aware application development toolkits for a mobile/wearable computing compared to those for ubiquitous computing [5-7]. Furthermore, most of the existing context-aware applications exploit only a few contextual cues such as location, identity and time [1]. On the other hand, it is burdensome for developers to adopt toolkits for ubiquitous computing into mobile/wearable computing because toolkits for ubiquitous computing include complicated mechanisms for sensors to detect where we are, what we want etc. Therefore, it is necessary of a toolkit to support acquisition of contextual information from wearable sensors, management of user's profile for feedback, and disclosure control of user's personal information for a proper service.

In this regard, we propose wear-UCAM, which is a toolkit for the rapid application development that supports mobile user interactions for personalized services in smart computing environments. The proposed wear-UCAM provides developers with methods for acquiring personal information from wearable sensors, methods for manipulating and analyzing the acquired information, and methods for processing flows of contextual information in applications for personalized services. In particular, the proposed wear-UCAM has the following three characteristics: 1) Acquisition and analysis of the user's physiological signals, 2) User profile management from the extracted and analyzed information, and 3) Control mechanisms for disclosure of personal information. In addition, any application based on wear-UCAM is split into *wearSensor* and *wearService* in order to support independence of sensors and services. This separation of sensors and services is also a key feature of ubiquitous computing [7]. Thus, there are many components in wear-UCAM to support extracting and analyzing a user's context from various sensors, which we explain in Section 2. To show the effectiveness of wear-UCAM, we utilize a physiological sensor and a indoor location tracking sensor as wearable sensors.

This paper is organized as follows. In Section 2, we sketch the overall architecture and design issues of wear-UCAM and its components briefly. In Section 3, 4 and 5, we explain the implemented components for context integration, context management and information disclosure control, respectively. The experimental setup and result with detailed analysis of wear-UCAM are described in Section 6. Finally, we discuss the proposed wear-UCAM as a toolkit for mobile user interactions and future works in Section 7.

2 wear-UCAM

wear-UCAM requires application developers to split an application into *wearSensor* and *wearService* and to connect *wearSensor* with *wearService* via the communicator (a networking interface) as shown in Figure 1 (a). Through this separation into sensor and service in the application, we can utilize not only wearable sensors on a user but also other sensors which probably have more computational power in environments. This mechanism can be crucial especially for resource restricted computing platforms like wearable and mobile computing. However, personal context, being highly relevant to personal privacy, is protected from other wearable computers and environmental monitoring by the explicit controls of the user. For context-aware applications



(a) (b)

Fig. 1. (a) Software Package Structure (b) Conceptual wear-UCAM Architecture

in future computing environments, application developers should be able to manipulate user's context easily. However, application developers need not know low-level details of context-aware mechanisms for implementing context-aware applications.

As illustrated in Figure 1 (b), wear-UCAM includes the following common packages: **context** package for context model, and **comm** package for a networking interface. Specifically, sensors and services are able to communicate with each other using a common context data format due to **context** and **comm** packages. The key packages of wear-UCAM are **wearsensor**, **wearservice**, and **examples**. Table 1 shows the core components for user's context processing with respect to sensor and service.

Table 1. Functionalities of components in wear-UCAM

wear-UCAM	Component	Functions
wearSensor	*Sensor**	Basic functions of wearSensor (signal processing, feature extraction, preliminary context)
Networking	Communicator	Dynamic connection among sensors and services
wearService	*Service**	Basic functions of wearService (service classification, register ServiceProvider)
	ContextIntegrator	Generates integrated context by analyzing preliminary context
	ContextManager	Generates final context by manipulating user's preference and the integrated context
	Interpreter	User's conditional context provision
	*ServiceProvider**	Basic interfaces for developers who develop context-aware application

** Abstract Class*

The package of **wearucam** includes various classes to support sensors and services which are applicable in mobile/wearable computing environments. For instance, *wearSensor* extracts signals from sensors and constructs a preliminary context by analyzing retrieved signals. Therefore, developers are able to implement their applications in mobile or wearable computing without taking care of the details of context-aware mechanisms for handling context-awareness. Furthermore, we separate different types of sensors in order to support various wearable sensors on a user.

In short, the architecture of wear-UCAM enables mobile/wearable computing application developers to exploit context-aware mechanisms as well as supports communications among heterogeneous applications through the context and networking interface in heterogeneous computing environments. As shown in Figure 1 (a), wear-UCAM requires sequences of context processing in order to exploit the user context in applications in which context information is acquired from various sensors and utilized in various services. Although we utilize the user-centric context model in this paper, the contextual elements are modified to fit into wear-UCAM [8]. For example, contextual information obtained from physiological signals is added.

3 Context Acquisition and Integration on Physiological Signals

In this section, we introduce the way how we can obtain the personal information from wearable sensors and integrate the contexts through the proposed wear-UCAM. *wearSensor* transfers acquired signals into preliminary contexts which include some elements of 5W1H. The first step of *wearService* is the integration of preliminary contexts. Thus, the role of *ContextIntegrator* in wear-UCAM is to fill out each contextual element (5W1H) by using the preliminary context obtained from various sensors. Namely, it creates the integrated context from the preliminary contexts which have several blanks in contextual elements because a sensor cannot fill out all contextual elements. In this Section, we focus on 'how' and 'why' contextual elements to integrate the preliminary context from physiological sensors, which delivers body conditions of a user such as heart rate, temperature and skin humidity. The 'why' contextual element gives information about higher level context of user's internal states (intention, attention, emotion, etc) by inferring, interpreting and analyzing other contextual elements (who, when, how and what). The overall procedure of acquisition and analysis for physiological signals in wear-UCAM is illustrated in Figure 2.

The *ContextIntegrator* in wear-UCAM basically has two kinds of fusion methodologies; temporal fusion and spatial fusion. These fusion functions are applied in each 5W1H field fusion. The temporal fusion reduces the burden of real-time data processing from various sensors to make one integrated context by using a voting algorithm

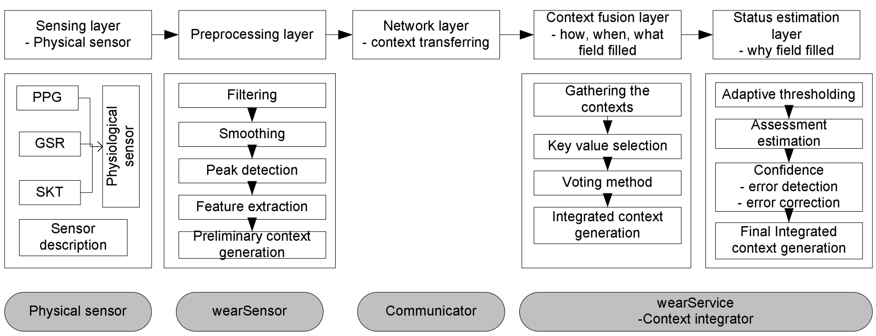
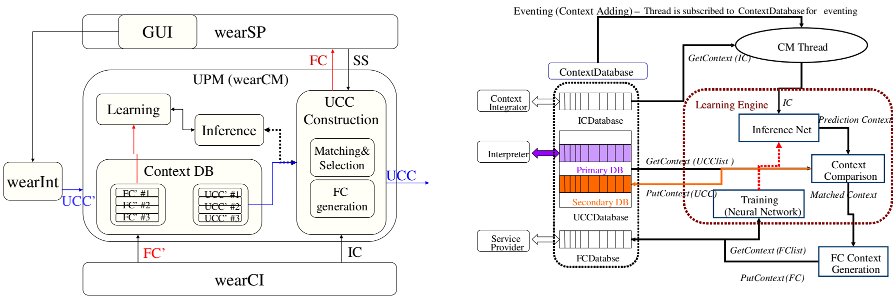


Fig. 2. The steps of acquisition and analysis of physiological signal in wear-UCAM

which selects the maximum number of key-value. For the voting algorithm, we use representative criteria to integrate the preliminary contexts under multiple hierarchies of context. For the spatial fusion methodology, the *ContextIntegrator* module supports the personalized analysis of physiological signals by adaptive thresholds. In 'why' fusion module, we use spatial fusion for user adaptive analysis of physiological signals, such as tension, stress, emotion, etc. In this paper, we analyze the tension level with GSR (Galvanic Skin Response) signal. Measurements of GSR reflect the amount of humidity in skin surface and are used to analyze the states of tension with a user-adaptive threshold [9, 10].

4 Context Management for User Profile

In general, context management is a key factor to utilize the processed contextual information in context-aware applications. The *UserProfileManager* (UPM) generates the final context by comparing the integrated context and the conditional context. Firstly, UPM receives the integrated contexts from *ContextIntegrator*. Then, it compares the integrated context with the conditional context, which consists of the user conditional context and the service conditional context. At last, it generates the final context and sends it to *ServiceProvider*. The role of UPM is to learn the preferences of a user to a certain service so that it can provide a personalized service. Specifically, it generates the user conditional contexts and disseminates them into the given service environment while automatically updating them.



(a) (b)

Fig. 3. (a) User Profile Manager in wearService (b) User Profile Handling Module

Figure 3 (a) shows the flows of contextual information in UPM, where IC is the integrated context, FC is the final context, SS is service status, and UCC is the user conditional context. In addition, FC' and UCC' are the results processed from the given FC and UCC. Specifically, user profiles are described in 5W1H of the user conditional context (UCC), 'who' being as a focal factor. Each element of 3W1H ('where', 'when', 'how', 'why') in UCC represents where a user is, when he/she enjoys a specific service, which gesture he/she makes, and what his/her stress level is. In UCC, 'what' element denotes what kind of services he/she likes to enjoy. The final context (FC) contains the results of service execution which a user desires. Thus, we can acquire the service-specific user preferences which are dynamically changed by collecting the final context from services. After learning and matching contexts, we can update the user conditional context (UCC), which represents a user's indirect intention to enjoy a specific service, so that users can be provided with personalized services.

Figure 3 (b) illustrates the process of constructing UCC and disseminating it into the environment through a network. The procedure of UCC construction and dissemination is as follows. First of all, we construct a primary database in UCC database from static user-related information provided through a GUI. Then, FCs are aggregated at a periodic interval and stored in the FC database. At a regular pace, the UPM analyzes and learns the dynamic user-related information from FCs. At last, through context matching process, the UPM generates a new UCC, updates a secondary database in the UCC database with it, and sends it to other services.

Although users new in smart environments can provide their personal information and service-specific preferences through a GUI offered by *Interpreter* in wear-UCAM, it is necessary to automatically update a user's service-specific preferences when a user's command is not explicitly provided. And what if users want to enjoy a service for which they have not initially set their preferences through the GUI? For these situations, it is also necessary to learn a user's service usage history to infer service-specific preferences, so that results of learning can be reflected into the dynamic update of user profile.

5 Personal Information Disclosure Management

Disclosure of personal information is inevitable to personalize services and applications in context-aware computing environments. Context-aware application developers need to provide the user with flexible ways to control when, to whom and at what level of detail he/she can disclosure his/her personal information.

Most of the research activities in this realm have been primarily focused on protecting personal information of users in context-aware systems deployed in institutes and organizations [11]. However, privacy concerns that arise in the office environments of organization and institutes do not necessarily apply to home environments. As pointed out by Hindus (1999), "homes are fundamentally different from workplaces", "customers are not knowledge workers", and "families are not organizations" [12]. Family members living in the same home are more closely knit than colleagues in a workplace, and privacy from other family members is not as big an issue as it is for an enterprise. However, while family members in a smart home may be quite frank to share the personal information with each other, they may desire to keep certain elements of the personal information obscure from context-aware services in a smart home. This signifies the need for mechanisms that provide the family members with the flexibility to adjust the granularity of context information disclosed to service providers.

In this regard, we illustrate this phenomenon with the model scenario of a context-aware movie service [13]. We draw inspiration for this service from the context-aware Movie Player system and physiological signal sensor. The movie service provides the user with appropriate movies based upon the detail level of context information he/she discloses to the service. The user can choose and assign priorities to his/her preferences for movies in "what" element of 5W1H in user conditional context (UCC), via a Graphical User Interface (GUI) offered by *Interpreter*. For example, if the user does not want to disclose some of his/her preferences to the service, he/she can set the value to "0" for those elements as shown in Table 2.

Table 2. User's preference on the context-aware movie service

Preference	Priority
SF	10
Horror	0
Drama	7
Comedy	5
Animation	3

There is a tradeoff between user's privacy and utility of service. The more a user discloses personal information, the more customizable and beneficial the service becomes. The more specific the user is in revealing his/her personal context information to context-aware service, the more relevant movies he/she is likely to receive from the service. In this way, wear-UCAM provides the user with the option to fine-tune disclosure of his/her personal information and avail the service in relation to the disclosed information.

6 Experimental Result

Figure 4 illustrates the interaction scene between wear-UCAM and services in ubiHome test-bed, where we use a PDA-based wear-UCAM simulator. We use 2

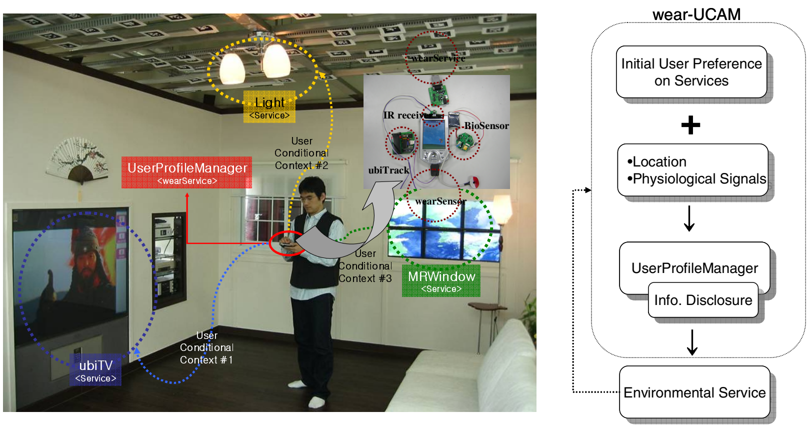


Fig. 4. Experimental setup and interactions between wear-UCAM and services

services, *MRWindow* and *ubiTV*, which can provide multimedia services based on user's preferences.

As shown in Figure 4, a user disseminates his or her preferences (initial user preference, i.e., UCC) in order to use services in ubiHome. However, the user can control the dissemination of his or her contextual information, i.e., whether to send it to services or not. Based on his or her preferences, a corresponding service is triggered. While the user is using the service, location and physiological signals are acquired from sensors. In this experiment, we exploit the processed physiological sensor which tells whether a user is in tension or not, and location sensor, called ubiTrack [14], which tracks a user's location in ubiHome. Any change of these signals and service is contextualized and reflected to user's preference. Through this procedure, the user can interact more naturally with services in smart environments.

In order to acquire contextual information relevant to a user, we employ two kinds of sensors; a wrist type physiological sensor and an IR (infrared)-based location sensor. In this experiment, we implement *BioSensor* and *ubiTrack* sensor module from *Sensor* class because this sensor class in wear-UCAM provides the necessary procedures for generating preliminary context. Table 3 shows generated preliminary context.

Table 3. The generated preliminary context from sensors

	Component		**Contextual Information**	**Note**
sensor	BioSensor	Who	Name=Dongpyo, Priority=4	Preliminary context
		What	SensorID=1, SensorType=BioSensor	
		How	GSR.mean=14.37 GSR.var=3.0 PPG.HR_mean=63.0 PPG.HR_var=3.4 PPG.HR_LF_power=70.0, PPG.HR_HF_power=62.0 SKT=36.5	
	ubiTrack	Who	Name=Dongpyo, Priority=4	
		Where	IndoorLocation.x=140 IndoorLocation.y=160 SymbolicLocation=TV Direction=0.0	
		What	SensorID=2, SensorType=LocationSensor	

*GSR(Galvanic Skin Response), PPG(photoplethysmogram), SKT(skin temperature)

The *ContextIntegrator* integrates a set of the generated preliminary contexts from various sensors through the temporal and spatial fusion methodologies. In the experiment on the context integration, we find that GSR can be used to analyze the states of tension. Although the preliminary context from *BioSensor* has 3 kinds of signals, we only process the GSR signals in the *ContextIntegrator* because PPG signal is not stable in noise and motion, and SKT signal is almost invariant. As a result, the integrated contextual information is used in updating user's preference while the user is using a service as shown in Table 4.

Table 4. The integrated context from ContextIntegrator

Component			Contextual Information	Note
service	ContextIntegrator	Who	Name=Dongpyo, Priority=4	Integrated context
		When	130527(hh:mm:ss)	
		Where	IndoorLocation.x=140 IndoorLocation.y=160 SymbolicLocation=TV Direction=0.0	
		What	-	
		How	GSR.mean=14.37 GSR.var=3.0	
		Why	Tension=0	

Table 5. The final context from ContextManager

Component			Contextual Information	Note
service	ContextManager	Who	Name=Dongpyo, Priority=4	Final context
		When	130640(hh:mm:ss)	
		Where	IndoorLocation.x=140 IndoorLocation.y=160 SymbolicLocation=TV Direction=0.0	
		What	ServiceName=ubiTV Parameter={"SF", 10} Function=Play	
		How	GSR.mean=14.37 GSR.var=3.0	
		Why	Tension=0	

With the integrated context and the initial user preference (UCC), UPM finds a matched service from the given service environment. Since UPM makes UCCs which reflect the user's preferences that suit him best, it can play a key role in providing personalized services in the environment. After a corresponding service is executed, UPM keeps updating user's preferences based on the integrated context from wearable sensors and the feedbacks from the service.

The final context is generated as shown in Table 5. When the final context is disseminated to the service, an altering message is displayed to a user which asks the user whether he or she wants to the service or not. In this experiment, the user discloses his or her preference on SF movie to *ubiTV* service because he/she sets the value to "10". Thus, the user does not have to take any action to have the service.

7 Discussion and Future Work

In this paper, we proposed the wear-UCAM as a rapid context-aware application development toolkit to support mobile user interactions. Through the wear-UCAM, a user is able to use contextual information obtained from physiological sensor and location sensor for his or her preference. In addition, the user can update his or her preferences on certain services with the contextual information and feedbacks from other services. Finally, the user can control whether to fine-tune disclosure of his or her personal context information or not. The proposed wear-UCAM can provide appropriated services to the users based on their preferences by retrieving personal information from sensors, processing it, and analyzing it. However, wear-UCAM is not yet fully experimented in an actual user study due to unstable physiological signals from user's movements. Moreover, the modeling of contextual information should be considered closely. Additionally, it is required to conduct experiments on the enhancement of sensing physiological signals.

References

1. Mari Korkea-aho, "Context-aware Application Survey," available in http://users.tkk.fi/~mkorkeaa/doc/context-aware.html
2. Guanling Chen, and David Kotz, "A Survey of Context-Aware Mobile Computing Research," Technical Report TR2000-381, November, 2000.
3. Bradley J. Rhodes, Nelson Minar and Josh Weaver, "Wearable Computing Meets Ubiquitous Computing: Reaping the best of both worlds," Proc. of The 3rd International Symposium on Wearable Computers (ISWC '99), San Francisco, CA, October 18-19 1999, pp. 141-149.
4. Jennica Falk,Staffan Bjork, "Privacy and information integrity in wearable computing and ubiquitous computing," Conference on Human Factors in Computing Systems archive CHI '00, pp.177-178, 2000.
5. Brown, P.J., Bovey, J.D., Chen X., "Context-Aware Applications: from the Laboratory to the Marketplace," IEEE Personal Communications, vol.4, no.5, pp. 58-64, 1997.
6. Hull, R., Neaves P. and Bedford-Roberts J., "Towards Situated Computing," 1st International Symposium on Wearable Computers, Cambridge, Massachusetts, October 13-14, 1997, pp. 146-153.
7. Anind K. Dey and Gregory D. Abowd, "The Context Toolkit: Aiding the Development of Context-Aware Applications," In the Workshop on Software Engineering for Wearable and Pervasive Computing, Limerick, Ireland, June 6, 2000.
8. S.Jang, W.Woo, "Unified Context Representing User-Centric Context: Who, Where, When, What, How and Why," ubiComp Workshop (ubiPCMM), pp.26-34, 2005.
9. B. Elie and P. Guiheneuc, "Sympathetic skin response: normal results. in different experimental conditions," Journal of Electroencephalography and Clinical Neurophysiology, vol. 76, pp. 258–267, 1990.
10. A.Barreto and J. Zhai, "Physiological Instrumentation for Real-time Monitoring of Affective State of Computer Users," WSEAS Transactions on Circuits and Systems, vol. 3, pp. 496-501, 2003.

11. Jason I. Hong, James A. Landay, "An Architecture for Privacy Sensitive Ubiquitous Computing," Second International Conference on Mobile Systems, Applications and Systems, pp. 177-189, 2004.
12. D. Hindus, "The Importance of Homes in Technology Research," 2nd Int'l Workshop Cooperative Buildings (CoBuild 99), LNCS, vol. 1670, Springer-Verlag, Berlin, pp.199-207, 1999.
13. Y.Oh, W.Woo, "A unified Application Service Model for ubiHome by Exploiting Intelligent Context-Awareness," Ubiquitous Computing Systems (LNCS), vol.3598, pp.192-202, 2005.
14. W.Jung, W.Woo, "Orientation tracking exploiting ubiTrack," ubiComp05, pp. 47-50, 2005.