ContextAlert: context-aware alert mode for a mobile phone

Santi Phithakkitnukoon
Massachusetts Institute of Technology, Cambridge, Massachusetts, USA, and
Ram Dantu
University of North Texas, Denton, Texas, USA

Received 12 November 2010
Accepted 21 July 2010

Abstract

Purpose – Mobile computing research has been focused on developing technologies for handheld devices such as mobile phones, notebook computers, and mobile IP. Today, emphasis is increasing on context-aware computing, which aims to build the intelligence into mobile devices to sense and respond to the user's context. The purpose of this paper is to present a context-aware mobile computing model (*ContextAlert*) that senses the user's context and intelligently configures the mobile phone alert mode accordingly.

Design/methodology/approach – The paper proposes a three-step approach in designing the model based on the embedded sensor data (accelerometer, GPS antenna, and microphone) of a G1 Adriod phone. As adaptivity is essential for context-aware computing, within this model a new learning mechanism is presented to maintain a constant adaptivity rate for new learning while keeping the catastrophic forgetting problem minimal.

Findings – The model has been evaluated in many aspects using data collected from human subjects. The experiment results show that the proposed model performs well and yields a promising result.

Originality/value – This paper is distinguished from other previous papers by: first, using multiple sensors embeded in the mobile phone, which is more realistic for detecting the user's context than having various sensors attached to different parts of user's body; second, by being a novel model that uses sensed contextual information to provide a service that better synchronizes the user's daily life with a context-aware alert mode. With this service, the user can avoid the problems such as forgetting to switch to vibrate mode while in a meeting or a movie theater, and taking the risk of picking up a phone call while driving, and third, being an adaptive learning algorithm that maintains a constant adaptivity rate for new learning while keeping the catastrophic forgetting problem minimal.

Keywords Mobile communication systems, Adaptive system theory

Paper type Research paper

1. Introduction

Having a handheld device recognized its user's context fits to the scope of context awareness, which is one of the hottest current research areas. Context-awareness aims to enhance our quality of life with intelligent computing devices sensing and reacting to the environment and presence of users. Existing handhled devices such as mobile phones and personal digital assistants (PDAs) have already taken steps towards this computing paradigm.

With the embedded sensors in today's mobile phones such as accelerometer, GPS, and audio sensor, the user's context can be sensed and estimated to some extent using machine learning techniques. In this article, we design and evaluate a context-aware mobile computing model, known as ContextAlert, that intelligently configures the mobile phone alert mode according to user's situational context. For example, the phone can be automatically set to vibrate mode while the user is in a meeting, automatically configured to handsfree mode while the user is driving, etc.

This work is supported by the National Science Foundation under grants CNS-0627754, CNS-0619871, and CNS-0551694.

International Journal of Pervasive Computing and Communications
Vol. 6 No. 3, 2010
pp. 311-332

1742-7371
DOI 10.1108/17427371011084266

Several previous works have been done in context awareness such as service discovery, online/mobile social modeling for providing better services, activity recognition, personal/object positioning, and person identification. Our work is closely related to activity recognition for which previous works have used either a single or multiple sensors attached to different parts of user's body.

We distinguish this work from other previous works by the following contributions:

- We use multiple sensors embeded in the mobile phone, which is more realistic for detecting the user's context than having various sensors attached to different parts of user's body.
- We propose a model that uses sensed contextual information to provide a service that better synchronizes the user's daily life with a context-aware alert mode control. With this service, the user can avoid the problems such as forgetting to switch to vibrate mode while in a meeting or a movie theater, and taking the risk of picking up a phone call while driving.
- As adaptivity is essential for context-aware computing, within our model we propose a learning mechanism that maintains a constant adaptivity rate for new learning while keeping the catastrophic forgetting problem minimal.

The rest of the article is organized as follows: Section 2 briefly reviews the literatures in context awareness that is related to our work. Section 3 presents the system overview of ContextAlert. Section 3 describes our proposed framework for ContextAlert. Our approach in designing ContextAlert is evaluated with several experiments and the results are shown in Section 5. We point out some limitations of our work in Section 6. Section 7 concludes this article with a summary and an outlook on future work.

2. Related work

Context-aware computing research is scoped by ubiquitous computing, a term that was coined by Weiser (1993, 1995). It has also been referred to as pervasive computing and ambient intelligence, which is a computing paradigm that makes multiple computing devices available throughout the physical environment and effectively invisible to the user. Several researchers have attempted to define "context" (Schmidt *et al.* 1999; Chen and Kotz, 2000; Dey, 2001; Hofer *et al.*, 2003; Prekop and Burnett, 2003) since Schilit *et al.* (1994) and Schilit and Theimer (1994) first introduced it in 1994. Han *et al.* (2008) divided context into physical, internal, and social context.

Several works in physical context have focused on service discovery in ubiquitous computing environments based on the user's context (e.g. Coen, 1998; Czerwinski *et al.*, 1999; Friday *et al.*, 2001; Chen *et al.*, 2001; Zhu *et al.*, 2003; Chetan *et al.*, 2005; Toninelli *et al.*, 2008; Park *et al.*, 2009). Meanwhile, research in the social context area has been reported in both online and mobile social networks by modeling social dynamics and using social context information to provide better service for users (e.g. Paulos and Goodman, 2004, Davis and Karahalios, 2005; Eagle and Pentland, 2005; Oulasvirta *et al.*, 2005; Buriano, 2006; Eagle and Pentland, 2006; Kostakos *et al.*, 2006; Jian *et al.*, 2008). Our work is in the area of internal context, which was defined as an abstract thing inside people such as feeling, thought, task, action, interest, and so on (Han *et al.*, 2008). Recent works include context extraction (e.g. Siewiorek *et al.*, 2003; Krause *et al.*, 2003; Adams *et al.*, 2006, Husna *et al.*, 2008), activity recognition (e.g. Bao and Intille, 2004; Munetoshi *et al.*, 2004, Koichi, 2004), personal/object positioning (e.g. Kourogi and

Kurata, 2003; Liu, 2006), and person identification (e.g. Bernardin and Stiefelhagen, 2007; Suutala and Röning, 2008; Grosse *et al.*, 2008).

Laerhoven and Cakmakci (2000) proposed a context-awareness system that learned the user's activities from two-axis accelerometers, passive infrared sensors, carbon monoxide sensor, microphones, pressure sensors, temperature sensors, touch sensors, and light sensors using Kohonen self-organizing maps and Markov models.

Lester *et al.* (2004) presented a method using accelerometers to determine if two devices were carried by the same person based on a coherence function (a frequency-domain linear correlation).

Lukowicz *et al.* (2004) presented a technique to automatically track the progress of maintenance or assembly tasks using body-worn three-axis accelerometers, microphones, and computers based on frequency-matching sound classification technique that combined the intensity analysis of signals from microphones at different parts of body and correlation analysis of surrounding sounds and user activity.

Bao and Intille (2004) developed a system to detect activities such as walking, sitting, standing, running, and so on using body-worn two-axis accelerometers based on mean energy, frequency-domain entropy, and correlation and decision tree classifiers.

Krause (2006) presented a multi-sensor wearable system that learned context-aware personal preferences by identifying individual user states and observing how the user interacted with the system in these states. This work was based on the previous model proposed by Siewiorek *et al.* (2003). Sensor data were preprocessed using different methods such as fast Fourier transform and principal component analysis (PCA), and then clustered using Kohonen self-organizing maps (Kohonen, 2001) and Markov models.

Jin *et al.* (2008) proposed a context-awareness system that distinguished user motion states and recognized emergency situations using a two-axis accelerometer, heat flux sensor, galvanic skin response sensor, skin temperature sensor, and near-body ambient temperature sensor based on a fuzzy inference model.

These recent works in internal context area adopt the wearable computer approach, which requires several sensors to be attached to specific parts of the user's body to sense the most accurate context data. These approaches are thus not realistic. Nevertheless, the preprocessing techniques, machine learning approaches, and probabilistic models used in these works are useful.

There are some recent studies reported in recognizing activities using an accelerometer attached to the mobile device. Iso and Yamazaki (2006) proposed a gait analyzer based on a three-axis accelerometer mounted on a mobile phone using a wavelet packet decomposition for preprocessing data and a self-organizing map with Bayesian theory for classification. Yi *et al.* (2005) conducted a study to determine what contextual information could be obtained from a three-axis accelerometer attached to a PDA by having subjects perform some activities while carrying PDAs.

3. Context-aware alert mode

The user's context is very complex to be comprehended entirely from sensor data. We nevertheless believe that it can be estimated and interpreted to some extent. With the embedded sensors in the mobile phones such as the accelerometer, GPS, and audio sensor, the user's movement, mobility, and ambient noise level can be sensed respectively.

We propose here a context-aware alert mode control (ContextAlert) that configures the call alert to the most suitable mode corresponding to user's context. With today's mobile phones, the user has three call alert options: ringer, vibrate, and handsfree. These options are suitable for different situations. Handsfree mode (bluetooth headset) is most suitable when the user is driving a vehicle. In fact, many states in the USA have prohibited drivers from talking on mobile phones while driving (Governors Highway Safety Association, 2009). Vibrate mode is most suitable when the user is in a meeting, theater, library, etc. Ringer is used mostly in general and it is preferred in situations in which the user can be interrupted by a ringer such as while shopping, having lunch, or walking in a park.

Our notion of context is therefore defined as a user's physical situation, which is a cluster of feature attributes obtained from the sensor data at an interval of time. Accordingly, the user's context can be divided into three states:

(1) *Uninterruptible by ringer (UR)*: In this state, user does not want to be interrupted by a ringer. Normally, this situation occurs while user is in a considerably quiet place with low movement and mobility e.g. in a meeting, in a theater, at a library, etc.

(2) *Interruptible by ringer – vehicular mode (IR-V)*: The user can be interrupted by a ringer in this state but it is unable to use hands to operate the phone. This is usually a driving situation, in which the environmental noise level is typically higher than in the UR state. The movement is normally low but the mobility is clearly high.

(3) *Interruptible by ringer – non-vehicular mode (IR-N)*: This state corresponds to situations in which user is interruptible by a ringer and not driving a vehicle. Situations include shopping in a mall, walking with friends in a hall way, having lunch, and jogging in a park. These situations are typically at high ambient noise level, high movement, and low mobility.

With these user context states, ContextAlert sets the alert option to the most suitable mode according to Definition 1, learns the user's preference from the feedback, and adjusts the inference engine accordingly (shown in Figure 1).

Definition 1. If the user's context is UR, then the most suitable alert option is vibrate mode. If the user's context is IR-V, then the most suitable alert option is handsfree mode. If the user's context is IR-N, then the most suitable alert option is ringer mode.

4. Framework

This section describes the ContextAlert framework, which includes our approach in designing the models and details on sensor data acquisition, data preprocessing, the context classifier, the inference engine, and the adaptive learning mechanism.

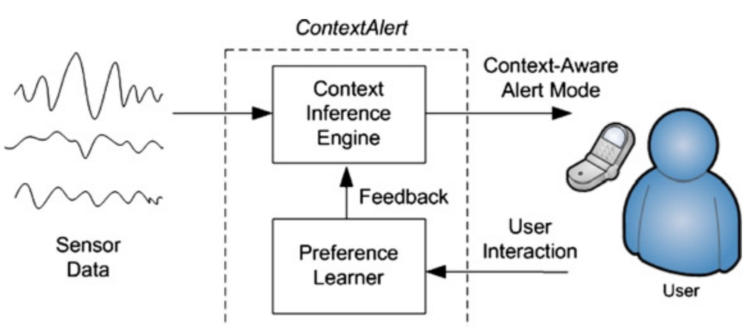


Figure 1.
System overview of ContextAlert

4.1 A three-step approach

There are three steps in our design. The first step is "training," which is an offline supervised learning process to construct an initial context map by classifying the labeled training samples into three different context states (UR, IR-V, and IR-N). In this step, sensor data are preprocessed to obtain useful features (details are described in section 4.3), then fed into classifier to generate an initial context map using PCA (details are described in section 4.4). The second step is "inferring," which is an online unsupervised learning process to analyze input sensor data and infer the user's context state based on k-nearest neighbor algorithm (k-NN) and finite state machine model (details are described in section 4.5). The third step is "user preference learning," which is an online supervised learning process to learn the user's preferences based on the feedback (details are described in section 4.6). This three-step approach is illustrated in Figure 2. On top of the three-step approach, a learning algorithm is applied for the system to remain adaptive for new learning while the catastrophic forgetting problem is maintained at a minimum (details are described in section 4.7).

4.2 Data acquisition

In this work, the data were collected using the embedded sensors of the G1 phones (T-Mobile, 2009) with Google Android 1.1 operating system, 32 bit Qualcomm MSM7201A (528 MHz CPU clock), 256 MiB ROM, and 192 MiB RAM. These sensors included a three-axis accelerometer, a GPS navigation system with Qualcomm MSM7201A gpsOne using NIMEA 0183 protocol, and an audio sensor with 16 bit nominal quantization and a sampling frequency of 44,100 Hz.

To acquire data from these sensors, we created an application for G1 phone using Android 1.1 SDK (2009). The phone was carried inside the front pants pocket while the data were collected.

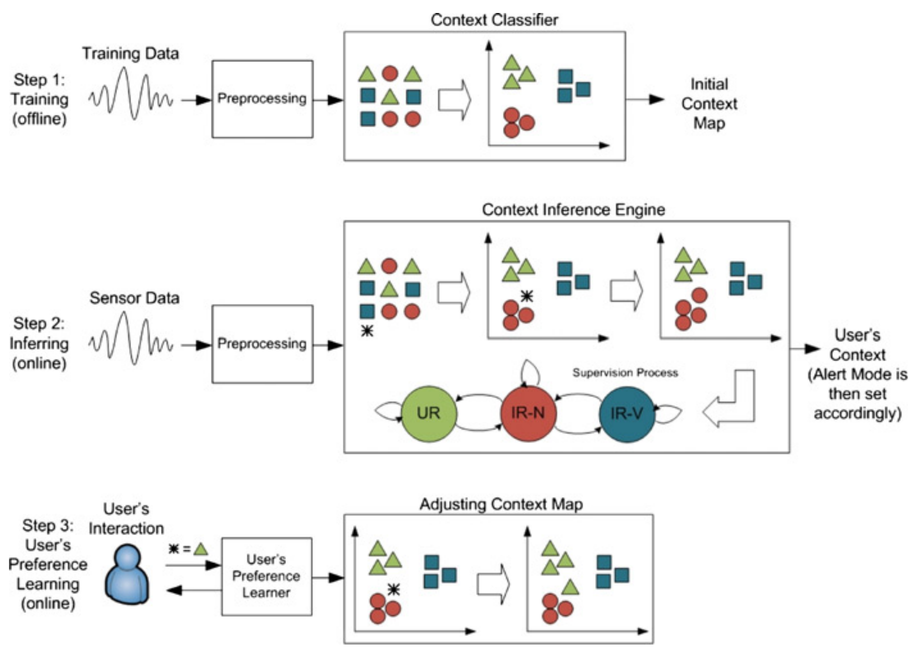


Figure 2.
Our three-step approach constructs an initial context map using supervised learning in the training step, then uses the initial map to estimate user's context in the inferring step, and learns user's preference from the feedback

4.3 Preprocessing methods

Preprocessing is needed to extract useful features from the raw sensor data. To estimate the user's movement, we compute the magnitude of the force vector by combining the measurements from all three axes using Equation (1) to derive a net acceleration (*a*) independent of orientation. Note that if there is no movement, the magnitude is approximately at 1 G due to Earth's gravity (9.8 m/s^2).

$$a = \sqrt{c_x^2 + c_y^2 + c_z^2}, \tag{1}$$

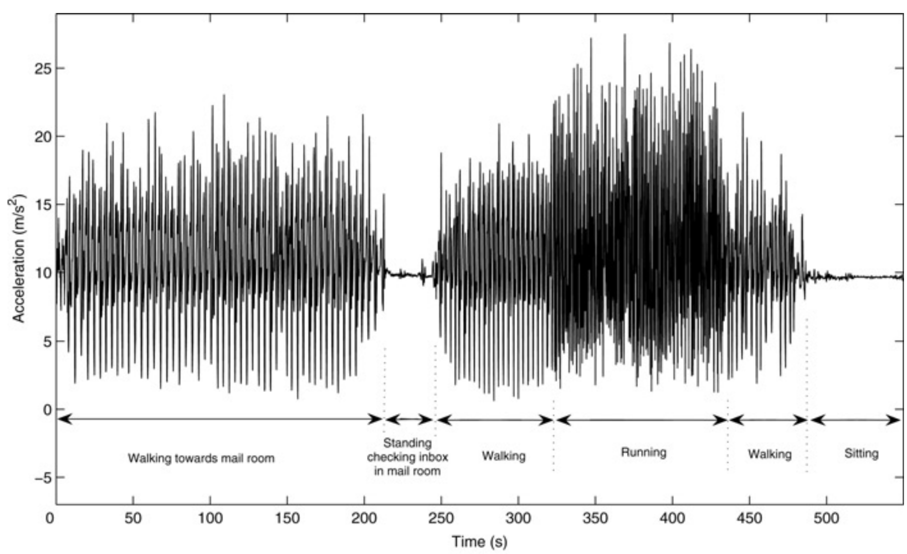
where c_x, c_y, and c_z are measurements from *x*, *y*, and *z*-axis of accelerometer, respectively.

Figure 3 shows the net acceleration's magnitude of a subject walking, standing, running, and sitting. The subject carried the phone in his pant pocket for this experiment and all other experiments in this article.

For estimating the user's mobility, we use GPS data to compute the traveling speed by calculating a distance (minimum distance or length of a displacement) between user's current position and the previous one based on the latitude and longitude information. Then the user's traveling speed can be obtained simply by dividing the distance by a time difference between two positions as given by Equation (2):

$$s = \frac{\sqrt{(\phi_1 - \phi_2)^2 + (\lambda_1 - \lambda_2)^2} \times 111}{\Delta T}, \tag{2}$$

where ϕ_i and λ_i denote a latitude and a longitude value at location *i*, respectively. The constant 111 is the approximated converting ratio of distance from one geographic degree to kilometer unit. Time difference between two locations is represented by ΔT in hour units.



Notes: Data show the subject walking to a mail room, checking his mail box, walking/running/walking back to an office, and sitting down on a chair

Figure 3.
An example of magnitude of the force vector by combining the measurements from all three axes from accelerometer

An example of the traveling speed based on GPS data is illustrated in Figure 4 as a subject walking to a car, driving to the destination, then walking away the car as he arrives.

For the audio sensor data, we sample the audio signal at 8 kHz and extract the running average envelope, which gives us a smoother signal (less noise) than its original signal and peak envelope as our interest in the loudness of the ambient noise (amplitude of the audio signal). We compute the running average envelope (e) with window size of 50 using Equation (3):

$$e(n) = \frac{1}{w}\{g(n-w) + g(n-w+1) + \cdots + g(n-1) + g(n) + g(n+1) + \cdots + g(n+w-1) + g(n+w)\}, \tag{3}$$

where $g(n)$ is the amplitude of audio signal with $n = \{1, 2, 3, \ldots\}$ and w is the size of window.

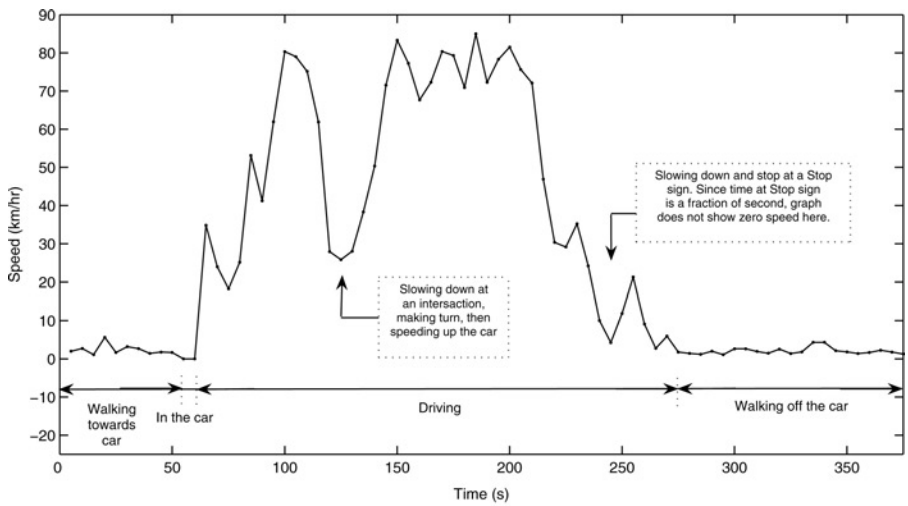
Figure 5 shows an example of the running average envelope while a subject is walking to a car, listening to music while driving, and walking away from the car after parking.

4.4 Context classifier

The context classifier is used in the offline training process (step 1) to take preprocessed data and project them onto feature space creating an initial "context map" with M trained data arrays.

With our preprocessed data, the (labeled) input data array of the classifier (x) at any interval of time T can be expressed as follows:

$$x_m = \begin{bmatrix} Var(A_m) \\ E(S_m) \\ E(E_m) \end{bmatrix}, \tag{4}$$



Notes: Data show the subject walking towards a car, driving, then walking away from the car as he reaches the destination

Figure 4.
An example of traveling speed based on GPS information

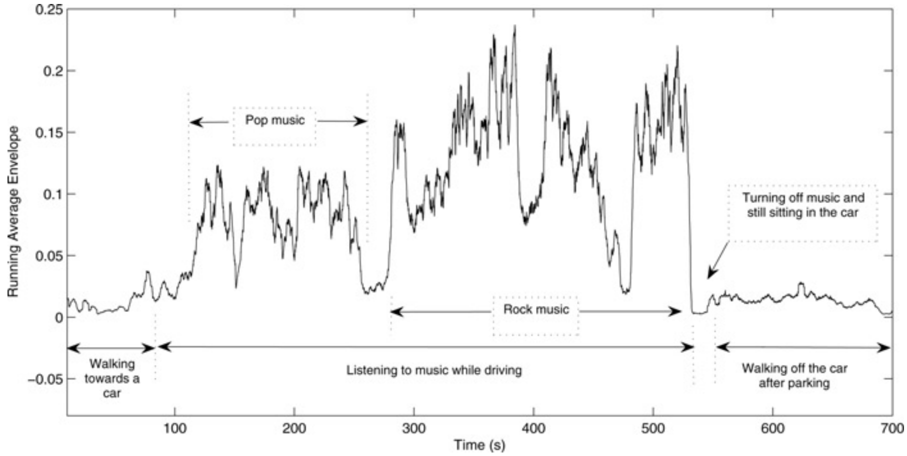


Figure 5.
An example of running average envelope while a subject is walking to a car, driving with music on, and walking away from the car after parking

where $A_m = \{a_m(1), a_m(2), \ldots, a_m(n_a)\}$, $S_m = \{s_m(1), s_m(2), \ldots, s_m(n_s)\}$, $E_m = \{e_m(1), e_m(2), \ldots, e_m(n_e)\}$, and n_a, n_s, n_e are the total numbers of data points within T of A_m, S_m, and E_m, respectively. We take the variance ($Var(\cdot)$) of A_m and expected values ($E(\cdot)$) of S_m and E_m. Hence the training data matrix for constructing the initial context map is $X_{trained} = \{x_1, x_2, \ldots, x_M\}$.

To project our training data onto a context map, we apply PCA (Jolliffe, 2002). We transform our three-dimensional input data to two-dimensional feature space by retaining two principal components that have the maximum variation in the original data array, namely the first and second principal components, i.e.

$$Y = W'_C X, \tag{5}$$

where Y is the data on a transformed space (or context map in our case), X is the data matrix, and W_C is the first C singular vectors ($C = 2$ in our case) where $W = [w_1 \quad w_2 \quad \ldots \quad w_p]$ (p is the original data's dimensionality, e.g. $p = 3$ in our case), the order of w is according to the variance or eigen value i.e. $var(w'_i X) = \lambda_i$ and $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_p$, and $'$ denotes transpose.

4.5 Context inference engine
The context inference engine is in the online inferring process (step 2), which takes a new (unlabeled) preprocessed data array along with the trained data arrays, projects them onto context space, and makes an initial classification for the new data based on k-NN algorithm (Shakhnarovich *et al.*, 2006) using the Euclidean distance. The initial classification is then fed into a transition supervision process based on a finite state machine model to make the final inference.

Thereby, the input data matrix for the PCA is $X = \{X_{trained}, x_{new}\}$ such that the new and trained data are transformed by the same function. With the new coordinates, the new data is then classified to the most likely context state (Z), which is the most common class amongst the k nearest neighbors in the context space, where $Z \in \{\text{UR}, \text{IR-V}, \text{UR-N}\}$. The initial classified context state then undergoes the supervision process to supervise the transitions from one context state to another. This supervision process uses a finite state machine architecture where each state represents the user's context and transitions are represented by edges between states.

4.6 User preference learning

This is a process of learning the user's personal preference. It is inevitable that the initial context map does not fit perfectly to the user's preference. This process is therefore essential to personalizing ContextAlert. The process can be as simple as the flow diagram shown in Figure 6. Once the user makes a change to the alert mode, the user will be prompted to have ContextAlert learn his/her setting. If the answer is "Yes," then the context map is adjusted accordingly. However, if the answer is "No," then no learning is needed and hence the context map is not modified.

4.7 Adaptive learning

With the proposed model, ContextAlert would start out highly adaptive with a high learning rate, would gradually become fixed as number of learned data increases. After this stage, it would be hardly capable of learning any more, which would create a problem as the system needs to remain adaptive. Overwriting previously learned data with the new learning can improve the adaptivity of the system. However, the tradeoff is known in the field of machine learning as the Stability-pasticity dilemma or catastrophic forgetting (French, 1999), which refers to the problem of designing a learning system to remain plastic or adaptive and preserve its previously learned knowledge while continuing to learn new things, which can also mean preventing the new learning from washing away the memories of prior learning.

To address this challenge in designing a context-awareness system, we propose a learning mechanism that remains adaptive while keeping catastrophic forgetting minimal. The adaptivity (Λ) can be defined simply as a learning rate for new data as:

$$\Lambda = \frac{\text{Amount of New Learning Data}}{\text{Amount of Learned Data}}. \tag{6}$$

In our case, the amount of new learning data is one and the amount of learned data is $M/3$ for each context state. Thus the learning rate decreases exponentially with M. To stay adaptive, M must be fixed and hence removing previously learned data is an option. In this approach, we cannot avoid the catastrophic forgetting problem. Nevertheless, we can minimize it.

Forgetting is a loss of memories, which can be quantified as a difference between the set of prior memories before and after a new learning. Let $\xi^{(b)}$ and $\xi^{(a)}$ denote the set of prior memories in three-dimensional space before and after a new learning, respectively.

Definition 2. If $x_k^{(b)}$ is the kth memory point before a new learning in three-dimensional space (d_1, d_2, d_3) and $x_k^{(a)}$ is the kth memory point after a new learning,

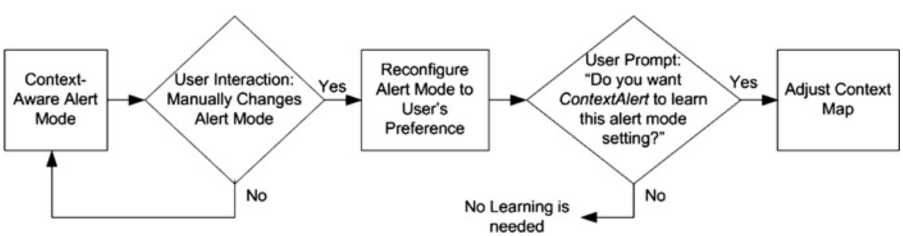


Figure 6.
User preference learning process flow

then the difference between the $x_k^{(b)}$ and $x_k^{(a)}$ (γ_k) can be computed using Euclidean distance as:

$$\gamma_k = \sqrt{(x_k^{(b)}(d_1) - x_k^{(a)}(d_1))^2 + (x_k^{(b)}(d_2) - x_k^{(a)}(d_2))^2 + (x_k^{(b)}(d_3) - x_k^{(a)}(p_3))^2}. \quad (7)$$

If $x_k^{(a)}$ does not exist (or has been removed), then $\gamma_k = \infty$ (complete loss of memory).

Definition 3. If $\xi^{(b)} = \{x_1^{(b)}, x_2^{(b)}, \ldots, x_M^{(b)}\}$ and $\xi^{(a)} = \{x_1^{(a)}, x_2^{(a)}, \ldots, x_M^{(a)}\}$, then the total loss of memories (Γ) is the sum of γ_k for $k = 1, 2, \ldots, M$, i.e.,

$$\Gamma = \sum_{k=1}^{M} \gamma_k. \quad (8)$$

To minimize the loss of memory from a new learning, we merge two nearest memory points to one memory point located at the mid point between the two. If $x_m^{(a)}$ is the merging of $x_i^{(b)}$ and $x_j^{(b)}$, then the total loss of memory is:

$$\begin{aligned}\Gamma &= \sqrt{(x_i^{(b)}(d_1) - x_m^{(a)}(d_1))^2 + (x_i^{(b)}(d_2) - x_m^{(a)}(d_2))^2 + (x_i^{(b)}(d_3) - x_m^{(a)}(d_3))^2} \\ &\quad + \sqrt{(x_j^{(b)}(d_1) - x_m^{(a)}(d_1))^2 + (x_j^{(b)}(d_2) - x_m^{(a)}(d_2))^2 + (x_j^{(b)}(d_3) - x_m^{(a)}(d_3))^2} \\ &= 2\sqrt{(x_i^{(b)}(d_1) - x_m^{(a)}(d_1))^2 + (x_i^{(b)}(d_2) - x_m^{(a)}(d_2))^2 + (x_i^{(b)}(d_3) - x_m^{(a)}(d_3))^2},\end{aligned} \quad (9)$$

and the merged memory point is occupied at:

$$\left(\frac{x_i^{(b)}(d_1)) + x_j^{(b)}(d_1)}{2}, \frac{x_i^{(b)}(d_2)) + x_j^{(b)}(d_2)}{2}, \frac{x_i^{(b)}(d_3)) + x_j^{(b)}(d_3)}{2}\right).$$

As an example, a graphical representation of the merging process is illustrated in Figure 7.

To summarize our design, a detailed algorithm of the ContextAlert is given in Figure 8.

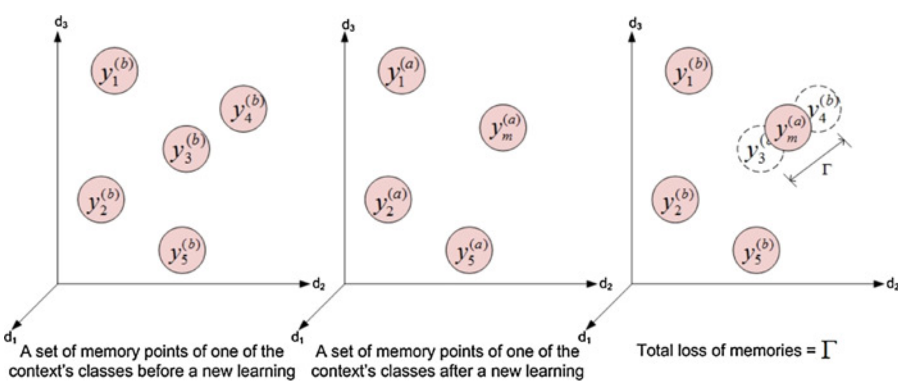


Figure 7.
An example of graphical representation of the merging process for adaptive learning

Input: Context map's data matrix ($X_{trained}$) and input data array (x_{new})

Output: Context-aware alert mode and a new context map's data matrix ($X_{new\_trained}$)

1. Project $X = \{X_{trained,x_{new}}\}$ onto the context space using PCA i.e. $Y = W_C' X$;
2. Classify y_{new} (transformed x_{new}) to the context $Z \in$ {UR,IR-V,IR-N} using k-NN;
3. Set the alert mode according to Definition 1;
4. IF There is an overwrite setting by the user
5. Prompt the user to have the system learned the setting;
6. IF Answer is "Yes"
7. Reclassify y_{new} according to the new setting;
8. ELSE
9. Do nothing;
10. END IF
11. END IF
12. $X_{new\_trained}$ = Merging of two nearest memory points of x_{new}'s class and the rest of $X_{trained}$;
13. Return the context-aware alert mode;
14. Return $X_{new\_trained}$;

Figure 8.
Context-aware alert mode

5. Experimental results

In this section, we describe our datasets (section 5.1) as well as conduct four experimental studies to evaluate our approach. In section 5.2, we show the impact of learning by comparing the performance of a model that uses only a fixed initial context map (no learning from new data) to a model that starts off with the same initial context map but its context map grows as it learns new data. In section 5.3, we once again point out that the growth of the context map with new learnings lowers the adaptivity rate and causes "the curse of dimensionality." We thus compare the performance of a model with growing context map with our proposed merging-based context map model (MCM). In section 5.4, we show the impact of the proposed supervision process that can improve the performance of the model. In section 5.5, we show the impact of applying PCA to our model by comparing the performance of our model with and without using PCA.

5.1 Datasets

For training, we collected data from three different subjects. Each subject performed ten different activities shown in Table I. Each activity was performed continuously for ten minutes by each subject. With a time interval (T) of five seconds (buffer time), we had 120 labeled data arrays. With ten different activities and 120 data arrays per subject, we thereby had 3,600 labeled data arrays available for constructing the initial context map.

For testing, we collected data from a different group of participating subjects. There were four subjects in this testing group. Each subject performed five different sequences of activities, which are listed on Table II. Each sequence was about one hour.

Table I.
A list of the ten different activities and their corresponding context states

Context state	Activity
UR	Attending a meeting Attending a class Watching movie at a theater Reading books in a library Working in an office
IR-N	Walking Jogging or running Eating at a restaurant Shopping at a supermarket
IR-V	Driving a car

Notes: Four participating subjects performed 20 min of each activity from which the training data arrays were obtained

Table II.
A list of five different sequences of activities with the corresponding context state and approximate duration

Sequence number	Sequence of activities with the corresponding context state and approximate duration
1	Jogging (IR-N, 3 min) ⇒ walking (IR-N, 2 min) ⇒ library (UR, 25 min) ⇒ walking (IR-N, 5 min) ⇒ driving (IR-V, 25 min)
2	Walking (IR-N, 5 min) ⇒ driving (IR-V, 30 min) ⇒ walking (IR-N, 10 min) ⇒ theater (UR, 15 min)
3	Walking (IR-N, 5 min) ⇒ library (UR, 25 min) ⇒ walking (IR-N, 10 min) ⇒ restaurant (IR-N, 20 min)
4	Meeting (UR, 25 min) ⇒ walking (IR-N, 3 min) ⇒ running (IR-N, 2 min) ⇒ class (UR, 30 min)
5	Working (UR, 10 min) ⇒ walking (IR-N, 5 min) ⇒ driving (IR-V, 25 min) ⇒ walking (IR-N, 5 min) ⇒ shopping (IR-N, 15 min)

Notes: Each sequence was about one hour. Testing data arrays were obtained from having each of four subjects performed these sequences

These sequences consisted of all ten activities listed on Table I, 31,690 s (6,338 data arrays) of UR, 21,430 s (4,286 data arrays) of IR-N, 18,940 s (3,788 data arrays) of IR-V, and total of 14,412 testing data arrays.

The subjects were asked to keep detailed time logs of activities performed, which were then used to do hand-labeling of the testing data.

5.2 Impact of learning

Ideally, we would like to have a model with one fixed context map that works perfectly for any user but this is not realistic. Therefore, in this section, we attempt to show that such a model performs well to some extent. However, we can improve its performance by learning from the user.

This experiment and others were set up as follows. The initial context map was constructed using 100 data arrays from each of the three context states by randomly selecting training data arrays obtained from the subjects in the training group. The model was tested with the data obtained from the five sequences of activities by four subjects described in section 5.1. The testing was done in the order of the sequence, i.e. testing with sequence 1, then sequence 2, then sequence 3, then sequence 3, and so on.

Table III shows the overall performance from four testing subjects in terms of accuracy rates per sequence (Acc./Seq.) as well as per context (Acc./Cont.) of a fixed initial context map model (FCM), which uses only the initial context map without learning from the user. The result per subject is shown in the Appendix.

Without learning, the FCM shows 76.63 percent overall accuracy. With the same initial context map, Table IV shows that we can achieve a much higher accuracy rate of 90.55 percent with a growing-with-learning context map model (GCM) that keeps all new learning data arrays as it is being tested. Hence the context map grows with learning (the amount of testing data). We assume here that the user corrects all misclassified data arrays (step 3 of the three-step approach) so that the GCM does not mislearn the data.

5.3 Adaptivity and the curse of dimensionality

A much higher accuracy rate of the GCM comes at a price. As the context map grows with learnings, its adaptivity decreases exponentially (according to Equation 6). This also increases the computational cost as the cost of k-NN rises with the number of learned data, which is a problem known as "the curse of dimensionality."

The adaptivity of GCM can be computed using Equation 6 as the average over three context data arrays as $\Lambda = (1/3)((1/6{,}338) + (1/4{,}286) + (1/3{,}788)) = 0.000218$. With our proposed MCM, which merges the two nearest learned data arrays after each new learning in the context map, Table V shows that we can achieve a competitive accuracy rate compared with the GCM at 89.34 percent (about one percent lower).

	UR		IR-N		IR-V		Overall		
Sequence number	Hit	Miss	Hit	Miss	Hit	Miss	Hit	Miss	Acc./Seq. (%)
1	1,138	0	316	50	1,012	108	2,466	158	93.98
2	750	0	638	76	1,356	108	2,744	184	93.72
3	1,252	0	564	1,112	0	0	1,816	1,112	62.02
4	2,686	0	208	84	0	0	2,894	84	97.18
5	512	0	594	644	18	1,186	1,124	1,830	38.05
Total	6,338	0	2,320	1,966	2,386	1,4,02	11,044	3,368	76.63
Acc./Cont. (%)	100.00		54.13		62.99		76.63		

Notes: The accuracy rate per context state (Acc./Cont.) is shown in the bottom of the table while the accuracy rate per sequence (Acc./Seq.) is shown in the last column

Table III. Performance of FCM in terms of hits and misses for each context state and each testing sequence of activities

	UR		IR-N		IR-V		Overall		
Sequence number	Hit	Miss	Hit	Miss	Hit	Miss	Hit	Miss	Acc./Seq. (%)
1	1,136	2	360	6	1,078	42	2,574	50	98.09
2	750	0	708	6	1,464	0	2,922	6	99.80
3	1,246	6	1,024	652	0	0	2,270	658	77.53
4	2,628	58	272	20	0	0	2,900	78	97.38
5	500	12	1,002	236	882	322	2,384	570	80.70
Total	6,260	78	3,366	920	3,424	364	13,050	1,362	90.55
Acc./Cont. (%)	98.77		78.53		90.39		90.55		

Notes: The accuracy rate per context state (Acc./Cont.) is shown in the bottom of the table while the accuracy rate per sequence (Acc./Seq.) is shown in the last column

Table IV. Performance of GCM in terms of hits and misses for each context state and each testing sequence of activities

However, the important improvement of the MCM over the GCM is a much higher adaptivity rate of $\Lambda = 1/100 = 0.01$. As the MCM prevents the adaptivity from decreasing while minimizing the loss of prior learning, the accuracy stays reasonably high with the context map that remains adaptive.

5.4 Impact of supervision process

With the supervision process, the context state transition is properly guided e.g. if the user is currently driving (IR-V), in the next five sections he/she will either be driving (IR-V) or walking away from the car (IR-N); he/she cannot be in a meeting or class. Adding the supervision process can help improve the accuracy of the model. In fact, experimental results in Table VI show that the accuracy rate of the MCM with the supervision process (MCM-S) is improved to 91.20 percent, which is higher than the MCM and GCM (with much better adaptivity rate than the GCM).

5.5 Impact of PCA

Typically, PCA is used to reduce the dimensionality of a dataset consisting of a large number of interrelated variables. In our case, PCA is used not only to reduce the dimensionality of our data matrices but to also reduce the noise from the embedded sensors (see Table VII). This noise reduction process can help improve the performance of the classifier and hence improve the accuracy of the model. Without applying PCA,

Table V. Performance of MCM in terms of hits and misses for each context state and each testing sequence of activities

	UR		IR-N		IR-V		Overall		
Sequence number	Hit	Miss	Hit	Miss	Hit	Miss	Hit	Miss	Acc./Seq. (%)
1	1,128	10	362	4	1,082	38	2,572	52	98.02
2	750	0	706	8	1,464	0	2,920	8	99.73
3	1,246	6	958	718	0	0	2,204	724	75.27
4	2,646	40	266	26	0	0	2,912	66	97.78
5	506	6	788	450	974	230	2,268	686	76.78
Total	6,276	62	3,080	1,206	3,520	268	12,876	1,536	89.34
Acc./Cont. (%)	99.02		71.86		92.93		89.34		

Notes: The accuracy rate per context state (Acc./Cont.) is shown in the bottom of the table while the accuracy rate per sequence (Acc./Seq.) is shown in the last column

Table VI. Performance of MCM-S in terms of hits and misses for each context state and each testing sequence of activities

	UR		IR-N		IR-V		Overall		
Sequence number	Hit	Miss	Hit	Miss	Hit	Miss	Hit	Miss	Acc./Seq. (%)
1	1,128	10	362	4	1,120	0	2,610	14	99.47
2	750	0	706	8	1,464	0	2,920	8	99.73
3	1,246	6	958	718	0	0	2,204	724	75.27
4	2,646	40	266	26	0	0	2,912	66	97.78
5	506	6	788	450	1,204	0	2,498	456	84.56
Total	6,276	62	3,080	1,206	3,788	0	13,144	1,268	91.20
Acc./Cont. (%)	99.02		71.86		100.00		91.20		

Notes: The accuracy rate per context state (Acc./Cont.) is shown at the bottom of the table while the accuracy rate per sequence (Acc./Seq.) is shown in the last column the accuracy of MCM-S is decreased to 85.64 percent as shown by the experimental results (see Appendix, Table AV).

We have conducted several experimental studies to evaluate our approach in designing a model for ContextAlert. To summarize the results, Table VIII shows the overall accuracy of each model. We have shown that "learning" improves the accuracy of the model but decreases adaptivity, the "merging-based model" helps maintains adaptivity with a reasonable accuracy rate, the "supervision process" is a key element that improves performance, and "PCA" is used to reduce sensor noise that can degrade the performance of the model. From Table VIII, our proposed model (MCM-S) has the highest performance in both accuracy rate and adaptivity.

Note that the result per subject of each model is available in the Appendix.

6. Limitations of the study

We are aware of the following limitations of this study:

- The evaluation of the user's preference learning step (step 3 of the three-step approach) cannot be done in this current study due to the capability of the current model of the G1 phone.
- With a limited number of testing data, we can only demonstrate the impact of learning and adaptivity to some extent. We are certain that a much longer period of testing would yield much clearer results than the current study.
- Similarly, with a larger number of testing subjects, the performance of our model would have been evaluated more accurately. In this study, we have learned that it is very difficult to recruit subjects to perform sequences of experiments in extended hours due to availability, willingness, and enthusiasm.

	UR		IR-N		IR-V		Overall		
Sequence number	Hit	Miss	Hit	Miss	Hit	Miss	Hit	Miss	Acc./Seq. (%)
1	1,138	0	290	76	1,120	0	2,548	76	97.10
2	750	0	580	134	1,464	0	2,794	134	95.42
3	1,250	2	534	1,142	0	0	1,784	1,144	60.93
4	2,686	0	208	84	0	0	2,894	84	97.18
5	512	0	606	632	1,204	0	2,322	632	78.61
Total	6,336	2	2,218	2,068	3,788	0	12,342	2,070	85.64
Acc./Cont. (%)	99.97		51.75		100.00		85.64		

Notes: The accuracy rate per context state (Acc./Cont.) is shown at the bottom of the table while the accuracy rate per sequence (Acc./Seq.) is shown in the last column

Table VII. Performance of MCM-S(no PCA) in terms of hits and misses for each context state and each testing sequence of activities

Model	Average adaptivity	Accuracy rate (%)
FCM	0.01	76.63
GCM	0.0000218	90.55
MCM	0.01	89.34
MCM-S	0.01	91.20
MCM-S (no PCA)	0.01	85.64

Table VIII. Overall performance comparison of different models in terms of adaptivity and accuracy rate

7. Conclusion

Forgetting to switch to vibrate mode while in a movie theater or a meeting, and taking the risk of picking up a phone call while driving can be avoided if the phone is smart enough to recognize its user's situational context. As the first step towards that direction, we propose a design for a context-aware mobile computing model known as ContextAlert that can intelligently switches the alert mode according to the user's context. We divide the user's context into three states: UR, IR-V, and IR-N. The alert mode is to be set to the recognized context state as vibrate, handsfree, and ringer mode for UR, IR-V, and IR-N, respectively. We have proposed a three-step approach in design based on the embedded sensor data from accelerometer, GPS antenna, and microphone of a G1 phone. We have evaluated our model in several aspects using training and testing data collected from participating subjects. Based on the experiments, the proposed model has shown a promising result. Nevertheless, our work had some limitations, such as capability of the phone, amount of testing data, and duration of testing. In our future work, we will continue to examine our model to improve its performance as well as investigate other applications of the model.

References

Adams, B., Phung, D. and Venkatesh, S. (2006), "Extraction of social context and application to personal multimedia exploration", *Multimedia'06: Proceedings of the 14th Annual ACM International Conference on Multimedia, ACM, New York, NY*, pp. 987-96.

Android (2009), "Android developers", available at: http://developer.android.com/

Bao, L. and Intille, S.S. (2004), "Activity recognition from user-annotated acceleration data", in *Pervasive*, pp. 1-17.

Bernardin, K. and Stiefelhagen, R. (2007), "Audio-visual multi-person tracking and identification for smart environments", *Multimedia '07: Proceedings of the 15th International Conference on Multimedia, ACM, New York, NY*, pp. 661-70.

Buriano, L. (2006), "Exploiting social context information in context-aware mobile tourism guides", in *Mobile Guide 2006*, ACM.

Chen, G. and Kotz, D. (2000), "A survey of context-aware mobile computing research", Technical Report, Hanover, NH.

Chen, H., Joshi, A. and Finin, T. (2001), "Dynamic service discovery for mobile computing: intelligent agents meet jini in the aether", *Cluster Computing*, Vol. 4 No. 4, pp. 343-54.

Chetan, S., Al-Muhtadi, J., Campbell, R. and Mickunas, M. (2005), "A middleware for enabling personal ubiquitous spaces", *Proceedings of the System Support for Ubiquitous Computing Workshop at Sixth Annual Conference on Ubiquitous Computing (UbiComp 2004)*.

Coen, M.H. (1998), "Design principles for intelligent environments", *AAAI '98/IAAI '98: Proceedings of the Fifteenth National/Tenth Conference on Artificial Intelligence/Innovative Applications of Artificial Intelligence, American Association for Artificial Intelligence, Menlo Park, CA*, pp. 547-54.

Czerwinski, S.E., Zhao, B.Y., Hodes, T.D., Joseph, A.D. and Katz, R.H. (1999), "An architecture for a secure service discovery service", *MobiCom '99: Proceedings of the 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking, ACM, New York, NY*, pp. 24-35.

Davis, B. and Karahalios, K. (2005), "Telelogs: a social communication space for urban environments", *MobileHCI '05: Proceedings of the Seventh International Conference on Human Computer Interaction with Mobile Devices and Services, ACM, New York, NY*, pp. 231-4.

Dey, A.K. (2001), "Understanding and using context", *Personal and Ubiquitous Computing*, Vol. 5 No. 1, pp. 4-7.

Eagle, N. and Pentland, A. (2005), "Social serendipity: mobilizing social software", *IEEE Pervasive Computing*, Vol. 4 No. 2, pp. 28-34.

Eagle, N. and Pentland, A. (2006), "Reality mining: sensing complex social systems", *Personal and Ubiquitous Computing*, Vol. 10 No. 4, pp. 255-68.

French, R.M. (1999), "Catastrophic forgetting in connectionist networks", *Trends in Cognitive Sciences*, Vol. 3 No. 4, pp. 128-35.

Friday, A., Davies, N. and Catterall, E. (2001), "Supporting service discovery, querying and interaction in ubiquitous computing environments", *MobiDe '01: Proceedings of the Second ACM International Workshop on Data Engineering for Wireless and Mobile Access, ACM, New York, NY*, pp. 7-13.

Governors Highway Safety Association (2009), "Cell phone driving laws", available at: www.ghsa.org/html/stateinfo/laws/cellphone_laws.html

Grosse, P.W., Holzapfel, H. and Waibel, A. (2008), "Confidence based multimodal fusion for person identification", *MM '08: Proceedings of the 16th ACM International Conference on Multimedia, ACM, New York, NY*, pp. 885-8.

Han, L., Jyri, S., Ma, J. and Yu, K. (2008), "Research on context-aware mobile computing", *AINA W '08: Proceedings of the 22nd International Conference on Advanced Information Networking and Applications – Workshops, IEEE Computer Society, Washington, DC*, pp. 24-30.

Hofer, T., Schwinger, W., Pichler, M., Leonhartsberger, G., Altmann, J. and Retschitzegger, W. (2003), "Context-awareness on mobile devices - the hydrogen approach", *HICSS '03: Proceedings of the 36th Annual Hawaii International Conference on System Sciences (HICSS'03) – Track 9, IEEE Computer Society, Washington, DC*, p. 292.1.

Husna, H., Phithakkitnukoon, S., Baatarjav, E.-A. and Dantu, R. (2008), "Quantifying presence using calling patterns", *COMSWARE*, IEEE, pp. 184-7.

Iso, T. and Yamazaki, K. (2006), "Gait analyzer based on a cell phone with a single three-axis accelerometer", *MobileHCI '06: Proceedings of the Eighth Conference on Human-Computer Interaction with Mobile Devices and Services, ACM, New York, NY*, pp. 141-4.

Jian, M.-S., Yang, K.S. and Lee, C.-L. (2008), "Context and location aware public/personal information service based on rfid system integration", *WTOS*, Vol. 7 No. 6, pp. 774-84.

Jin, G.H. Lee, S.B. and Lee, T.S. (2008), "Context awareness of human motion states using accelerometer", *Journal of Medical Systems*, Vol. 32 No. 2, pp. 93-100.

Jolliffe, I.T. (2002), *Principal Component Analysis*, 2nd ed., Springer, Berlin, October.

Kohonen, T. (2001), *Self-Organizing Map*, 3rd ed., Springer, New York, NY.

Koichi, S. (2004), "Measurement and analysis of human behavior using wearable sensors", *Proceedings of the 25th Asian Conference on Remote Sensing*, Vol. 2, pp. 1218-23.

Kostakos, V., O'Neill, E. and Shahi, A. (2006), "Building common ground for face to face interactions by sharing mobile device context." in Hazas, M., Krumm, J. and Strang, T. (Eds), *LoCA*, ser. Lecture Notes in Computer Science, Springer, Vol. 3987, pp. 222-38.

Kourogi, M. and Kurata, T. (2003), "Personal positioning based on walking locomotion analysis with self-contained sensors and a wearable camera", *ISMAR '03: Proceedings of the Second IEEE/ACM International Symposium on Mixed and Augmented Reality, IEEE Computer Society, Washington, DC*, p. 103.

Krause, A. (2006), "Context-aware mobile computing: learning context-dependent personal preferences from a wearable sensor array", *IEEE Transactions on Mobile Computing*, Vol. 5 No. 2, pp. 113-27.

Krause, A., Siewiorek, D.P., Smailagic, A. and Farringdon, J. (2003), "Unsupervised, dynamic identification of physiological and activity context in wearable computing", *ISWC '03:*

Proceedings of the Seventh IEEE International Symposium on Wearable Computers, IEEE Computer Society, Washington, DC, p. 88.

Laerhoven, K.V. and Cakmakci, O. (2000), "What shall we teach our pants?", *ISWC'00: Proceedings of the Fourth IEEE International Symposium on Wearable Computers, IEEE Computer Society, Washington, DC*, p. 77.

Lester, J., Hannaford, B. and Borriello, G. (2004), "Are you with me? – using accelerometers to determine if two devices are carried by the same person", in *Pervasive*, pp. 33-50.

Liu, X. (2006), "System support for pervasive multimedia systems", PhD dissertation, adviser-Shenoy, Prashant and Adviser-Corner, Mark D.

Lukowicz, P., Ward, J., Junker, H., Stager, M., Troster, G., Atrash, A. and Starner, T. (2004), "Recognizing workshop activity using body worn microphones and accelerometers", Lecture Notes in Computer Science, Vol. 3001/2004, pp. 18-32.

Munetoshi, U., Ken'ichiro, K., Atsuya, T. and Takeshi, H. (2004), "Autonomy position detection by using recognition of human walking motion", *IEICE Japan*, Vol. J87-A No. 1, pp. 78-86.

Oulasvirta, A., Raento, M. and Tiitta, S. (2005), "Contextcontacts: re-designing smartphone's contact book to support mobile awareness and collaboration", *MobileHCI'05: Proceedings of the Seventh International Conference on Human Computer Interaction with Mobile Devices and Services, ACM, New York, NY*, pp. 167-74.

Park, K.-L., Yoon, U.H. and Kim, S.-D. (2009), "Personalized service discovery in ubiquitous computing environments", *IEEE Pervasive Computing*, Vol. 8 No. 1, pp. 58-65.

Paulos, E. and Goodman, E. (2004), "The familiar stranger: anxiety, comfort and play in public places", *CHI'04: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, ACM, New York, NY*, pp. 223-30.

Prekop, P. and Burnett, M. (2003), "Activities, context and ubiquitous computing", *Computer Communications*, Vol. 26, p. 1168.

Schilit, B., Adams, N. and Want, R. (1994), "Context-aware computing applications", *WMCSA'94: Proceedings of the 1994 First Workshop on Mobile Computing Systems and Applications, IEEE Computer Society, Washington, DC*, pp. 85-90.

Schilit, B.N. and Theimer, M.M. (1994), "Disseminating active map information to mobile hosts", *IEEE Network*, Vol. 8, pp. 22-32.

Schmidt, A., Aidoo, K.A., Takaluoma, A., Tuomela, U., Laerhoven, K.V. and d. Velde, W.V. (1999), "Advanced interaction in context", *HUC'99: Proceedings of the First International Symposium on Handheld and Ubiquitous Computing, Springer-Verlag, London*, pp. 89-101.

Shakhnarovich, G., Darrell, T. and Indyk, P. (2006), *Nearest-Neighbor Methods in Learning and Vision: Theory and Practice (Neural Information Processing)*, The MIT Press, Cambridge, MA.

Siewiorek, D., Smailagic, A., Furukawa, J., Krause, A., Moraveji, N., Reiger, K., Shaffer, J. and Wong, F.L. (2003), "Sensay: a context-aware mobile phone", *ISWC'03: Proceedings of the Seventh IEEE International Symposium on Wearable Computers, IEEE Computer Society, Washington, DC*, p. 248.

Suutala, J. and Röning, J. (2008), "Methods for person identification on a pressure-sensitive floor: experiments with multiple classifiers and reject option", *Information Fusion*, Vol. 9 No. 1, pp. 21-40.

T-Mobile (2009), "T-mobile's g1 phone", available at: www.t-mobileg1.com/

Toninelli, A., Corradi, A. and Montanari, R. (2008), "Semantic-based discovery to support mobile context-aware service access", *Computer Communications*, Vol. 31 No. 5, pp. 935-49.

Weiser, M. (1993), "Some computer science issues in ubiquitous computing", *Communications of the ACM*, Vol. 36 No. 7, pp. 75-84.

Weiser, M. (1995), "The computer for the 21st century", in Baecker, R.M., Grudin, J., Buxton, W.A.S. and Greenberg, S. (Eds), *Human-Computer Interaction: Toward the Year 2000*, Morgan Kaufmann, San Francisco, CA, pp. 933-40.

Yi, J.S., Choi, Y.S., Jacko, J.A. and Sears, A. (2005), "Context awareness via a single device-attached accelerometer during mobile computing", *MobileHCI'05: Proceedings of the Seventh International Conference on Human Computer Interaction with Mobile Devices and Services, ACM, New York, NY*, pp. 303-6.

Zhu, F., Mutka, M. and Ni, L. (2003), "Research on context-aware mobile computing", in *Splendor: A Secure, Private, and Location-Aware Service Discovery Protocol Supporting Mobile Services*, Fort Worth, TX; *Proceedings of the First IEEE Ann. Conf. Pervasive Computing and Comm. (PerCom'03)*, pp. 1-8.

Appendix

Detailed experimental results

This section includes the experimental results of each subject for each model (see Tables AI-AV).

	UR		IR-N		IR-V		Overall		
Sequence number	Hit	Miss	Hit	Miss	Hit	Miss	Hit	Miss	Acc./Seq. (%)
Subject 1									
1	300	0	96	0	288	12	684	12	98.28
2	180	0	142	38	318	42	640	80	88.89
3	300	0	142	278	0	0	442	278	61.39
4	660	0	40	20	0	0	700	20	97.22
5	120	0	116	184	12	288	248	472	34.44
Total	1,560	0	536	520	618	342	2,714	862	75.89
Acc./Cont. (%)	100.00		50.76		64.38		75.89		
Subject 2									
1	270	0	70	26	182	78	522	104	83.39
2	220	0	172	8	338	14	730	22	97.07
3	300	0	138	292	0	0	438	292	60.00
4	666	0	46	24	0	0	712	24	96.74
5	132	0	142	180	4	304	278	484	36.48
Total	1,588	0	568	530	524	396	2,680	926	74.32
Acc./Cont.	100.00		51.73		56.96		74.32		
Subject 3									
1	290	0	76	20	272	18	638	38	94.38
2	170	0	160	14	334	26	664	40	94.32
3	300	0	150	296	0	0	450	296	60.32
4	674	0	68	22	0	0	742	22	97.12
5	126	0	78	214	2	286	206	500	29.18
Total	1,560	0	532	566	608	330	2,700	896	75.08
Acc./Cont.	100.00		48.45		64.82		75.08		
Subject 4									
1	278	0	74	4	270	0	622	4	99.36
2	180	0	164	16	366	26	710	42	94.41
3	352	0	134	246	0	0	486	246	66.39
4	686	0	54	18	0	0	740	18	97.63
5	134	0	258	66	0	308	392	374	51.17
Total	1,630	0	684	350	636	334	2,950	684	81.18
Acc./Cont.	100.00		66.15		65.57		81.18		

Table AI. Performance of FCM

Sequence number	UR Hit	UR Miss	IR-N Hit	IR-N Miss	IR-V Hit	IR-V Miss	Overall Hit	Overall Miss	Acc./Seq. (%)
Subject 1									
1	300	0	96	0	300	0	696	0	100.00
2	180	0	174	6	360	0	714	6	99.17
3	296	4	208	212	0	0	504	216	70.00
4	652	8	56	4	0	0	708	12	98.33
5	108	12	252	48	236	64	596	124	82.78
Total	1,536	24	786	270	896	64	3,218	358	89.99
Acc./Cont.	98.46		74.43		93.33		89.99		
Subject 2									
1	268	2	92	4	218	42	578	48	92.33
2	220	0	180	0	352	0	752	0	100.00
3	300	0	184	246	0	0	484	246	66.30
4	642	24	64	6	0	0	706	30	95.92
5	132	0	282	40	226	82	640	122	83.99
Total	1,562	26	802	296	796	124	3,160	446	87.63
Acc./Cont.	98.36		73.04		86.52		87.63		
Subject 3									
1	290	0	94	2	290	0	674	2	99.70
2	170	0	174	0	360	0	704	0	100.00
3	300	0	332	114	0	0	632	114	84.72
4	656	18	90	0	0	0	746	18	97.64
5	126	0	170	122	214	74	510	196	72.24
Total	1,542	18	860	238	864	74	3,266	330	90.82
Acc./Cont.	98.85		78.32		92.11		90.82		
Subject 4									
1	278	0	78	0	270	0	626	0	100.00
2	180	0	180	0	392	0	752	0	100.00
3	350	2	300	80	0	0	650	82	88.80
4	678	8	62	10	0	0	740	18	97.63
5	134	0	298	26	206	102	638	128	83.29
Total	1,620	10	918	116	868	102	3,406	228	93.73
Acc./Cont.	99.39		88.78		89.48		93.73		

Table AII.
Performance of GCM

Sequence number	UR Hit	UR Miss	IR-N Hit	IR-N Miss	IR-V Hit	IR-V Miss	Overall Hit	Overall Miss	Acc./Seq. (%)
Subject 1									
1	298	2	96	0	300	0	694	2	99.71
2	180	0	174	6	360	0	714	6	99.17
3	296	4	188	232	0	0	484	236	67.22
4	656	4	54	6	0	0	710	10	98.61
5	114	6	162	138	246	54	522	198	72.50
Total	1,544	16	674	382	906	54	3,124	452	87.36
Acc./Cont.	98.97		63.83		94.38		87.36		
Subject 2									
1	266	4	94	2	222	38	582	44	92.97
2	220	0	178	2	352	0	750	2	99.73
3	300	0	166	264	0	0	466	264	63.84

(*continued*)

Table AIII.
Performance of MCM

Sequence number	UR Hit	UR Miss	IR-N Hit	IR-N Miss	IR-V Hit	IR-V Miss	Overall Hit	Overall Miss	Acc./Seq. (%)
4	652	14	70	0	0	0	722	14	98.10
5	132	0	216	106	258	50	606	156	79.53
Total	1,570	18	724	374	832	88	3,126	480	86.69
Acc./Cont.	98.87		65.94		90.43		86.69		
Subject 3									
1	288	2	94	2	290	0	672	4	99.41
2	170	0	174	0	360	0	704	0	100.00
3	298	2	314	132	0	0	612	134	82.04
4	658	16	80	10	0	0	738	26	96.60
5	126	0	116	176	230	58	472	234	66.86
Total	1,540	20	778	320	880	58	3,198	398	88.93
Acc./Cont.	98.72		70.86		93.82		88.93		
Subject 4									
1	276	2	78	0	270	0	624	2	99.68
2	180	0	180	0	392	0	752	0	100.00
3	352	0	290	90	0	0	642	90	87.70
4	680	6	62	10	0	0	742	16	97.89
5	134	0	294	30	240	68	668	98	87.21
Total	1,622	8	904	130	902	68	3,428	206	94.33
Acc./Cont.	99.51		87.43		92.99		94.33		

Table AIII.

Sequence number	UR Hit	UR Miss	IR-N Hit	IR-N Miss	IR-V Hit	IR-V Miss	Overall Hit	Overall Miss	Acc./Seq. (%)
Subject 1									
1	298	2	96	0	300	0	694	2	99.71
2	180	0	174	6	360	0	714	6	99.17
3	296	4	188	232	0	0	484	236	67.22
4	656	4	54	6	0	0	710	10	98.61
5	114	6	162	138	300	0	576	144	80.00
Total	1,544	16	674	382	960	0	3,178	398	88.87
Acc./Cont.	98.97		63.83		100.00		88.87		
Subject 2									
1	266	4	94	2	260	0	620	6	99.04
2	220	0	178	2	352	0	750	2	99.73
3	300	0	166	264	0	0	466	264	63.84
4	652	14	70	0	0	0	722	14	98.10
5	132	0	216	106	308	0	656	106	86.09
Total	1,570	18	724	374	920	0	3,214	392	89.13
Acc./Cont.	98.87		65.94		100.00		89.13		
Subject 3									
1	288	2	94	2	290	0	672	4	99.41
2	170	0	174	0	360	0	704	0	100.00
3	298	2	314	132	0	0	612	134	82.04
4	658	16	80	10	0	0	738	26	96.60
5	126	0	116	176	288	0	530	176	75.07
Total	1,540	20	778	320	938	0	3,256	340	90.55
Acc./Cont.	98.72		70.86		100.00		90.55		

(*continued*)

Table AIV. Performance of MCM-S

Sequence number	UR Hit	UR Miss	IR-N Hit	IR-N Miss	IR-V Hit	IR-V Miss	Overall Hit	Overall Miss	Acc./Seq. (%)
Subject 4									
1	276	2	78	0	270	0	624	2	99.68
2	180	0	180	0	392	0	752	0	100.00
3	352	0	290	90	0	0	642	90	87.70
4	680	6	62	10	0	0	742	16	97.89
5	134	0	294	30	308	0	736	30	96.08
Total	1,622	8	904	130	970	0	3,496	138	96.20
Acc./Cont.	99.51		87.43		100.00		96.20		

Table AIV.

Sequence number	UR Hit	UR Miss	IR-N Hit	IR-N Miss	IR-V Hit	IR-V Miss	Overall Hit	Overall Miss	Acc./Seq. (%)
Subject 1									
1	300	0	86	10	300	0	686	10	98.56
2	180	0	140	40	360	0	680	40	94.44
3	298	2	138	282	0	0	436	284	60.56
4	660	0	42	18	0	0	702	18	97.50
5	120	0	120	180	300	0	540	180	75.00
Total	1,558	2	526	530	960	0	3,044	532	85.12
Acc./Cont.	99.87		49.81		100.00		85.12		
Subject 2									
1	270	0	62	34	260	0	592	34	94.57
2	220	0	160	20	352	0	732	20	97.34
3	300	0	136	294	0	0	436	294	59.73
4	666	0	46	24	0	0	712	24	96.74
5	132	0	154	168	308	0	594	168	77.95
Total	1,588	0	558	540	920	0	3,066	540	85.02
Acc./Cont.	100.00		50.82		100.00		85.02		
Subject 3									
1	290	0	74	22	290	0	654	22	96.75
2	170	0	134	40	360	0	664	40	94.32
3	300	0	138	308	0	0	438	308	58.71
4	674	0	68	22	0	0	742	22	97.12
5	126	0	96	196	288	0	510	196	72.24
Total	1,560	0	510	588	938	0	3,008	588	83.65
Acc./Cont.	100.00		46.45		100.00		83.65		
Subject 4									
1	278	0	68	10	270	0	616	10	98.40
2	180	0	146	34	392	0	718	34	95.48
3	352	0	122	258	0	0	474	258	64.75
4	686	0	52	20	0	0	738	20	97.36
5	134	0	236	88	308	0	678	88	88.51
Total	1,630	0	624	410	970	0	3,224	410	88.72
Acc./Cont.	100.00		60.35		100.00		88.72		

Table AV.
Performance of MCM-S(noPCA)

Corresponding author
Santi Phithakkitnukoon can be contacted at: santi@mit.edu