PAPER    *Special Section on Emerging Technologies of Ubiquitous Computing Systems*

# Shaka: User Movement Estimation Considering Reliability, Power Saving, and Latency Using Mobile Phone

Arei KOBAYASHI[†a)], Shigeki MURAMATSU[†], Daisuke KAMISAKA[†], Takafumi WATANABE[†], Atsunori MINAMIKAWA[†], *Nonmembers*, Takeshi IWAMOTO[††], *and* Hiroyuki YOKOYAMA[†], *Members*

**SUMMARY**    This paper proposes a method for using an accelerometer, microphone, and GPS in a mobile phone to recognize the movement of the user. Past attempts at identifying the movement associated with riding on a bicycle, train, bus or car and common human movements like standing still, walking or running have had problems with poor accuracy due to factors such as sudden changes in vibration or times when the vibrations resembled those for other types of movement. Moreover, previous methods have had problems with has the problem of high power consumption because of the sensor processing load. The proposed method aims to avoid these problems by estimating the reliability of the inference result, and by combining two inference modes to decrease the power consumption. Field trials demonstrate that our method achieves 90% or better average accuracy for the seven types of movement listed above. Shaka's power saving functionality enables us to extend the battery life of a mobile phone to over 100 hours while our estimation algorithm is running in the background. Furthermore, this paper uses experimental results to show the trade-off between accuracy and latency when estimating user activity.
*key words:*  activity recognition, accelerometer, microphone, GPS, reliability, power saving, latency

## 1.    Introduction

Mobile phones have become increasingly sophisticated in recent years with the integration of sensors such as cameras, GPS and RF-ID, and the provision of various new services such as human navigation and mobile payments. Another category of mobile phone services currently being considered is those based on an awareness of user presence using the SIP protocol and so forth. How to obtain this user presence information is one of the main technical challenges for realizing these services.

This paper looks in particular at the user movement aspect of user presence information. Here, movement refers to both the person's state of activity, such as standing still, walking or running, and to the movement of any vehicle they are in such as a bicycle, car, bus or train. A wide range of potential applications will become possible if these states can be determined automatically using a mobile phone, such as automatically switching the mobile phone to silent mode when riding on public transport, detecting any abnormal movement of a commuting student or estimating a person's calorie usage from their movement.

This paper proposes a method that uses an accelerometer, a microphone, and a GPS implemented in a mobile phone to estimate the user's movement state automatically. The method improves estimation accuracy by estimating the reliability of inference results, and by combining two inference modes to decrease the power consumption. The power spectrum calculated from the accelerometer measurements is used to identify the running, walking, stationary and bicycle riding states, the power spectrum calculated from the microphone measurements is used to identify when the user is riding in a car, and the average speed calculated from the GPS measurements is used to identify riding in a train or bus. Also, the method aims to avoid the problems with poor accuracy caused by sudden changes in vibration or periods of time when the measurements resemble those for a different type of movement.

This paper starts by summarizing related research, then describes the requirements and associated problems. The proposed method is explained next, followed by the results of performance testing and conclusions.

## 2.    Related Work

### 2.1    User Movement Estimation Based on Activity Recognition

Numerous different methods have been proposed for human activity recognition. Kern et al.[1] attached acceleration sensors to different parts of the body and used these to identify activities such as "sitting", "standing", "walking", "ascending stairs", "descending stairs", "shaking hands", "writing on a blackboard" and "typing on a computer". Intille et al.[2],[3] fitted 5 accelerometers to the waist, wrists, and ankles, and used these to identify 20 different types of activity. An advantage of this work is that threshold settings are not necessary adjusted for each individual user. The SenSay[4] system implemented applications such as switching the phone ring to silent mode on the basis of four different states: "busy", "active", "idle" and "normal". WearNET[5] estimated the location, environment, user status, and use activity using multiple sensors. Lee[6] identified when a user was walking, climbing stairs, turning left, or turning right using an accelerometer, a geomagnetic sensor, and a gyroscope. Choudhury et al.[7] estimated 12 types of user activity using a specially designed mobile sensing device and supervised training.

Yin et al. [8] estimated abnormal activity such as slipping on the ground and falling down backwards using wearable sensors. Györbíró et al. [9] estimated six types of user activity using feed-forward/ back-propagation neural networks and an accelerometer, geomagnetic sensor, and gyroscope called "MotionBand". Piero et al. [10] estimated ten types of user activity from wearable body sensors with a system designed for low power consumption.

Although past research has been able to identify a range of different human activities using various different sensors. In many cases the techniques are impractical because of their requirements about how to and how many sensors should be attached to the human body.

Iso et al. [11] used a mobile phone fitted with an accelerometer that identified the walking states (walking normally, going up and down stairs, brisk walking, running) regardless of where the user carried the mobile phone. Kawahara et al. [12] presented a method for identifying the "walking", "standing", "sitting", and "running" states that used a single accelerometer and identified where the sensor was fitted. Moreover, some research has been conducted using standard commercial mobile phones. Yang [13] identified sitting, standing, walking, running, driving, and bicycling activity using a mobile phone. Miluzzo et al. [14] used various sensors (microphone, accelerometer, GPS, and camera) implemented in a commercial smart phone for activity recognition. Brezmes et al. [15] implemented a real-time system for activity recognition using a mobile phone. Anderson et al. [16] monitored user activity levels for three types of user's activity to promote health and fitness. A weak point of these researches is the reliability of their performance evaluation, that is, the number of subjects is 10 at most. Kwapisz et al. [17] identified walking, jogging, ascending stairs, descending stairs, sitting, and standing activies using an accelerometer in an Android smart phone. Although, the experiment scale of this research is relatively large (29 users), there is room for improvement in the estimation accuracy. Kobayashi et al. [18] pointed out the problem of sudden changes in vibration and times when the vibrations resembled those for other types of movement. Moreover, while their method could identify seven types of user movement using HMM, there is also room for improvement in the estimation accuracy.

Furthermore, no research has been carried out into movement estimation methods that consider vehicle movement states such as travel on a car, bus or train and that also consider of reliability, power consumption, and the latency when implemented on an actual commercial mobile phone.

## 2.2 Location-Based User Movement Estimation

Much researches has been conducted into location-based user movement estimation. Kourogi et al. [19] estimated user location using GPS and an accelerometer. However, apart from the direction of movement, this approach suffers from a high level of noise in the acceleration value which limits where the sensor can be located.

Thrun et al. [20] proposed a robot location estimation method called SLAM (Simultaneous Localization and Mapping). This method estimates robot location using a HMM generated from interaction between the robot and its environment. Although this approach can be applied to human location estimation, this method needs a stereo camera and laser range finder for recognition of the environment, and these sensors need to be attached to the robot. Therefore, this approach is impractical for user movement estimation.

Aoki et al. [21] proposed a user location estimation method using a wearable camera. This method is a pattern matching approach that compares the captured image to predefined images. Therefore, this approach can be used for user movement estimation by storing predefined images of specific vehicles. However, performed frequent image capture from the camera in a mobile phone is impractical for reasons of power consumption and general practicality.

## 3. Requirements and Issues

This section defines the requirements for movement estimation using a mobile phone. Next, the associated problems are highlighted with reference to a previous method.

### 3.1 Requirements for User Movement Estimation Using a Mobile Phone

A system for estimating user movement using a mobile phone should identify the type of movement with a high degree of accuracy under the conditions listed below.

- Requirement 1: A single sensor of each type located in the mobile phone
  For reasons including cost and the burden on the user, it is desirable to use only a single sensor located in the mobile phone. As mobile phones may be held in many different ways, the estimation method must not be dependent on how the phone is held.
- Requirement 2: Independent of the user
  Considering the burden on the user, it is desirable that the method should not require functions such as learning or setup that are dependent on the user.
- Requirement 3: Fully automatic estimation
  Considering the burden on the user, the estimation method should be fully automatic and not require any manual operation by the user.
- Requirement 4: Practical level of estimation processing load
  As the available processing power and battery on a mobile phone is limited, the processing load and power consumption associated with the estimation method must not be too large.

### 3.2 Issues with Previous Methods

To highlight the problems, Table 1 shows the estimation results obtained using a previous method by Kobayashi

**Table 1**  Estimation accuracy of previous method.

| Movement type | Precision | Recall | F measure |
|---|---|---|---|
| Running | 0.997 | 0.959 | 0.977 |
| Walking | 0.934 | 0.942 | 0.938 |
| Bike | 0.895 | 0.951 | 0.922 |
| Stationary | 0.732 | 0.741 | 0.736 |
| Car | 0.419 | 0.323 | 0.365 |
| Bus | 0.481 | 0.342 | 0.400 |
| Train | 0.525 | 0.448 | 0.483 |
| Average | 0.712 | 0.672 | 0.689 |



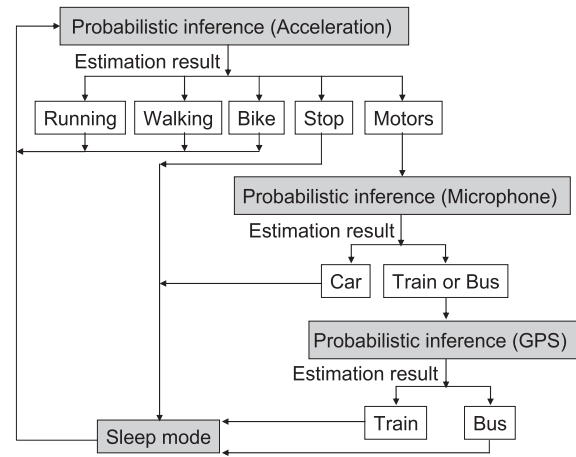**Fig. 1**  Processing sequence of proposed method.

et al. [18]. The estimation method used HMM to consider time-axis changes in the power spectrum calculated from the acceleration value. The method attempted to avoid problems with loss of accuracy caused by factors such as sudden changes in vibration or times during which vibration patterns resemble those of other states. The experimental data is the same as that referred to in Sect. 5.2. As Table 1 shows, the accuracy of this method was 0.9 or better for running, walking, and bike riding but much poorer for other types of movement. This means that the above problems cannot be avoided completely by using HMM to consider time-axis changes in the power spectrum (problem 1). Moreover, the previous method was not designed to minimize power consumption and therefore suffers from poor battery life (problem 2). This point is very important in mobile phones.

## 4. Proposed Method: "Shaka"

This section describes the "Shaka[†]" method which aims to solve these problems and satisfy the requirements described in the previous section. As specified in the list of requirements, the Shaka method uses a single accelerometer, microphone, and GPS in the mobile phone to estimate the user's movement, without being dependent on how the sensors are held (requirement 1), without being dependent on the user (requirement 2), and with all operation being fully automatic with no manual operation by the user (requirement 3). Also, the method uses sleep mode to keep the estimation processing load per iteration low (requirement 4). The proposed method also aims to solve the problem 1 by estimating the reliability of the inference result, and problem 2 by using sleep mode to minimize the power consumption associated with estimation.

### 4.1  Overview

The proposed method consists of three steps. Figure 1 shows the overall sequence of processing for Shaka. Estimation is performed by applying the probabilistic inference to the power spectrum obtained from the acceleration values. If the result indicates running, walking, biking, or stationary, this result is output and the estimation process changes to sleep mode. If result indicates travel in a motor vehicle, the probabilistic inference is applied to the power spectrum obtained from the recorded microphone data. If

the result indicates a car, this result is output and the estimation process changes to sleep mode. If the result indicates a train or bus, the probabilistic inference is applied to the average speed obtained from GPS positioning data to determine whether the user is in a train or a bus. This result is output and the estimation process changes to sleep mode. This allows the different types of movement to be identified with a high degree of accuracy while keeping the processing load to a minimum. The steps that make up this processing are described below.

### 4.2  Probabilistic Inference Using the Acceleration Data

Initially, Shaka produces an estimation result by applying probabilistic inference to the power spectrum calculated from the acceleration measurements. This proposed method divides the recognition time into multiple segments called *frames*. The method estimates movement type using a cluster model for each frame, and evaluates the reliability of the estimation result. The method seeks to improve the recognition accuracy by only using the recognition results from frames with high reliability called *key frames* to establish the movement type result.

Figure 2 shows an overview of the recognition process. The method achieves recognition using two different cluster models. The first model is the user movement model using the feature values obtained from the sensors to estimate the movement type. The second model is the reliability model using the movement type probabilities produced

---

[†]The name derives from a scene in the classic Chinese tale "Xi You Ji" (Monkey: Journey to the West) in which the character Songoku (Monkey) bets Shaka (Buddha) that he can take possession of the heavens. He makes a sketch of five pillars located at the end of the world then goes and urinates on them, but when he comes back it turns out he has only traveled around the palm of Shaka's hand. The name "Shaka" was chosen for this method to represent how we wanted the mobile phone to be like the palm of Shaka's hand whereby we can obtain a shrunk-down model of the user's world by using sensors fitted inside a mobile phone to obtain and share presence information about the user.
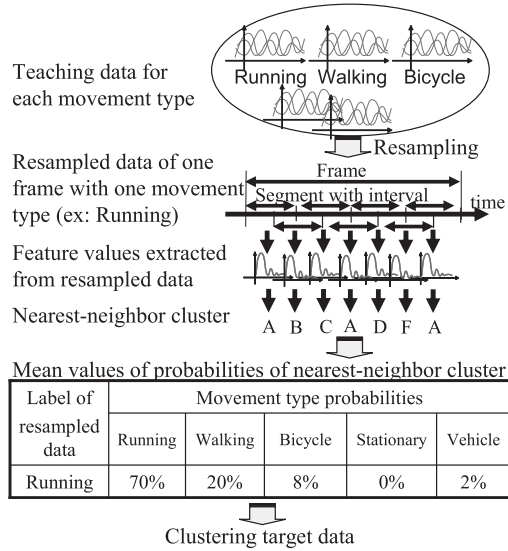
**Fig. 2** Overview of the recognition process.



**Fig. 3** Creation of recognition model using feature values obtained from teaching data.

**Table 2** Clustering parameters for power spectrum learning step.

| Parameter | Value |
|---|---|
| Population (Input data) | Power spectrums for user movement estimation (f dimensional vector) f: Sampling frequency of acceleration sensor |
| Number of clusters | 2 (recursive) |
| Distance function | Sum of squares for the differences between each power spectrum frequency |
| Division Stopping rule | $S_1 > 1/2 * S \quad | \quad N_c < 0.001 * P$ $S_1$ : Largest cluster share , $S$ : The whole one cluster $N_c$ : Number of power spectrums, $P$ : Population |
| Output data for each cluster | - Average power spectrum of all power spectrums in the cluster - Occurrence probability table of user movement in the cluster $P_{ps} : (P_{running}, P_{walking}, P_{bike}, P_{stop}, P_{motors})$ |

by the user movement estimation process to evaluate the reliability. These two models are used to generate key frames. The key frames are used to determine the high-reliability time periods called *key frame periods* from the multiple key frames produced by this process, and the final recognition result is output.

In short, this proposed method consists of a learning step for the power spectrum, a learning step for reliability, an inference step using the power spectrum, a reliability estimation step, and a post-processing step. The sequences for each step are as follows.

Learning step for the power spectrum learns on the movement type estimation model using teaching data. The learning process uses the X-Means method [22] to perform recursive K-means clustering with K=2. The Euclidean distance between feature values is used as the evaluation function for clustering. The reason for using the X-Means method is because using the standard K-Means method would result in a large number of small clusters in cases such as running that produce feature values with a wide variance while failing to separate into clusters for such states that produce feature values with a narrow variance, e.g, stationary and vehicle. Clustering is used in the proposed method because data for the different movement types is skewed in such a way that it produces clusters that are distinctive of the different movement types, so use of the X-Means method is appropriate. The division stopping rule in the X-Means method uses AIC or BIC as an index. However, given the purpose of clustering in the proposed method, the skewing of the distributions for each movement type contained in a cluster is used as the index.

As shown in Fig. 3 and Table 2, the data is first separated into two clusters and then the movement type distribution is obtained for each cluster. Next, if the most frequent movement type makes up more than half of the cluster, the cluster's distribution is deemed to be sufficiently skewed to that movement type and therefore that cluster is not divided further. If the most frequent movement type does not make up more than half of the cluster, clustering is performed again to produce two further clusters and the same evaluation process is repeated. Division also stops regardless of the movement type distribution of the teaching data in the case of small clusters that contain less than a predefined number of teaching data. This process is repeated until there are no

more clusters still to be divided. Next, the probability table containing the movement type distribution for each cluster and the mean value of their feature values are obtained for use as the model for movement type estimation.

The inference step for the power spectrum is based on the cluster matching method. Initially, this method divides sensor data into time section units called frames, and then, divides each frame into time steps of FFT window size and calculates the power spectrums. It then searches for the cluster with the least difference (nearest) by comparing the power spectrum with the average power spectrum of each cluster to obtain $P_{ps}$ for the nearest cluster for each power spectrum. Finally, this method calculates the average $P_{ps}$ for the frame, and selects the user movement with the best probability as the inference result for the frame. This inference result is defined as $R_{ps}$, and the average $P_{ps}$ in a frame is defined as $P_{psavg}$.

The following describes the reliability of the inference result. The learning process also uses the X-Means method. However, the clustering target data for learning

**Fig. 4** Generation of clustering target data by resampling process for reliability estimation model.
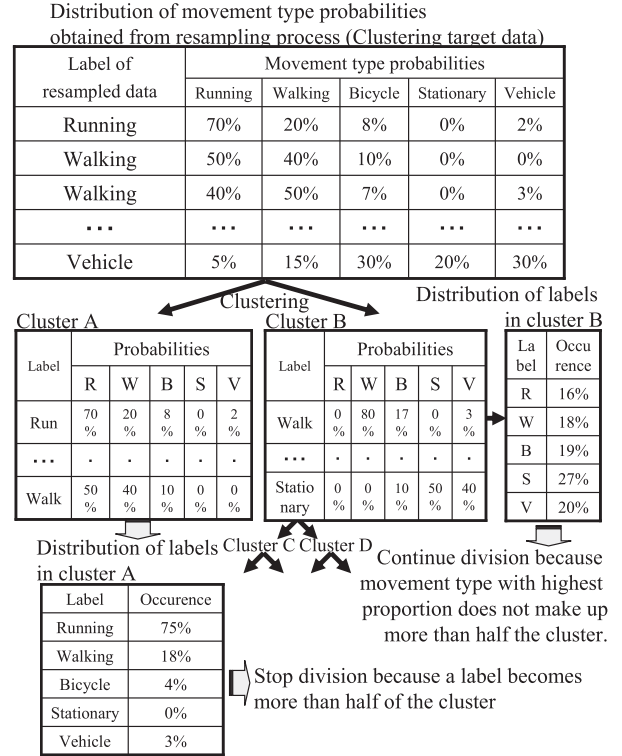


**Fig. 5** Clustering of movement type probabilities for building the reliability estimation model.

**Table 3** Clustering parameters for reliability learning.

| Parameter | Value |
|---|---|
| Population (Input data) | $P_{psavg} : (P_{running}, P_{walking}, P_{bike}, P_{stop}, P_{motors})$ (where the labels indicates the user movement type) |
| Number of clusters | 2 (recursive) |
| Distance function | Sum of squares for the differences between the probability of each user movement type $(P_{running}, P_{walking}, P_{bike}, P_{stop}, P_{motors})$ |
| Division Stopping rule | $S_1 > 1/2 * S \mid N_c < 0.001 * P$ $S_1$ : Largest cluster share , $S$ : The whole one cluster $N_c$ : Number of $P_{psavg}$ , $P$ : Population |
| Output data of each cluster | - Average of all $P_{psavg}$ in the cluster - Occurrence probability table of user movement in the cluster $P_{pt} : (P_{running}, P_{walking}, P_{bike}, P_{stop}, P_{motors})$ |

on this model is produced by resampling process using the teaching data as shown in Fig. 4. In the resampling process, one frame of sensor data with one movement type is resampled randomly from the teaching data. Next, resampled data is split into segments, and the feature value is extracted for each segment. Then, the nearest-neighbor clusters of model for each segment is searched, and calculates the mean values of probabilities of nearest-neighbor cluster. That is, the clustering target data consists of the mean values of probabilities and the label of user movement type collected from all the teaching data from all movement types (all frames), as shown in Fig. 5.

The clustering process uses the Euclidean distance between the mean values of movement type probabilities is used as the evaluation function for clustering as shown in Table 3. The method uses the skew of the distribution of the label of movement type in each cluster as its index. After clustering is complete, the label of movement type with the highest number of occurrences in each cluster is determined and the mean value of movement type probabilities obtained from each cluster is used as the reliability estimation model.

As explained in Sect. 3.2, there is a problem with loss in performance caused by periods of time during which the acceleration data resembles that for other types of movement. When this happens, power spectrum inference produces similar $P_{psavg}$ results for different types of movement. Therefore, the clustering result for the $P_{psavg}$ values for all types of user movement expresses the reliability. For example, if $P_{psavg}$ is (0.7, 0.2, 0.1, 0.0, 0.0), and the $P_{pt}$ of the nearest cluster is (0.99, 0.01, 0.0, 0.0, 0.0), this means that most of the cases of $P_{psavg}$ (0.7, 0.2, 0.1, 0.0, 0.0) correspond to running, so this inference result is reliable. In another example, if $P_{psavg}$ is (0.0, 0.1, 0.2, 0.4, 0.3) and the $P_{pt}$ of the nearest cluster is (0.0, 0.0, 0.1, 0.4, 0.5), this means that $R_{ps}$ is stationary, but most cases of $P_{psavg}$ (0.0, 0.1, 0.2, 0.4, 0.3)

correspond to vehicle travel, so this inference result isn't reliable. This proposed method uses X-means clustering using $P_{psavg}$ as the learning data for the reliability learning step.

The reliability estimation step is based on the cluster matching method. Initially, this method gets the result of the power spectrum inference step ($P_{psavg}$). It then searches for the cluster with the least difference by comparing $P_{psavg}$ with the average probability table for each cluster to find $P_{pt}$ for the nearest cluster. The estimation result is the cluster share for the user movement with the highest $P_{pt}$ probabil-

| Frame No. | Inference result using power spectrum | Estimation result of reliability | Estimation result |
|---|---|---|---|
| 1 | Walking | Reliable | Walking |
| 2 | Walking | Reliable | Walking |
| 3 | Running | Unreliable | Walking |
| 4 | Walking | Reliable | Walking |
| 5 | Running | Unreliable | Walking |
| 6 | Running | Reliable | Running |
| 7 | Running | Unreliable | Running |
| 8 | Running | Reliable | Running |

Key frame = Walking (frames 1–5)
Key frame = Running (frames 6–8)

**Fig. 6**  Post-processing example.



**Fig. 7**  Example microphone power spectrum data.

ity. This is defined as $R_{pt}$. Finally, this method compares $R_{ps}$ with $R_{pt}$. If $R_{ps}$ equals $R_{pt}$, the inference result $R_{ps}$ is a reliable. If not, $R_{ps}$ isn't reliable. In this method, a reliable frame is called a candidate key frame.

The following describes the post-processing step. As described in the previous paragraph, the result of the previous steps indicates whether each frame is a reliable or unreliable frame, as in the example in Fig. 6. In the post-processing step, the method estimates the key frame using the following generation rule.

-Two consecutive candidate key frames have the same estimation result for user movement.

-Some unreliable frames exist between candidate key frames (there are no candidate key frames with a different status).

-The key frame duration must be longer than the threshold time for each type of user movement (For example, the thresholds for running and walking are 20 seconds).

In the example in Fig. 6, frames No.1 and No.2 are walking candidate key frames indicating that the key frame is "walking" and therefore the estimation result is "walking". Next, the inference result for frame No.3 and No.5 is running, but these are unreliable frames, and therefore the estimation result remains "walking". At frame No.6, the candidate key frame changes to running, and the "running" result of frame No.8 leaves this unchanged. Although this post-processing rule improves the accuracy of user movement estimation by only using reliable frames, it means that the latency when user movement changes becomes longer as explained in Sect. 6.

### 4.3 Probabilistic Inference Using the Microphone Data

If the result of the probabilistic inference described above indicates travel in a motor vehicle, the next step is to perform probabilistic inference using the microphone data. This method also includes a power spectrum learning step, reliability learning step, power spectrum inference step, reliability estimation step, and the post-processing step. The sequence of each step is as follows.

Although power spectrum learning for the microphone data uses the same method as the acceleration data, the method used to convert the raw data to an input vector is different. This method first calculates the power spectrum from the recorded audio data and then calculates the average am-
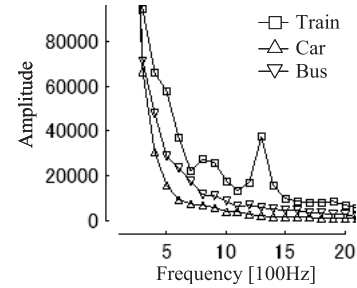
plitude for each 100 Hz band between 100 Hz and 1500 Hz. This produces input data in the form of a 14-dimensional vector. The other parameters are the same as described in the previous section. The method uses the microphone to detect the ambient sounds characteristic of different types of vehicle. Distinctive ambient sounds include the sound of the electric motor or metallic sounds when riding in an electric train and the engine sound when riding in a bus. As shown in Fig. 7, the frequency range in which the three different states can be distinguished is 100 to 1500 Hz. The ambient sound of a car in particular is notably quieter than the other two states. Accordingly, the proposed method uses the power spectrum from the microphone data, and the feature value consists of the mean amplitude for each 100 Hz band in 100 to 1500 Hz range of interest.

This method seeks to improve estimation accuracy by using microphone data to identify characteristic differences present when riding in a car.

The power spectrum inference step, reliability learning step, reliability inference step, and post-processing step are the same as those used for acceleration data as described in the previous section.

### 4.4 Probabilistic Inference Using the GPS Positioning Data

If the result of the probabilistic inference described above indicates travel by train or bus, the next step is to perform probabilistic inference using GPS positioning data. Although both the learning and estimation methods are the same as described in the previous section, the method used to convert the data to input data is different. First, regular GPS positioning measurements are made and the average speed calculated from pairs of consecutive measurements. The resulting input data consists of a set of unsigned integer values. The average speed characteristics of trains and buses are different, as can be seen in the example in Fig. 8.

This method seeks to improve estimation accuracy by using GPS positioning data to identify characteristic differences between travel by train or bus.

### 4.5 Sleep Mode

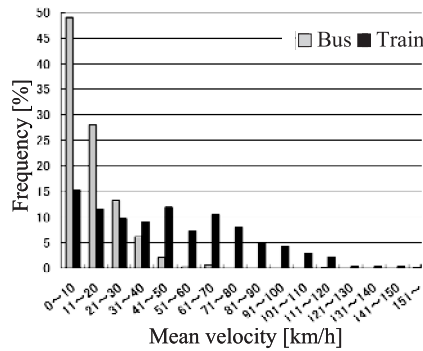If the result of the above user movement estimation indicates that the user is stationary, or travelling by car, train,

**Fig. 8**　Comparison of bus and train average speed.



**Fig. 9**　Evaluation device (au W62CA [23]).

**Table 4**　Specification of evaluation device.

| Parameter | Specification |
|---|---|
| Processor | MSM7500 (ARM9 400MHz, cdma2000) |
| Battery | 850mAh |
| Accelerometer | 3 axis, 16Hz, with a pedometer wired logic |
| Microphone | 8000Hz, 16bit PCM |
| GPS | about 1Hz (Network assisted measurement) |
| Runtime | BREW 4.0.1Ja |

**Table 5**　Operating parameters for the estimation method.

| Parameter | Value |
|---|---|
| FFT window size (Acceleration) | 32 samples (2sec) |
| Sliding size of FFT window (Acceleration) | 16 samples (1sec) |
| Frame length (Acceleration) | 10sec |
| FFT window size (Microphone) | 250msec |
| Sliding size of FFT window (Microphone) | 250msec |
| Frame length (Microphone) | 5sec |
| GPS measurement interval | 5sec |
| Frame length (GPS) | 60sec |
| Polling interval (sleep mode) | 30sec |
| Threshold of time length key frame period (Run, Walk) | 20sec |
| Threshold of time length key frame period (Bike) | 60sec |
| Threshold of time length key frame period (Stationary, Motors, Car, Bus, Train) | 120sec |

or bus, this method changes to sleep mode. Commercial mobile phones usually include a pedometer implemented by wired logic, and also have a function to obtain the location of the nearest CDMA base station. As the load for these processes is much lower than that for the procedures described in the preceding sections, the proposed method includes use of sleep modes which depends on the type of user movement. This mode is provided to reduce power consumption and involves checking the hard-wired pedometer and the base station latitude and longitude for changes. Although this mode is unable to recognize the seven different movement types, it can determine whether a change has occurred in the already established movement type.

If the movement estimation result indicates that the user is stationary, then neither the pedometer value nor the location of nearest CDMA base station can be expected to change. Accordingly, the sleep mode for a stationary user performs a polling check on the pedometer value and the base station location. During this time, use of the sensors (acceleration, microphone, GPS) and the methods described in the preceding sections is suspended. If any change in the pedometer or base station indicates a potential change in state and triggers waking up from sleep mode. When the movement estimation result indicates that the user is riding in a car, train, or bus, then the pedometer value can be expected to remain unchanged. Accordingly, waking up from sleep mode occurs if there is a change in the pedometer or the base station latitude and longitude remain unchanged for longer than a fixed time period. When the system resumes, it is assumed that the next movement type is walking or running, so the system goes to the acceleration mode as shown in Fig. 1.

This method seeks to improve power consumption by using the pedometer and the base station location.

## 5. Performance Evaluation and Results

An evaluation system was implemented to verify the effectiveness of the proposed method.

### 5.1　Evaluation System Implementation

Figure 9 shows a photograph of the device used for evalu-

ation and Table 4 lists the specifications of the device. As the device is a commercial mobile phone that already contains an accelerometer, microphone, and GPS, this indicates that the sensor configuration used by the proposed method is practical. The estimation software was written in C and runs on the above device. Table 5 lists the operating parameters for the estimation methods used by the system. For the purpose of this evaluation, the estimation accuracy is indicated by the precision, recall, and F-measure.

### 5.2　Evaluation Data Collection

The experimental data was obtained from 214 experimental subjects, covered seven types of movement (running, walking, bicycle, stationary, car, train, bus), and used 3 different device locations (held in the hand, trouser pocket, and inside a bag) as shown in Fig. 10.

Three impractical cases including holding the device in hand while riding a bicycle were not covered. Approximately 1000 hours of acceleration data, microphone data, and GPS data were collected covering the above combinations of cases, as shown in Table 6.

**Fig. 10**    Device location examples.

**Table 6**    Experimental data combinations.

| Parameter | Value |
|---|---|
| Experimental subjects | 214 people, approximately 1000 hours |
| Age of experimental subjects | 18 – 60 years old |
| Device locations | Held in the hand / trouser pocket / inside a bag |
| Running situations | Outdoor, flat road, jogging |
| Walking situations | Outdoor/indoor, flat road / upstairs / downstairs |
| Bike riding situations | Outdoor, flat road, cycling |
| Stationary situations | Indoor, standing / sitting |
| Car travel situations | Outdoor, taxi / rental car (4 types of car) |
| Train travel situations | 5 railway routes / 2 subway routes |
| Bus traveling situations | 2 short distance bus routes |

**Table 7**    Estimation accuracy of proposed method and previous method for five types of user movement.

| Movement type | Proposed Method | | | Previous Method | | |
|---|---|---|---|---|---|---|
| | Precision | Recall | F measure | Precision | Recall | F measure |
| Running | 0.998 | 0.949 | 0.973 | 0.997 | 0.956 | 0.976 |
| Walking | 0.921 | 0.985 | 0.952 | 0.926 | 0.939 | 0.933 |
| Bike | 0.961 | 0.970 | 0.965 | 0.891 | 0.952 | 0.920 |
| Stationary | 0.966 | 0.939 | 0.952 | 0.712 | 0.769 | 0.739 |
| Motors | 0.943 | 0.940 | 0.942 | 0.732 | 0.642 | 0.684 |
| Average | 0.958 | 0.957 | 0.957 | 0.852 | 0.852 | 0.851 |

This experimental data is divided into 3 groups (each group has the data from 71 experimental subjects). One group of data was used for learning and the other two groups of data were used for estimation, and this performance test operated using each group data for learning. The performance values were calculated from the averages of these test results (six sets).

## 5.3    Result

Table 7 shows the performance test results for the estimation accuracy of the proposed method and a previous method [18]. Here, the "proposed method" means the accelerometer-based probabilistic inference described in Sect 4.2. Table 7 shows that the inference accuracy for five types of user movement using the proposed method is around 95% or better. This is significantly better than the estimation performance of the previous method described in Sect. 3.2.

Table 8 shows a comparison of the estimation accuracy of the proposed method using accelerometer, microphone, and GPS and a subset of the proposed method that uses accelerometer data only (as described in Sect. 4.2) for seven types of user movement. The full method improves the av-

**Table 8**    Estimation accuracy of proposed method and proposed method using accelerometer only for seven types of user movement.

| Movement type | Proposed Method | | | Accelerometer Only | | |
|---|---|---|---|---|---|---|
| | Precision | Recall | F measure | Precision | Recall | F measure |
| Running | 0.998 | 0.949 | 0.973 | 0.998 | 0.949 | 0.973 |
| Walking | 0.921 | 0.985 | 0.952 | 0.921 | 0.985 | 0.952 |
| Bike | 0.961 | 0.970 | 0.965 | 0.961 | 0.970 | 0.965 |
| Stationary | 0.966 | 0.939 | 0.952 | 0.966 | 0.939 | 0.952 |
| Car | 0.974 | 0.831 | 0.897 | 0.947 | 0.772 | 0.851 |
| Bus | 0.850 | 0.948 | 0.897 | 0.787 | 0.989 | 0.876 |
| Train | 0.941 | 0.833 | 0.884 | 0.840 | 0.779 | 0.808 |
| Average | 0.944 | 0.922 | 0.931 | 0.917 | 0.912 | 0.911 |

**Table 9**    Power consumption during performance testing for each estimation step.

| Estimation mode | Power consumption | Continuation operation time |
|---|---|---|
| Probabilistic inference using acceleration data | 38 mA | 22 hours |
| Probabilistic inference using microphone data | 70 mA | 12 hours |
| Probabilistic inference using GPS data | 130 mA | 6.5 hours |
| Sleep mode | 2.2 mA | 390 hours |
| Proposed method (office worker example) | 7.2 mA | 118 hours |

**Table 10**    Average of latency for proposed method.

| Movement | Run | Walk | Bike | Stationary | Motors | Car | Bus | Train |
|---|---|---|---|---|---|---|---|---|
| latency[sec] | 23.0 | 23.2 | 63.3 | 182.9 | 252.9 | 396.3 | 828.6 | 790.2 |

erage estimation accuracy for the seven types of user movement with 93.1%.

Table 9 shows the power consumption recording during performance testing for the proposed method using the experimental device as shown in Fig. 9. The power consumption during sleep mode is clearly much lower than in other modes and the proposed method achieves over 100 hours of continuous operation on a commercial mobile phone in the office worker examples (the average of eight office workers).

Table 10 shows the average of latency for user movement estimation. The latency for running and walking is shorter than for other types of user movement, and the latency depends on the threshold time used in the key frame generation rule in the post-processing.

## 6.    Discussion

As described in the previous section, the probabilistic inference method using acceleration data was able to recognize running, walking, bike riding, stationary, and travel by motor vehicle with about 95% accuracy as shown in Table 7. In particular, use of reliability estimation and the key frame period generation rule achieved significantly improved performance. In short, from the point of view of accuracy, the user movement estimation method does not need to use all of the time sections, and better results can be achieved by us-
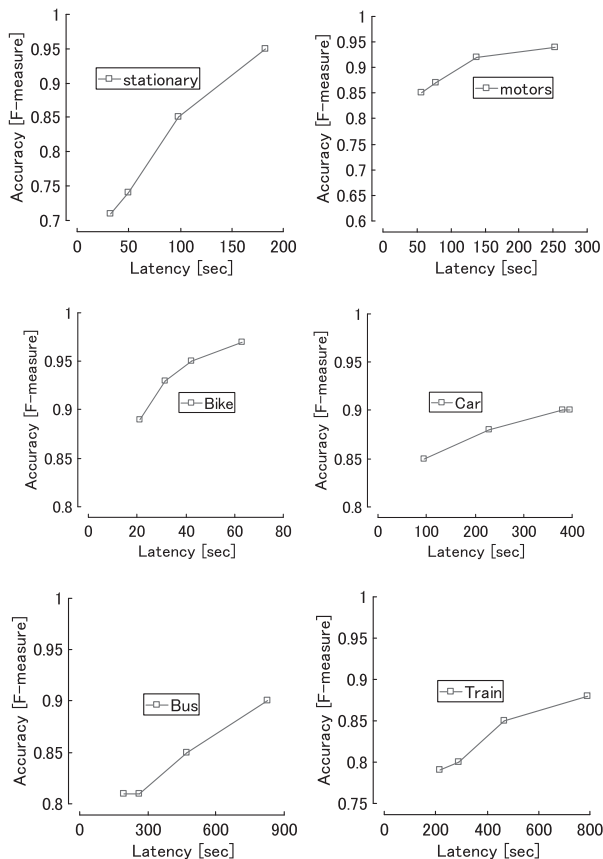
**Fig. 11** Trade-off between the accuracy and the latency.

ing characteristic time sections only. This proposed method represents one approach to the identification of characteristic time sections using reliability estimation.

Table 8 shows that microphone data is effective for identifying when the user is traveling by car and or by train or bus, and GPS positioning data is effective for identifying when the user is traveling by train or bus. The result in Table 8 shows that the proposed method achieves 90% or better accuracy for all seven of the different types of user movement. Table 9 shows that the power consumption for estimating using microphone or GPS data is higher than for the accelerometer and Table 8 shows that probabilistic inference using accelerometer data only achieves accuracy of 80% or better. Therefore, the best estimation mode depends on the user's circumstances. This indicates that each service and user should be able to select the estimation mode.

Table 9 shows the power consumption in each mode and how sleep mode contributes greatly to longer battery life. Of course, battery life depends on the user's activity. For example, the worst case is for a user running a marathon or perticipating in a bicycle race because in these cases the proposed method can never change to sleep mode. Regardless of this, 22 hours of battery life is good enough for these usage cases. On the other hand, the proposed method achieves a battery life of over 2 weeks for a mobile phone left on the user's desk. This level of power consumption is

similar to that of a mobile phone in standby mode. When we consider use cases, the ability of the method to run in the background is a very important point.

The key frame generation rule in the proposed method uses different threshold times for different types of movement as shown in Table 5. The average latency in Table 10 becomes shorter as the movement state getting easy to be identified, e.g. running and walking. Conversely, the latency for car, bus, and train are longer. In this method, one key frame period needs at least two key frames. This evaluation uses a frame length of 10 sec as shown in Table 5, so the latency of all movement types is longer than 20 sec. Moreover, it should be noted that the latency of the proposed method depends on the observed time of a key frame that contains distinctive sensor data for a particular movement type. The improvement of latency is left for further study.

Figure 11 shows a trade-off between latency and estimation accuracy for each user movement type. This means that appropriate parameters of this method should be defined consider the service requirements.

Moreover, because the current proposed method focuses on periodic characteristic patterns, we believe there is a potential for use in identifying other types of movement with periodic action. On the other hand, it is difficult to identify types of movement with the aperiodic action. This is a subject for future work.

## 7. Conclusion

This paper presents a method for identifying the movement of a user using a system that combines an accelerometer, microphone and GPS and is implemented in a mobile phone. We found that the proposed method can improve estimation accuracy by considering the reliability estimation, and the method can improve power consumption through use of sleep mode. Performance testing showed that the proposed method can identify seven different types of movement with accuracy of around 90% or better, and the power saving functionality can extend the battery life of a mobile phone to over 100 hours while our estimation algorithm is running in the background.

## Acknowledgments

**References**

[1] N. Kern, "A model for human interruptability: Experimental evaluation and automatic estimation from wearable sensors," ISWC04, pp.158–165, 2004.

[2] S. Instille, "Acquiring in situ training data for context aware ubiquitous computing applications," CHI 2004, pp.1–9, 2004.

[3] L. Bao and S. Intille, "Activity recognition from user-annotated acceleration data," Pervasive 2004, LNCS, vol.3001, pp.1–7, Springer, A. Heidelberg, 2004.
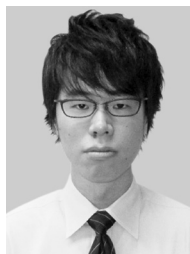
[4] D. Siewiorek, SenSay, "A context-aware mobile phone," ISWC'03, 2003.

[5] P. Lukowicz, H. Junker, and M. Troster, "WearNET: A distributed multi-sensor system. for context aware wearables," UbiComp2002, pp.361–370, 2002.

[6] S. Lee and M. Kenji, "Activity and location recognition using wearable sensors," Pervasive2002, pp.10–18, Springer, A. Heidelberg, 2002.

[7] T. Choudhury, L. Anthony, L. Louis, R. Ali, R. Adam, B. Gaetano, H. Bruce, K. Karl, A.L. James, L. Jonathan, W. Danny, and H. Dirk, "The mobile sensing platform: An embedded activity recognition system," IEEE Pervasive Computing, vol.7, no.2, pp.32–41, 2008.

[8] J. Yin, Q. Yang, and J.J. Pan, "Sensor-based abnormal human-activity detectio," IEEE Trans. Knowl. Data Eng., vol.20, no.8, pp.1082–1090, 2009.

[9] N. Györbíró, Á. Fábián, and G. Hományi, "An activity recognition system for mobile phones," Mob. Netw. Appl., vol.14, no.1, pp.82–91, 2009.

[10] P. Zappi, C. Lombriser, T. Stiefmeier, E. Farella, D. Roggen, L. Benini, and G. Tröster, "Activity recognition from on-body sensors: Accuracy-power trade-off by dynamic sensor selection," EWSN 2008, pp.17–33, 2008.

[11] T. Iso and K. Yamazaki, "Gait analyzer based on a cell phone with a single three-axis accelerometer," ACM 8th Conference on Human-Computer Interaction with Mobile Devices and Services (Mobile-HCI2006), pp.141–144, 2006.

[12] Y. Kawahara, T. Hayashi, H. Tamura, H. Morikawa, and T. Aoyama, "A context-aware content delivery service using off-the-shelf sensors," 2nd International Conference on Mobile Systems, Applications, and Services, 2004.

[13] J. Yang, "Toward physical activity diary: Motion recognition using simple acceleration features with mobile phones," 1st International Workshop on Interactive Multimedia for Consumer Electronics (IMCE), ACM Multimedia 2009, 2009.

[14] E. Miluzzo, N.D. Lane, K. Fodor, R. Peterson, H. Lu, M. Musolesi, S.B. Eisenman, X. Zheng, and A.T. Campbell, "Sensing meets mobile social networks: The design, implementation and evaluation of the CenceMe application," 6th ACM Conference on Embedded Networked Sensor Systems, pp.337–350, 2008.

[15] T. Brezmes, J.L. Gorricho, and Cotrina, "Activity recognition from accelerometer data on mobile phones," Proc. 10th International Work-Conference on Artificial Neural Networks, IWANN2009, pp.796–799, 2009.

[16] I. Anderson, J. Maitland, S. Sherwood, L. Barkhuus, M. Chalmers, M. Hall, B. Brown, and H. Muller, "Shakra: Tracking and sharing daily activity levels with unaugmented mobile phones," Mobile Networks and Applications., vol.12, pp.185–199, 2007.

[17] J.R. Kwapisz, G.M. Weiss, and S.A. Moore, "Activity recognition using cell phone accelerometers," SensorKDD, 2010.

[18] A. Kobayashi, T. Iwamoto, and S. Nishiyama, "UME: Method for estimating user movement using an acceleration sensor," International Symposium on Applications and the Internet, pp.169–172, 2008.

[19] M. Kourogi and T. Kurata, "Personal positioning based on walking locomotion analysis with self-contained sensors and a wearable camera.," Proc. ISMAR03, pp.103–112, 2003.

[20] S. Thrun, D. Fox, and W. Burgard, "A probabilistic approach to concurrent mapping and localization for mobile robots," Autonomous Robots, vol.5, pp.253–271, 1998.

[21] H. Aoki, B. Schiele, and A. Pentland, "Realtime personal positioning system for wearable computers," International Symposium on Wearable Computers (ISWC'99), 1999.

[22] D. Pelleg and A.W. Moore, "X-means: Extending K-means with efficient estimation of the number of clusters.," 17th International Conference on Machine Learning, pp.727–734, 2000.

[23] au W62CA Information, http://www.au.kddi.com/seihin/ichiran/cdma1x_win/w62ca/

**Arei Kobayashi** received B.E. and M.E., degrees in Electrical Engineering from Hokkaido University, Hokkaido, Japan, in 1996 and 1998. He joined KDD R&D Laboratories Inc. (currently KDDI R&D Laboratories Inc.) in 1998. His research interests include mobile computing, sensor data mining, mobile AR, CGM based market economy, Terrestrial digital broadcasting, XML data compression, and SVG Mobile. He received the Good Paper Award of MBL in 1999, the Best Paper Award of FIT2003, and the ARIB Electric Wave Achievement Award in 2008.

**Shigeki Muramatsu** received his bachelor's degree in Electrical Engineering from The University of Tokyo, Tokyo, Japan, in 1997 and master's degree in Information and Communication Engineering from The University of Tokyo, Tokyo, Japan, in 1999, respectively. He joined KDD R&D Laboratories Inc.(currently KDDI R&D Laboratories Inc.) in 1999 and engaged in research on information retrieval, web mapping, vector graphics and ITS. His current research interests include positioning and sensor data mining.

**Daisuke Kamisaka** received his M.E. degree in System Engineering from Wakayama University, Wakayama, Japan in 2005. He joined KDDI R&D Laboratories, Inc. in 2005 and has engaged in research on context awareness. His current research interests include location and activity recognition using sensors and mobile phones. He received the IPSJ's Young Researchers' Award in 2003 and 2008, the IPSJ's Special Session Presentation Award in 2008, and the Best Paper Award of the International Symposium on Ubiquitous Computing Systems (UCS'09) in 2009

**Takafumi Watanabe** received the his bachelor's degree in Informaiton and Computer Science from The University of Osaka, Osaka, Japan, in 2006 and the master's degree in Information Science and Technology from The University of Osaka, Osaka, Japan,in 2008,respectively. He joined KDDI R&D Laboratories Inc. in 2008. His research interests include activity recognition and location using sensors and mobile phones.

**Atsunori Minamikawa** received his B.S. degree in Electrical Engineering from Kyoto University, Kyoto, Japan in 2003 and M.S. degree in Communications and Computer Engineering from Kyoto University, Kyoto, Japan in 2005. He joined KDDI R&D Laboratories, Inc. in 2005 and has engaged in research on ubiquitous application and data mining. His current research interests include lifelog mining and linguistic psychology. He received the IEICE's Young Researchers' Award in 2010.

**Takeshi Iwamoto** received his Bachelors degree in Environmental Information at Keio University in 1998, and his Masters degree and Ph.D. degree in Media and Governance at Graduate School of Keio University in 2000 and 2005. He joined the Faculty of Engineering at Toyama Prefectural University in 2009. His research interests include middleware for ubiquitous application, indoor location estimation and application framework.

**Hiroyuki Yokoyama** received B.E., M.E., and Ph.D. degrees in Electrical Engineering from Kyoto University, Kyoto, Japan, in 1992, 1994, and 2006, respectively. He joined the Research and Development Laboratories of Kokusai Denshin Denwa Company Ltd. (currently KDDI Corp.) in 1994 and has engaged in research on communications network planning. His current research interests include sensor data mining and its applications for mobile phones. He received the IEICE's Young Researchers' Award in 2000.