Electronic Preprint for Journal of Information Processing Vol.20 No.1

Regular Paper

A Large Scale Gathering System for Activity Data using Mobile Devices

Yuichi Hattori[1] Sozo Inoue[1,a)]

Received: April 11, 2011, Accepted: September 12, 2011

Abstract: In this paper, we introduce a large-scale activity gathering system with mobile sensor devices such as smart phones and accelerometers. We gathered over 35,000 activity data from more than 200 people over approximately 13 months. We describe the design rationale of the system, analyze the gathered data through statistics and clustering, and application of an existing activity recognition method. From the recognition, the performance of existing algorithm drastically deteriorated using the gathered data as training data. These results show that we were still able to find a challenging field for activity recognition in larger-scale activity data.

Keywords: activity recognition, mobile device, 3-axis accelerometer

1. Introduction

If human activity could be objectively quantified then we could apply this knowledge to a wide range of applications. For example, people's lifestyles could be quantified and this knowledge utilized to prevent lifestyle-related diseases.

In this research, we aim to gather open data sets for evaluating activity recognition methods. Activity recognition methods have been proposed actively in the literature, and are still being improved, but there are few benchmarking data sets. For this, we developed a large-scale activity data gathering system named ALKAN. ALKAN is a server-client system that gathers a large number of missions with mobile sensor devices, such as smart phones with accelerometers. It enables a simple way of recording and semi-automatic way of uploading activities.

In this paper, we introduce a large-scale activity gathering system with mobile sensor devices such as smart phones and accelerometers. We gathered over 35,000 activity data from more than 200 people over approximately 13 months. We describe the design rationale of the system, analyze the gathered data through statistics and clustering, and application of an existing activity recognition method. From the recognition, the performance of existing algorithm drastically deteriorated using the gathered data as training data. This results show that we found still a challenging field for activity recognition in larger-scale activity data. The system requirements are as follows:

(1) Large-scale: activity data varies widely among users. Moreover, there are a lot of activity types in real human life. Therefore, we need a lot of activity data for using them for training activity recognition algorithms.
(2) Open: many existing work use small-scale data such as from 1 to 20 people. If the collected activity data are available to researchers then they can proceed with the research based on the open large-scale data. However, currently there is no such open large-scale activity data in the world, each researcher has to evaluate the proposed method using their own small-scale data.
(3) Daily life: many of the existing work use activity data obtained in laboratory settings. However, this is unnatural, since human activities are not always performed in laboratories. Therefore, existing activity recognition algorithms cannot be only applied to real daily life of users.

This paper is organized as follows. Related work is described in Section 2. The ALKAN system is described in Section 3. The gathered activity data are overviewed in Section 4. Section 5 shows the nature of gathered data by applying an existing activity recognition method. Section 6 gives the summary and discussion for future work.

2. Related Work

In the literature, a lot of work has tried to recognize activities with sensor devices. Chambers et al. [1] tried to distinguish activity and movement of arm using two sensors on the arm. Laerhoven et al. [6] tried to distinguish activities, postures, and riding bicycle, using two sensors at waist. These studies however target only a single user. They don't evaluate for varieties of people like this paper.

Lee et al. [4] recognize the types and the strength of movement with eight users using several sensors at waist and thighs. Mantyjarvi et al. [5] use 6 sensors on the waist, and recognized activities and postures with six users. Laerhoven et al. [6] attach 2 sensors on the back thigh, and 7 activities, postures, and bicycles are recognized with 10 users. Herren et al. [2] uses 2 sensors on the back and on feet, and recognizes angles and walking speed with 20 users.

1 Graduate School of Engineering, Kyushu Institute of Technology, Kitakyushu, Fukuoka 804–8550, Japan
a) sozo@mns.kyutech.ac.jp

However, these studies acquire activity data in semi-artificial environments, and the users moved by the instruction. The data was not obtained from actual daily life.

As the researches which aim at activity recognition in daily life, the following work are presented. Uiterwaal et al. [8] used two sensors at waist and one sensor at thigh, and measured movements and postures in working environments. Kern et al. [3] used 36 sensors at each joint, and measured movements such as typing, chair, handshake, and writing on a blackboard. However, these two researches used only one user. Uiterwaal et al. [8] utilized an accelerometer on the wrist to record the activity of six types of activities on ten users on daily life. The result showed the possibility to distinguish activities in multiple users.

In contrast to these studies, our work collects not the small scale and artificial data, but large-scale activity data of real life. Moreover, we assume only one sensor per user due to requirements of usability and feasibility.

Bao et al. [7] discuss how to learn activity recognition from annotated data by users. They explained the procedures and examples of each activity to users in advance, and eliminated the variance of annotation. Upon which, they collected the data of 5 sensors on the body, and obtained 84% of accuracy. They also attached 2 sensors on upper and lower body each keeps accuracy well as 5 sensors.

Our study also uses parts of the same feature vectors as Bao et al. [7], but we use single sensor, and also focus on the system to enable gathering activity data.

Berchtold et al. [11] propose an activity recognition service with mobile phones and achieve 97% accuracy at best for 20 subjects. While our system is similar to this work, our system focuses on gathering open data for activities with accurate labels with low stress. Moreover, our work shows the result from gathering massive data.

Kawaguchi et al. [10] proposed a promotion to gather open activity data from multiple laboratories, and has 6,700 accelerometer data from 540 subjects in total. While their work is not a system proposal, our approach is to provide a platform system to gather activity data anytime and anywhere.

3. ALKAN System

To collect activity data efficiently, we developed a large-scale activity gathering system named ALKAN. In this section, we describe the requirement analysis and the system design.

3.1 Requirements

For collecting large-scale activity data, the following requirements are addressed.

(1) (Accuracy) Labels, such as activity classes and the position of a device on the body, are accurately added.
(2) (Usability) Users can record activity data anytime and anywhere with minimum stress.
(3) (Motivation) Users need some type of motivation to promote recording activity data.
(4) (Flexibility) Labels must be extended if a new activity class or new position is discovered. Also, the utilization or application of the activity classes might be changed accordingly.
(5) (Scalability) The system can record the data from multiple users.

To address requirement (1), the most accurate way of labeling is *managed labeling:* to predefine labels, and designate users to repeat activities under the labels for designated times in managed environment such as in a laboratory. However, this method is too costly when collecting data on a large scale from thousands of samples from over hundreds of people. It is not only costly for experimenter, but also costly for participant (users) to be forced to perform designated activities. Moreover, it is almost impossible to cover any kinds of activity classes in laboratory situations, such as sitting.

Another option of labeling is *unmanaged labeling:* labeling that asks users to label their activity of their real life log, once per certain period of hours/days. Although this is easier, since users can record their activities in instant ways in their life, and the experimenter can obtain ‘real’ activities without preparing artificial environments, the labels might become contaminated in the sense that they are mixed with others, polysemous, insufficient, omitted, and/or mistimed. Therefore, compromised way of managed/unmanaged labeling is required.

To compromise them, we introduce the idea of a “mission.” A *mission* is a sequence of choosing an activity class, choosing the position on the body, and performing the activity. Using this method, users can record activities anytime they want, and the label is accurately stamped within a few second. This method is not suitable for the sequence multiple activities, it can be effective for single activity recordings as first-level activity recognition.

For requirement (2), users must be able to record activities without network connectivity. In spite of the fact that we live in the era of pervasive network access, there are still a lot of environments without network connection, such as in subway. Not only that, but also low latency in usage is important for usability. If a user has to wait for the response from the server before/after an activity, it will be an obstacle for her/him to record. Activity recording should be independent of network connection and latency. The data collected can be uploaded to the server spare time after a while between recordings.

For usability, we adopted smart phones as mobile sensor devices. Standard smart phones were equipped with 3-axis accelerometers, non-volatile storage, and wireless communication. By this, an activity can be performed anytime, and stored. The data can then be uploaded to the server when it is connected to the network.

For requirement (3), since the users are human, some means to prevent boring them was required. A simple way is to give feedback by e-mail. It will become more interactive if the feedback is done within the sensor device. The contents for feedback can be created in variety of ways, such as the summary of their own log, statistic data of the whole data, and ranking of each/whole activity classes. Furthermore, domain specific applications can be provided data results, such as calorie consumption estimation, training logs, practicing sports or dances. If these applications are provided, they also motivate users to record activity data.

In ALKAN, we prepared several feedback services of 1) ranking of activity execution and 2) calendar of activity history, at the first stage. Although they can be extended as mentioned below, these can help to motivate users.

To address requirement (4), we provide feedback services that are dynamically updated through web browser interfaces on mobile devices. As for scalability of the network, we tested concurrent data transmissions to send activity information from 50 iPodTouches to the server through the internet, and confirmed there are no network traffic overflows. Moreover, we do not assume that users will upload the activity data continuously in the real time, but periodically as a batch processing. It means that the timing of uploading will be distributed among users. Therefore, we can assume that the network can handle more than 50 clients for the ALKAN system. For requirement (5), smart phone client software is easy to scale up by installing client software through application deploying services. On the other hand, the server can be scaled up by existing distributed web technology.

3.2 Mission

A *Mission* is a sequence of three operations: 1) a user selects an activity class and a position of the device on the body; 2) starts the activity and the corresponding three axis acceleration data are recorded; 3) inputs additional information as a text comment.

3.3 System Architecture

The ALKAN system consists of A) mobile device clients and B) a server which gathers activity data. A user records missions using the mobile device client. The information is uploaded to the server when it is online and accumulated in the server database. The user can view statistical information of the uploaded data, such as a calendar of activity history and rankings, by connecting to the web server through the mobile device or another web browser on a PC.

3.3.1 Client

We developed client software both for iOS and Android OS. In this paper, we show the views on iOS, which runs on iPhones or iPodTouches by Apple, Inc. in **Fig. 1** and **Fig. 2**.

The client software has the following functionalities:

(1) Mission execution
(2) View and send mission history
(3) View statistical information of the server

In (1), users first select an activity class as in Fig. 1 (a) and a position as in Fig. 1 (b). Then they start the activity and finish it. The sensor can record GPS information and three axis accelerometer data at 20 Hz.

In (2), users can view the recorded mission history and add a comment to each mission as an annotation. Users can also delete missions if he or she does not wish to upload to the server. The mission data can be sent to the server as activity data either by each mission or all at once. After the mission data is sent, it is removed from the history.

In (3), the software shows a web browser access to the server, and shows statistical information such as ranking (Fig. 2 (a)) and calendar history (Fig. 2 (b)). This architecture of web browser interface is suitable not only when we update the statistic information, but also when we serve new information or even when we add a service to specific a user group.

Fig. 1 Mission views in ALKAN: (a) select activity class, (b) select device position and start sensing. (c) start activity, and (d) finish activity.

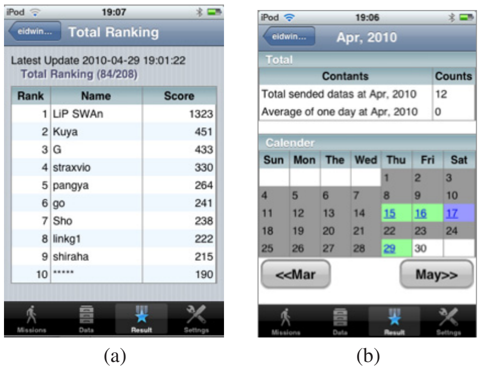

Fig. 2 Statistical information Viewed in a web browser in ALKAN: (a) ranking of the number of activities, and (b) calendar of activity history.

3.3.2 Server

The server gathers the activity data sent from clients, stores it to the database, and calculates and provides statistical information as a web server.

An example of current statistical information is shown in the ranking of the number of executed missions among users. The rankings are divided into the total ranking and those for each activity class. The ranking are supposed to motivate users to perform missions. Moreover, the total ranking can be weighted by activity class to promote collecting fewer activities.

Other statistical information is the history of executed missions for each user. Users can view the start/end date/time, activity class, positions, and GPS information linked from a calendar format. This information is similar to lifestyle-related service, such as managing the lifestyle-related diseases, in which users record their own lifestyles. Moreover, in the near future this will be provided with an automatic lifestyle recognition algorithm when in the near future.

3.3.3 Data Structure

The communication between a client and the server is done over HTTP. Upon connection, the client is authenticated by a user account, and XML-formatted data or CSV-formatted data are transferred between the client and the server.

The transferred data consists of 3 types: 1) Mission information, 2) Position information, and 3) Activity data set.

1) Mission list is a list of activity classes. 2) Position list is a list of device positions on the body. 1) and 2) are represented in XML format, and are provided by the server to each client. This makes updating the candidates of activity classes and positions dynamically in operation. To enable it, 1) and 2) also have version information to be able to intermingle multiple versions into the field.

Finally, 3) Activity data set is a set of mission executions, sent from each client to the server. When we call *an activity data*, we mean that it corresponds to a single mission execution and *a activity data set*, it corresponds to one or more mission executions. An activity data includes the device product information, the user ID, the activity class ID, the position, and the sensor data with time stamps. They are mainly represented in XML format for extensibility, but the sensor data is in CSV format for efficiency of data size.

The activity class ID is one of the 1) Mission list. Moreover, the position can be either represented by position ID listed in 2) Position list, or by XYZ coordinates in the body image in Fig. 1 (b).

Sensor data is in CSV format, and currently contains the data from the three-axis accelerometer and GPS coordinates, but it can be easily extended by adding columns.

4. Collected Activity Data

ALKAN offers an opportunity to obtain large-scale activity data, which contributes to researches of not only activity recognition, but also context awareness, and various of social sciences. As far as we know, such large scale of open activity data does not exist other than our work. By viewing and analyzing how the data are gathered, we can expect the knowledge attaining such large-scale activity data gathering. In this section, we investigate the property of gathered activity data.

Since December 3, 2009, we have delivered 216 iPod touches, as mobile sensors, to university students and staff. So far, 216 devices were delivered by the January 21st, 2011.

We asked users a favor to collect activity data once a day on average. The activity data recorded by a device can be uploaded at any time when it is online to the Internet. When there is no Internet connection, the data is accumulated to the mobile device, and uploaded when it is connected. We are asking the user to perform missions at least once a day. But we are not forcing them to do so. Therefore, there exist users who get tired and do not transmit data. Moreover, there also exist the users who don't send once a day. Therefore, on average, each person has uploaded only 1 activity data at every other day.

4.1 Setting

The *activity classes*, which are the types of activities, are based on "Exercise Guide 2006" by the Ministry of Health Labour and Welfare, Japan, which gives a guideline to calculate *mets*: the strength of each activity. For this reason, several activities are divided to several strength, such as "walk slowly," "walk," and "walk fast." Additionally, to enrich the activity classes for transportations, some vehicle activities such as "train sit (sit in a train)" were added. Furthermore, we added classes for recreational events, such as "bbq" and "darts."

Finally, we adopted 46 activity classes and 19 device positions as shown in **Fig. 3**.

Activity Classes: bbq, bicycle, bowling, bus.sit, bus.stand, car, change.cloth, chat.sit (sit and chat), chat.stand (stand and chat), cook.stand, cut.grass, darts, eat.sit (sit and eat), elevator.down, elevator.down.5fl (go down in elevator over 4 floors.), elevator.up, elevator.up.5fl, escalator.down, escalator.up, escalator.walk.down, escalator.walk.up, ferry, mono.rail (ride monorail), motorbike, play.courage (play a test of courage), radio.taiso1 (a kind of physical excercise), recline, run, shinkansen (rapid train), sit, slope.down (walk down slope), slope.up, stair.down, stair.run.down (run stairs down), stair.run.up, stair.up, stand, stretch, test.data (for test), train.sit (sit in train), train.stand, type (do typing), walk, walk.fast, walk.slow, weight (weight training)
Device Positions: belt, breast.pocket, hand.bag, left.ankle, left.arm, left.hand, left.jacket.pocket, left.pants.pocket, left.wrist, neck.strap, no.label (as a result, right.ankle, right.arm, right.hand, right.jacket.pocket, right.pants.pocket, right.wrist, rucksack, shoulder.bag

Fig. 3 Listed activity classes and device positions.

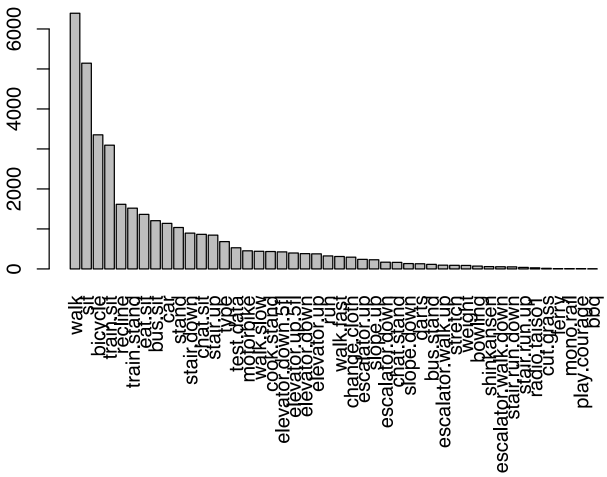

Fig. 4 Histogram of collected activity classes.

4.2 Activities

We acquired activity data, and reached a level of 35,310 activity data by the beginning of February 2011.

Figure 4 shows the number of missions for each activity type.

From the figure, we can see that simple daily life activities were the sources of most collected data.

4.3 Positions

Figure 5 shows the number of missions for each position. For simplicity, we only used those which are listed in the position information, and ones represented in XYZ coordinates on the body are classified as "no labels" as well as ones with no position information.

Together with no labels, "pants pocket" and "jacket pockets" were in the majority. Unsurprisingly, positions such as "ankle," "arm," and "belt" are hardly collected. "neck.strap" data are also

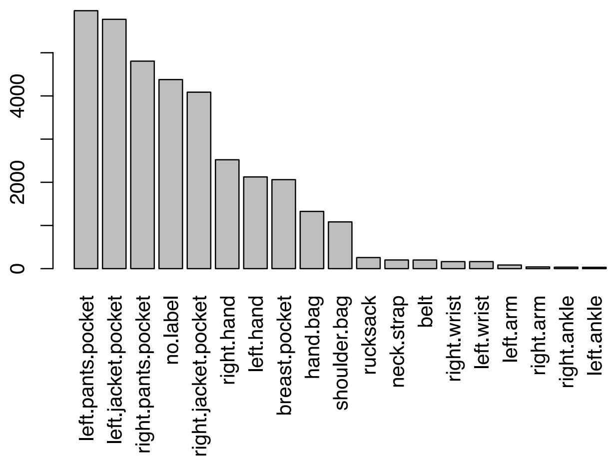

Fig. 5 Histogram of collected device positions on the body.

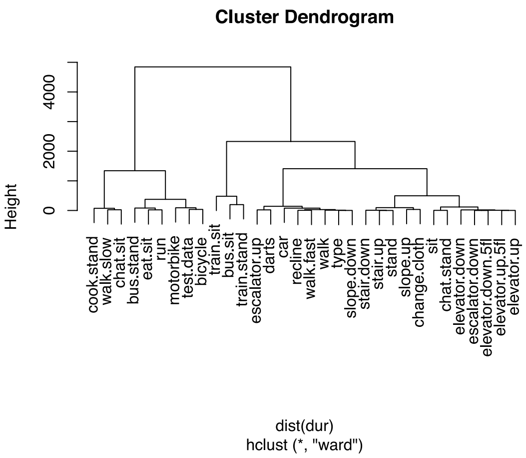

Fig. 6 Dendrogram of hierarchical clusters of activity classes by durations.

rarely collected, but may be more frequently collected in office situations. The reason why "backpack" data are not collected often is likely because putting a device in and out of a backpack is cumbersome, though it is more popular among students.

4.4 Clustering

To find the differences of durations among activity classes or device positions, we applied hierarchical cluster analysis as for mean durations to the groups by activity classes or device positions. In the analysis, Ward's minimum variance method with Euclidean distance was used.

Figure 6 is the dendrogram of the clustering result for the groups by activity classes as for mean durations, After dropping those groups with under 100 missions. In the dendrogram, a branch or a leaf means a cluster, and the sum of the height of any paths denotes the distance between the clusters of the end of the path.

From the figure, we can see that we can first divide the activity classes with a non-negligible distance into 2 clusters. From this, we can infer that there are both classes of longer duration and shorter duration.

4.5 Lessons Learned

From the long-running and large-scale data collection, we can conclude the following:

(1) The ALKAN system is well designed and developed. Since the system is stable and the data has been collected constantly, we can infer that the requirements for usability, motivation, and scalability addressed in Section 3.1 are satisfied as far as we know.

(2) The number of data collected is unbalanced in both activity classes and device positions. There is no problem. If the imbalance is caused by the probability distribution in natural life. However, since this might also be by the ease of recording data for each activity using ALKAN, but we have to balance them in some way.

(3) The durations have distributions with specific variance, but it can be used for clustering both activity classes and device positions. This means this knowledge can be used for improving recognition techniques for sequences of activities. Also, device position recognition in sequences can also utilize the technique.

As for (2), we propose and use a method to balance the number of activity classes and device positions in the following section.

5. Activity Recognition

In this section, we demonstrate activity recognition based on Bao et al. [7]. However, the goal of this section is not to improve the accuracy of activity recognition. As shown in Bao et al. [7], the accuracy of a single mobile sensor does not compare well against that from multiple mobile sensors. Moreover, better recognitions are proposed in the literature. Instead, the goal of this section is to show the nature of gathered data by ALKAN, using the most standard activity recognition method.

5.1 Sampling

As addressed in Section 4.5, the number of data collected has large biases. Therefore, we need a way to sample activities uniformly from the gathered data. Here, we present an algorithm to pick samples from the activity data sets according to the following policy:

(1) (uniform positions) The sampled set has the same number of positions.

(2) (uniform activity classes) The sampled set has the same set of activity classes among positions.

(3) (uniform users) The sampled set has the same number of users for each pair of position and activity, while the entities of users might be different for each pair.

(4) (uniform samples) The sampled set has the same number of samples for each combination of position and activity and user.

Algorithm 1 is the algorithm to achieve the policy. In the algorithm, we assume the Pos as the position set, Act as the set of activity classes, $User$ as the set of users, and $Sensor$ as the set of sensor data. D denotes the set of all the data, and we use the notation $Pos(D)$ ($Act(D)$, $User(D)$) to denote the position (activity class, users, respectively) factors of D.

Moreover, we use the notation of D_{apu} to specify the subset of D with activity class $a \in Act$, position $p \in Pos$, and user $u \in User$. We also introduce wild card notation such as D_{*p*}, which means a subset of D with any activity class and any user with position p.

Algorithm 1 Data Item Sampling

Input: $D = Pos \times Act \times User \times Sensor$, the number of users for each activity of position: n_u, and the number of samples for each user n.
Output: $D' \subseteq D$ which satisfies the policy above.

```
    // REMOVE DATA ITEMS WITH FEW SAMPLES
 1: for all (p, a, u) ∈ Pos(D) × Act(D) × User(D) do
 2:     D ← D − D_pau if ||D_pau|| ≤ n
 3: end for
    // REMOVE ACTIVITIES WITH FEW USERS
 4: for all (p, a) ∈ Pos(D) × Act(D) do
 5:     D ← D − D_*a* if ||User(D_pa*)|| ≤ n_u
 6: end for
    // USER SAMPLING
 7: D' ← ∅
 8: for all (p, a) ∈ Pos(D) × Act(D) do
 9:     randomly sample n_u users from User(D_pa*), and add the corresponding data to D',
10: end for
    // DATA SAMPLING
11: Result ← ∅
12: for all (p, a, u) ∈ Pos(D') × Act(D') × User(D') do
13:     randomly sample n items from D'_pau, and add them to Result
14: end for
15: return Result
```

In the algorithm, line 1–3 are for (4) uniform samples, by omitting the combinations of (position, activity, user) with a few samples and to ensure no less than n samples for any combination. In line 11–14, n samples are sampled for each combination.

Line 4–6 are for (3) uniform users, by omitting activities with few users, for any position, and to ensure no less than n_u users for any activity. In line 7–10, n_u users are sampled for each pair of position and activity.

We used the statistical processing software R [12] for data processing and machine learning shown in the rest of the paper.

Using the algorithm, we applied sampling upon gathered data, and investigated how many activity classes are sampled for several samples n, users n_u for single position. **Figure 7** is that of one of the positions, where $Pos = \{$`"left.pants.pocket"`$\}$.

From Fig. 7, the number of activity classes starts from 39 with 1 users, but decreases to 12 activity classes with 20 users, 6 with 60 users, and 2 with 80 users, when the number of samples is 1. If we take more samples, the values become lower, such as 8 activity classes for 20 users and 2 samples. We omit the results for other positions, but the curves are similar for other positions.

Thus, the number of activity classes and number of data is rapidly reduced when using the sampling algorithm. However, since the number of activity classes, users, and samples for each users and activity classes are balanced, we adopt this algorithm for the rest of the paper.

5.2 Feature Extraction

We extracted feature vectors from the 3-axis accelerometer data. At first, we removed 10 seconds of the beginning and ending, since it will include the action of touching, attaching, or operating the mobile sensors. By doing this, we were able to omit data of less than 20 seconds.

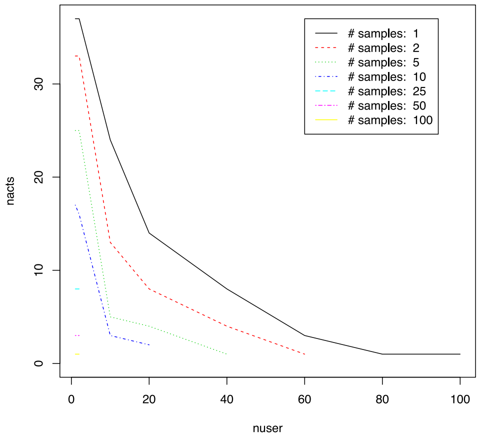

Fig. 7 Changes of # of activity classes for position: "left pants pocket."

After that, we applied the sampling algorithm shown above, and calculated feature vectors based on Bao and Intille [7] for sampled activity sensor data.

For a sensor data item, time windows of 5-second durations are extracted at first, shifting 2.5 seconds for each extraction. For the time window, we calculated:

- Mean value of each axis.
- Frequency-domain energy: the sum of the absolute FFT values divided by the number of the FFT values for each axis.
- Frequency-domain entropy: the entropy of the absolute FFT values minus the mean of the FFT values for each axis.
- Correlation among axes: the correlations between x-y, y-z, and z-x values. In Bao and Intille [7], they also used the correlations among multiple sensors on the body, but we only used correlations among individual sensors, since our setting is to use single sensor.

Thus, 12 dimensions are used for each feature vector.

5.3 Training

From the calculated feature vectors, we applied machine learning to generate a model for activity recognition. As machine learning algorithms, we used 1) a recursive partitioning tree, 2) a Naive Bayes classifier, 3) nearest neighbor classification, and 4) a support vector machine with a radial basis function kernel for the feature vectors.

To evaluate accuracy of recognition, we applied a special case of cross validation. Usual n-fold cross validation is to divide the data (feature vectors) to n groups, use $n-1$ groups for training and the rest 1 for test, and repeat it for n groups. Beside the traditional cross validation, we first picked up the subset for each activity, divided the users into n groups, and then divided the feature vectors according to the user groups. This is to maintain the balance of the numbers of each activity and user. The result is that users for training and testing are different.

Note that since each activity data item has variety of duration, and since we extract multiple time windows from it, the number for each activity for machine learning will not be uniform, but it can be assumed to be natural, since the duration was decided by the ALKAN users.

We adopted the position of "left.pants.pocket," picked 40 users,

Table 2 Confusion matrix of activity recognition.

→ trained ↓predicted	eat.sit	bicycle	car	sit	stand	train.sit	train.stand	walk
eat.sit	**1224**	170	719	1703	177	1742	165	184
bicycle	58	**1764**	1054	41	161	178	182	462
car	146	965	**3371**	918	44	1028	98	1041
sit	1193	22	538	**4156**	160	1685	139	65
stand	255	293	15	4210	**392**	1321	701	379
train.sit	803	61	1132	2625	228	**1751**	934	20
train.stand	780	172	403	341	988	272	**2684**	1072
walk	388	470	252	1278	514	50	431	**3732**

Table 1 F-measures(%) for each activity recognition algorithm.

Activity Class	Rpart	NB	1-NN	SVM
eat.sit	10.74	5.19	17.58	22.40
bicycle	40.91	46.61	36.64	45.10
car	37.69	16.82	25.38	44.65
sit	33.77	55.09	30.42	35.76
stand	2.64	1.71	11.11	7.68
train.sit	36.77	8.79	23.83	22.48
train.stand	48.89	62.67	38.12	44.56
walk	54.16	50.95	42.03	53.09

and sampled 1 activity data for each pair of activity and user. By this sampling, we obtained 8 activity classes. For the sampled data, we applied feature extraction, trainings, and 3-fold cross validations by each algorithm. **Table 1** is the F-measure for each activity class and algorithm.

Even though we utilized the same feature vectors, the accuracy in this table is worse than that obtained by Bao and Intille [7], although we used the same feature vectors. Aside from using single mobile sensor, the following are considered as the reasons:

- A mobile sensor was not firmly fixed to the body, and was shaken around in the pocket.
- The number of users is as large as 40 users. It may make the feature vectors non-general.
- Activity classes are similar to each other. **Table 2** is the confusion matrix of the recognition. As we can imagine, similar activity pairs such as "eat.sit"–"sit" and "sit"–"train.sit" are often misrecognized.
- Actual activities may have varieties. Since users have performed activities in their own situations, environments could differ greatly on each trial.
- Labels are not clearly understood by the users. Since we only showed the names of labels, users may have understood each activity in varieties of ways.

Although these factors will decrease the recognition accuracy, they can produce a more challenging data set for activity recognition since these situations are more realistic than laboratory settings which most of the existing work studied.

In this sense, these results of worse accuracy imply the data set gathered by ALKAN is a 'good' data. Especially, it will be valuable as a good test bed when we find sophisticated feature vectors in the near future.

6. Conclusion

In this paper, we developed an activity data gathering system using mobile sensor devices. We also described the gathered data, introduced the sampling algorithm from unbalanced data, and presented activity recognition results to show the nature of obtained data.

ALKAN data are open and free to use. ALKAN users have already agreed to make the data available to the public. Open data is necessary since several techniques have been proposed for activity recognition. Methodologies must be evaluated using the same data set. ALKAN could prove a useful platform for evaluating current or future activity recognition methodologies.

Reference

[1] Chambers, G.S., Venkatesh, S., West, G.A.W. and Bui, H.H.: Hierarchical recognition of intentional human gestures for sports video annotation, *Proc. 16th International Conference on Pattern Recognition*, Vol.2, pp.1082–1085, IEEE Press (2002).

[2] Herren, R., Sparti, A., Aminian, K. and Schutz, Y.: The prediction of speed and incline in outdoor running in humans using accelerometry, *Medicine & Science in Sports & Exercise*, Vol.31, No.7, pp.1053–1059 (1999).

[3] Kern, N., Schiele, B. and Schmidt, A.: Multi-sensor activity context detection for wearable computing, *European Symposium on Ambient Intelligence* (*EUSAI*) (2003).

[4] Lee, S.W. and Mase, K.: Activity and location recognition using wearable sensors, *IEEE Pervasive Computing*, Vol.1, No.3, pp.24–32 (2002).

[5] Mantyjarvi, J., Himberg, J. and Seppanen, T.: Recognizing human motion with multiple acceleration sensors, *Proc. IEEE International Conference on Systems, Man, and Cybernetics*, pp.747–752, IEEE Press (2001).

[6] Van Laerhoven, K. and Cakmakci, O.: What shall we teach our pants? *The 4th International Symposium on Wearable Computers*, pp.77–83, IEEE Press (2000).

[7] Bao, L. and Intille, S.S.: Activity Recognition from User-Annotated Acceleration Data, *International Conference on Pervasive 2004*, pp.1–17, Springer-Verlag GmbH (2004).

[8] Uiterwaal, M., Glerum, E.B., Busser, H.J. and van Lummel, R.C.: Ambulatory monitoring of physical activity in working situations, a validation study, *Journal of Medical Engineering & Technology*, Vol.22, No.4, pp.168–172 (1998).

[9] Bao, L. and Intille, S.S.: Activity Recognition from User- Annotated Acceleration Data, *International Conference on Pervasive 2004*, pp.1–17, Springer-Verlag GmbH (2004).

[10] Kawaguchi, N., Ogawa, N., Iwasaki, Y., Kaji, K., Terada, T., Murao, K., Inoue, S., Kawahara, Y., Sumi, Y. and Nishio, N.: HASC Challenge: Gathering Large Scale Human Activity Corpus for the Real-World Activity Understandings, *Proc. Augmented Human Conference*, (to appear) (Mar. 2011).

[11] Berchtold, M., Budde, M., Gordon, D., Schmidtke, H.R. and Beigl, M.: ActiServ: Activity Recognition Service for Mobile Phones, *Proc. Int'l Symp. Wearable Computers* (*ISWC*), pp.1–8 (2010).

[12] R Development Core Team: *R: A language and environment for statistical computing*, R Foundation for Statistical Computing, Vienna, Austria (2010). ISBN 3-900051-07-0, available from ⟨http://www.R-project.org/⟩.

[13] ALKAN web site, available from ⟨http://alkan.jp⟩.

Yuichi Hattori was born in 1987. He is a Ph.D. candidate at Kyushu Institute of Technology, Japan. His reseach interests include human activity recognition with smart phones, healthcare application of web/pervasive/ubiquitous systems. He is a member of the ACM and the Information Processing Society of Japan (IPSJ).

Sozo Inoue is an associate professor in Kyushu Institute of Technology, Japan. His research interests include human activity recognition with smart phones, healthcare application of web/pervasive/ubiquitous systems. Inoue has a Ph.D. in Engineering from Kyushu University. He is a member of IEEE Computer Society, ACM, the Information Processing Society of Japan (IPSJ), the Institute of Electronics, Information and Communication Engineers (IEICE), and the Database Society of Japan (DBSJ).