Toward High-level Activity Recognition from Accelerometers on Mobile Phones

Sozo INOUE
Faculty of Engineering,
Kyushu Institute of Technology
Kitakyushu, Japan
sozo@mns.kyutech.ac.jp

Yuichi HATTORI
Graduate School of Engineering,
Kyushu Institute of Technology
Kitakyushu, Japan
j350925y@tobata.isc.kyutech.ac.jp

***Abstract*—In this paper, we propose an unsupervised method for multi-level segmentation, which could be used for a pre-process of non-sequential activity recognition, and could construct a high-level activity recognition using accelerometers on mobile phones. We extend single-level segmentation to multi-level by sweeping the temporal parameter. To confirm the validity of our approach. we pursued the experiment of gathering accelerometer data of real nursing in a hospital. After the experiment and multi-level segmentation, we confirmed several phenomena to imply the validity of multi-level segmentation such that sequence seems to be properly segmented fitting to the annotations transcribed from the voice, that there are peaks of lower-level segment boundaries without higher-level boundaries, and that higher-level boundaries are not lower-level boundaries.**

***Keywords*-Human activity recognition, three-axis accelerometer, smart phone, multi-level segmentation**

I. Introduction

Recent deployment of smart phones equipped with accelerometers will make it possible to recognize activities of the users. If human activity can be objectively measured, we can expect various applications. For example, lifestyle aspects can be quantified and used for prevention of lifestyle-related diseases. In that of agriculture, farmers can improve efficiency by automatically obtaining their own activity record. Moreover, in more domain specific application such as nursing management, nursing activity will be quantified and optimized for various type of process in hospitals.

However, existing work for activity recognition has several problems, such as the following:

- It does not consider sequential activities. If so, the accuracy of recognition around the period of transferring one to another in a naive time-window approach.
- It only targets on simple types of activities. More complex activities, which are sometimes more abstracted or composed of simple activities, are not tried to be recognized regarding the complexity.

In our research, we take the approach of dividing the activity recognition problem into:

1) firstly apply a segmentation method which outputs multi-level segments.
2) upon the multi-level segments, apply a traditional activity recognition method, adjusting the level of segmentation adaptively.

In this paper, we focus on the first step, and propose unsupervised method for multi-level segmentation. The proposed method is the modified version of [7] to equally weight for each feature vector. Moreover, we extend single-level segmentation to multi-level by sweeping the temporal parameter. By this, we can construct a tree of segments which represents structure among levels.

To confirm the validity of our approach. we pursued the experiment of gathering accelerometer data of real nursing in a hospital. To collect activity data efficiently, we used a large-scale activity gathering system named ALKAN[2], [3], [6] with smart phones. We asked nurses working at a large hospital to place iPodTouches in to their breast pockets with a roughly fixed direction. During the measurement of accelerometer, we also recorded the sound by the same iPodTouches. The voice data recorded were transcribed to annotations by third parties, who have somewhat knowledge about nursing.

After the experiment and multi-level segmentation, we confirmed several phenomena to imply the validity of multi-level segmentation such that sequence seems to be properly segmented fitting to the annotations transcribed from the voice, that there are peaks of lower-level segment boundaries without higher-level boundaries, and that higher-level boundaries are not lower-level boundaries.

II. Requirement for High-level Activity Recognition

In this section, we discuss the requirement for high-level activity recognition.

A. Annotation for sequential activities

In our previous experience (abstracted in Sec. IV-E), we adopted to obtain activities and their annotation one by one, by the concept of ‘mission’ to keep annotation correct on time axis. However, if we challenge real data acquisition, we cannot avoid obtaining sequential activities, in which the occurrence probabilities of each activity can be provided, and also new activities we do not know beforehand are included.

When we obtain sequential activity data, annotation becomes more difficult. In the following, we discuss the difficulty from several aspects: segmentation and multi-level activity classes.

1) Segmentation: Segmentation, which is to divide the sequential data into several activity parts, can be considered to be as a first part of annotation for sequential data, and to be the major reason why manual annotations are inaccurate. If we do offline segmentation afterwards, many activity information will be lost, and the timing will be largely misfocused. If we do online, a user has to do additional action, or often forgets to do segmentation. If we engage an observer to annotate the user's activity, it often costs too much to scale up the data.

2) Multi-level activity classes: A single activity can often be captured by observers as different activities. For example, 'cooking' consists of 'cutting', 'boiling', and so on, and each of which consists of more low-level activities such as 'moving hands', and 'standing'.

One of the countermeasures for it is to structure annotations for activities as multi levels. An activity can consist of several activities, and it can also be a part of others. By this, we can expect to reduce the space for annotation words by knowing the relationship among them, and the recognition of a high-level activity which consists of a sequence of lower-level activities becomes easier.

B. Knowing device status

From the result of ALKAN, the variation of device status might affect the recognition accuracy. Device status consists of such as the direction, position on the body, and clothes types. Since these could provide different sensor data, or provide noises for activity recognition, knowing them is important for refining activity recognition.

C. Requirements

Since the challenges of annotations for sequential activities and of the device status critically affect the recognition accuracy, solutions for them are highly important. Here, we discuss two directions for solutions: unsupervised method and complementary information.

1) Unsupervised method: When annotation cannot be trusted, we can consider using unsupervised method. Unsupervised methods do not require annotations in training data, and only analyze the relationship among sensor data items. Then, we can extract information about activities, which is independent from annotations.

Recently, several methods utilizing unsupervised methods are proposed such as [4], [5]. These approaches should be more discovered.

Of course, unsupervised method must be combined with some supervised methods with annotations. However, performing basic parts such as segmentations and knowing device status with unsupervised methods as much as possible, and assigning the rest to supervised methods are a hopeful direction, since it can minimize the inaccuracy of manual annotations.

2) Complementary information: The other direction is to use complementary information along with the target sensor data. Sound data is nowadays easy to be recorded with the same device with accelerometers. Videos can be also useful when there are such environments, or observers. Kinetic information is recently become easy to be captured using widely spread consumer devices. Since these data are objective, they can help obtaining annotations for activities if the timing is synchronized correctly.

Moreover, several informations can be obtained from application systems. For example, location information can be also obtained in location-based services. Moreover, in healthcare systems, body features and lifestyle information of each user will be stored in the system database.

The challenges for these complementary informations reside not only in data analysis, but also in system design, where these complementary information should be obtained with less stresses of users, no privacy risk, and lower costs.

III. Modeling High-level Activity Recognition

Based on the requirements discussed in the previous section, we describe our approach of dividing the recognition problem into multi-level segmentation and traditional activity recognition.

A. Segmentation for high-level activity recognition

As mentioned in Sec. II-A2, structuring multi-level activity model is important for high-level activity recognition. When we consider temporal sequences, the multi-level can mean in two ways:

1) An activity can be captured as another activity of the same time range.
2) An activity can be a part of another one of longer range.

Among them, 1 can be assumed to be independent of temporal information, so it is handled as a problem of how to model the activity class in a structured way. On the other hand, 2 is highly dependent on temporal information. Since the relation of temporal inclusion among shorter activity and longer activity can be structured as a tree graph, it can be considered as one of the methods for multi-level activity model.

Then, we take the approach of dividing the activity recognition problem into:

1) firstly apply a segmentation method which outputs multi-level segments.

2) upon the multi-level segments, apply a traditional activity recognition method, adjusting the level of segmentation adaptively.

In this paper, we focus on the first step, and propose unsupervised method for multi-level segmentation.

The advantages of our approach are as follows:

- Since activity sequences are firstly divided into multi-level segments, we apply activity recognition of traditional way in multi candidates of time ranges. Even if one might fail to recognize, or inappropriate segmentation, another can be OK, so we can improve the recall ratio of recognition as a result.
- Since we take the approach of unsupervised segmentation, it does not need any training data, and not need to take care of the quality of the training data. Since the quality of training data highly affects the accuracy of later steps, that kind of matter should be eliminated especially in the earlier segmentation step.
- The information obtained from segmentation can be utilized in the later steps. In our experience (described in Sec. IV-E), the duration information of each activity can be helpful to do activity recognition. Since segmentation step of course outputs the information of duration of each segment, the over all activity recognition can be improved if the former segmentation is done precisely.
- Our segmentation approach use the same precomputed feature vectors as the latter activity recognition step, so we can reduce the overhead of segmentation than using independent values.

B. Multi-level segmentation

In this section, we propose a method for multi-level segmentation.

Here, we assume that we have a feature vector $F(t)$ for each time t where t is simplified to be an integer, and it is denoted as a vector of dimension M:

$$F(t) = (f_1(t), f_2(t), \cdots, f_M(t))$$

Feature transition measure for an element $f_m(t)$ with an integer $1 \leq m \leq M$ of a feature vector $F(t)$ is defined as:

$$ftm_m^N(t) = \frac{\sum_{n=1}^{N} f_m(t+n) - \sum_{n=1}^{N} f_m(t-n)}{N}$$

Here, N is the number of time points before or after t to take into consideration. Since feature vectors are extracted from sequential time windows, we name the time interval $[t-N, t+N]$ an *input region*.

The definition of ftm here is the modified version of [7] to equally weight $f(t+n)$ for each n. In the original[7], the features are weighted by the time difference from t, which means $n \times f(t+n)$ before normalization. However, since there is no reason why farther features have more effect, we set to weight equally.

Upon ftm for single dimension, we define *Feature Transition Measure* $FTM_N(t)$ for the feature vector $F(t)$ as a quadratic mean of ftm for all m:

$$FTM_N(t) = \sqrt{\frac{\sum_{m=1}^{M} ftm_N(t)^2}{M}}$$

We can segment activities and find time boundaries by a way such as smoothing $FTM_N(t)$ and extracting peaks over some threshold value.

The value N can be a parameter for input region. However, since it has a temporal information, we can consider to use N as a level parameter for multi-level segmentation where the height of levels are related to temporal durations.

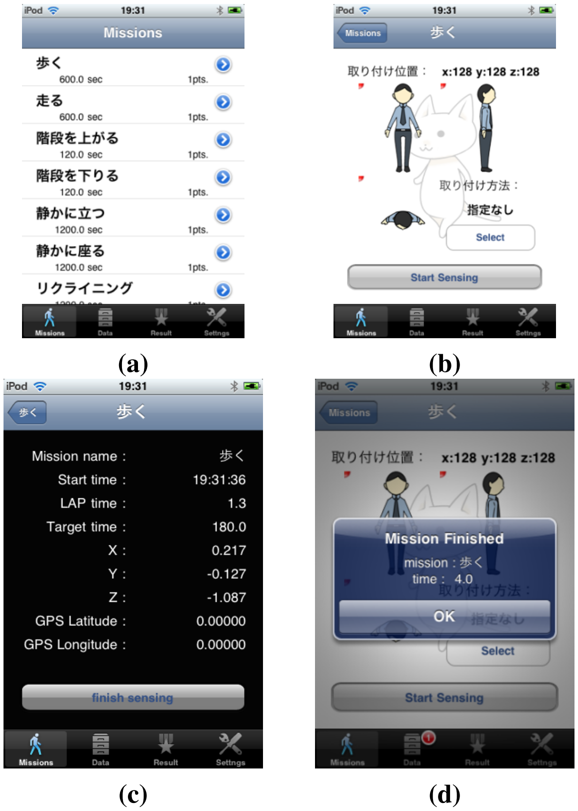
Using varying value of N, we construct multi-level segments from multi-level input regions. Here, let $T_N = \{t_{N,1}, t_{N,2}, \cdots\}$ be the finite set of time boundaries segmented by $FTM_N(t)$.

1) Choose several values $N_1, N_2, \cdots, N_{\mathcal{N}}$ where $0 < N_1 < N_2 < \cdots < N_{\mathcal{N}}$.
2) Calculate $FTM_{N_i}(t)$ for each $1 \leq i \leq \mathcal{N}$, and obtain T_{N_i}.
3) For any T_{N_i} and T_{N_j} where $1 \leq i < j \leq \mathcal{N}$, let $T_{N_j} \leftarrow T_{N_i} \cup T_{N_j}$.

The last step means that the boundaries of higher-level segments also become those of lower-level segments. By this, any segment $s \in T_{N_i}$ except for the highest-level segments has one-level higher segment $s' \in T_{N_{i+1}}$ which includes s, which means the range of s is included by that of s'. Thus, we can construct a tree of segments, where the segment s is a child of s'.

IV. ALKAN System

To collect activity data efficiently, we used a large-scale activity gathering system named ALKAN[2], [3], [6]. In this section, we describe the system design. In ALKAN, to achieve accuracy of annotations, we introduce the idea of "mission". A *mission* is a sequence of choosing an activity so-called *activity class*, choosing the position of the device on the body, and performing the activity. Using this method, users can record activities anytime they want, and the annotation is accurately stamped within deviations of few seconds. And, for usability, we adopted smart phones as mobile sensor devices. Most smart phones are equipped with 3-axis accelerometers, storage, and wireless communication, which enable recording activity data anytime. The data can be uploaded to the server when it is connected to the network. 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.



歩く 600.0 sec 1pts.
走る 600.0 sec 1pts.
階段を上がる 120.0 sec 1pts.
階段を下りる 120.0 sec 1pts.
静かに立つ 1200.0 sec 1pts.
静かに座る 1200.0 sec 1pts.
リクライニング

(a)

歩く

取り付け位置： x:128 y:128 z:128

取り付け方法： 指定なし

(b)

Mission name :	歩く
Start time :	19:31:36
LAP time :	1.3
Target time :	180.0
X :	0.217
Y :	-0.127
Z :	-1.087
GPS Latitude :	0.00000
GPS Longitude :	0.00000

(c)

Mission Finished
mission : 歩く
time : 4.0

(d)

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

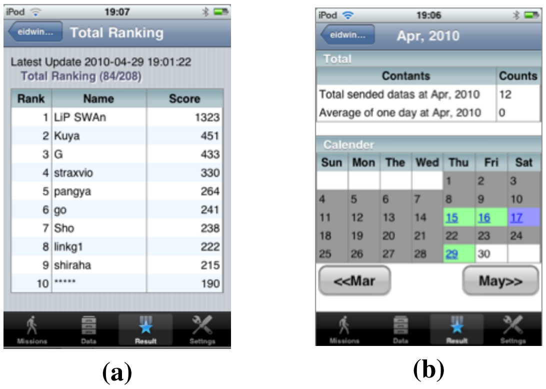
A. System Architecture

The ALKAN system consists of mobile device clients and a server. 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 smart phone or another web browser on a PC.

1) Client: As for the client, we developed 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 Fig2.

The client software has the following functionalities:

- Mission execution: users first select an activity class as in Fig.1(a) and a position as in Fig.1(b). Then they start the activity as in Fig.1(c) and finish as in Fig.1(d). The sensor can record three axis accelerometer data at 20Hz.
- View and send mission history: users can view the recorded mission history and add comments to each mission as an annotation. Users can also delete missions if s/he 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 by all at once. After the mission data are sent, they are removed from the history.



Total Ranking

Latest Update 2010-04-29 19:01:22
Total Ranking (84/208)

Rank	Name	Score
1	LiP SWAn	1323
2	Kuya	451
3	G	433
4	straxvio	330
5	pangya	264
6	go	241
7	Sho	238
8	linkg1	222
9	shiraha	215
10	*****	190

(a)

Apr, 2010

Total

Contants	Counts
Total sended datas at Apr, 2010	12
Average of one day at Apr, 2010	0

Calender

Sun	Mon	The	Wed	Thu	Fri	Sat
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	

(b)

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

- View statistical information of the server: the software shows a web browser to access the server and show statistical information, such as ranking as in Fig.2(a) and calendar history as in 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.

B. Server

The server gathers the activity data sent from clients, stores it to the database, and calculates and serves statistical information as a web server. An example of current statistical information displayed is the total/individual rankings of the number of executed missions among users. Other statistical information is the history of executed missions for each user. Users can view the start/end date/time, activity class, and positions This is similar to lifestyle-related services in which users record their own lifestyles.

C. 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 or CSV-formatted data are passed between the client the server. In XML-formatted data, activity classes and position list are provided by the server to clients, and the metadata for each mission is uploaded.

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.

D. Sound data

In the recent version up of the client software, we added the functionality of recording sound data from the mic equipped on iPhones and recent version of iPodTouches. Once a sound is recorded, the data is treated in the same way as acceleration data in a separated file, and uploaded to the server. Since server is flexible to accept different types of data, the server needs no need to be specialized for this data.

Recording sound data with the same device as accelerometer has a merit that the time is automatically synchronized in the client software. In an experiment with multiple sensors, timing synchronization is simple but cumbersome work. If the synchronization fails unconsciously, it has significant bad effect to the later analysis steps. By integrating sound recorder and accelerometer in a single device, we can eliminate this risk.

Of course, the sound data is considered to be sensitive to privacy in some cases, since the device might record the sound of the environment, or the people around. The sound data should be carefully treated even if it is agreed with the users. Therefore, in ALKAN, the sound data is not opened to the public by default.

E. Lessons Learned

We have delivered 216 iPodTouchs as smart phones to university students and staff. We asked users a favor to collect activity data once a day on average. As a result, we gathered 35,310 missions during about 14 months.

Upon the sampled data, we applied the well-known activity recognition method by Bao and Intille[1]. Surprisingly, the result was worse than shown in the single sensor case in [1]. The following are considered as the reasons:

- A mobile sensor is not firmly fixed to the body, but shaken in the pocket.
- Activity classes are similar to each other. As we can imagine, similar activity pairs such as "eat.sit"–"sit" and "sit"–"train.sit" are often mis-recognized.
- Actual activities may have varieties. Since users have performed activities in their own situations, environments could differ greatly on each trial.
- Labels are ambiguously understood by users. Since we do not have a method to verify activities, users are even possible to lie performing activities.

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 traditional laboratory settings.

V. Experiment

In this section, we describe the experiment for multi-level segmentation of activity sequence data in a real hospital.

A. Setting

Using ALKAN, we pursued the experiment of gathering accelerometer data of real nursing in a hospital. We asked nurses working at a large hospital to place iPodTouches in to their breast pockets with a roughly fixed direction. Since we did not use any tool to fix the devices, they have a possibility to be tilted or shaken during actions.

During the measurement of accelerometer, we also recorded the sound by the same iPodTouches. Recording accelerometer data and sound data at the same device has an advantage that the times are automatically synchronized each other.

Moreover, in nursing, voice information is thought to be useful from our experience, since nurses usually talks to patients about what to do from the time. If the voices are recorded in the sounds, we can use the information to annotate and segment the activity sequence.

The voice data recorded were transcribed to annotations by third parties, who have somewhat knowledge about nursing. At transcription, templates for several activities were originally prepared, and the transcribers tried to adopt the templates as much as possible. The activities difficult to use the templates were written in natural language. Moreover, activities and ranges which are too unclear to transcribe were left blank.

This experiment is ongoing for longtime and for larger population. However, we only target on the 29 trials each of which has under 4 hours.

B. Feature Extraction

We extracted feature vectors from the 3-axis accelerometer data, based on [1] for sampled activity sensor data.

At first, time windows of 5-second durations are extracted at first, shifting 2.5 seconds for each extraction. For the time window, we calculated mean, frequency-domain energy, and frequency-domain entropy for each axis.Moreover, correlations among axes, which are single sensor versions of [1]., are calculated.

- Mean value of each axis.
- Frequency-domain energy: the sum of the absolute FFT values divided by the number of them for each axis.
- Frequency-domain entropy: the entropy of the absolute FFT values minus the mean of them for each axis.
- Correlation among axes: the correlations between x-y, y-z, and z-x values. Bao et al.[1] also used the correlations among multiple sensors on the body, but we did only used those inside single sensor device.

Thus, 12 dimensions are used for each feature vector.

C. Multi-level segmentation

For the feature vectors of 12 dimensions, we calcurated $FTM_N(t)$ for $N = 50(,20,10,5)$. Since the size of time window is 5 seconds, the duration of the input region is $250(,100,50,25,$ respectively) seconds.

D. Observation

We show a case of segmentation and annotations in Fig.3. The figure corresponds to an nursing activity sequence of 1) entering the room, 2) measuring blood pressure of the patient, 3) measuring cardiogram, 4) assisting walk, 5) measuring blood pressure, 6) measuring cardiogram, 7)

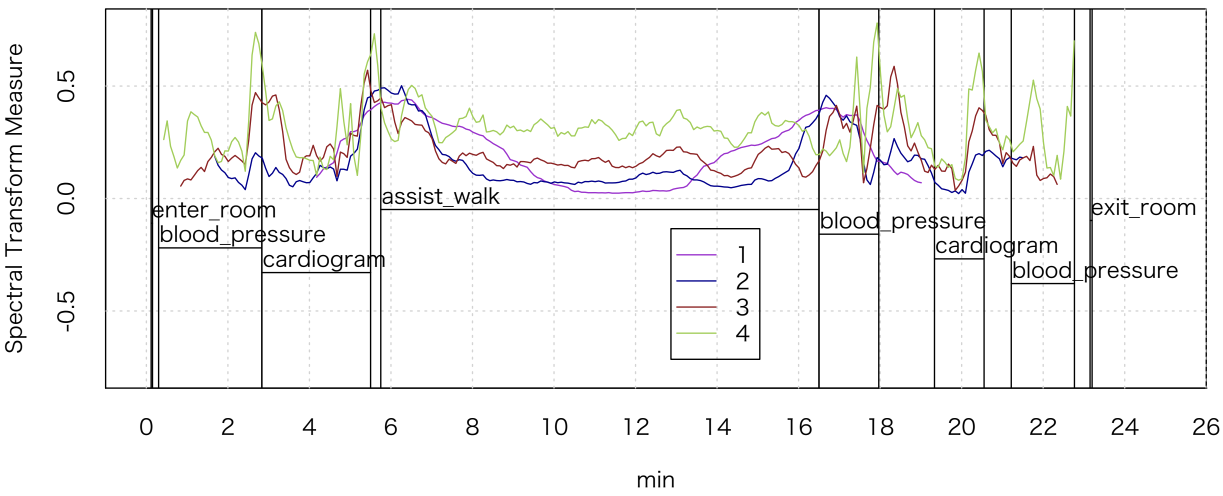


Figure 3. **Multi-level Feature transform measures for an activity sequence of 1) entering the room, 2) measuring blood pressure of the patient, 3) measuring cardiogram, 4) assisting walk, 5) measuring blood pressure, 6) measuring cardiogram, 7) measuring blood pressure, and 8) exiting the room. Line 1 is $N = 50$ (250 sec), line 2 is $N = 20$ (100 sec), line 3 is $N = 10$ (50 sec), and line 4 is $N = 5$ (25 sec).**

measuring blood pressure, and 8) exiting the room. We describe the observation of the figure in the following:

- The higher the level, the wave becomes smoother, since it uses longer input regions. It means that there are peaks of lower-level FTMs without higher-level peaks.
- Ther higher-level FTM has a lack in the beginning and the ending of the sequence, since it has to use longer input regions.
- Although we need to apply smoothing, peak detection, and thresholding, sequence seems to be properly segmented fitting to the annotations transcribed from the voice.
- Most periods with Large peaks of a higher-level FTM seems also to be those of lower-level FTM, but there are exceptions, such as at slightly after the boundary at 16.5 min, where the higher-level FTM of line 2 has a peak while the lower-level FTM of line 3 does not.

E. Discussion and Challenges

From the observation above, we can observe the necessity to sweep multiple-levels of segmentations, and to find the optimum level for each time period. Otherwise, an important boundary might be missed, or an incorrect boundary might be found.

Why this happens can be considered to be since FTMs are calculated from temporal information of input ranges. Unnecessarily longer input range includes ranges over single neighbor segments, and unnecessarily shorter input range might miss important features in the neighbor segment. The input range should be selected according to the range of the segment. However, the range of the segment is unknown at first. An adaptive method to select the input range (level of segment) is a future challenge.

Another challenge is to utilize periodic patterns of feature vectors. For example, some activity segment might have periodic patterns, such as stopping at several bus stops in riding a bus. Since these periodic patterns are considered to be flattened by the summations in ftm, more improvement will be required.

Integrating other data is also a challenge. In this time, we used the voice data for annotation and comparison with segments by accelerometers only, but they can be also considered to be used for improving segmentation. These kinds of sensor data fusions are also interesting research field.

VI. Conclusion

In this paper, we propose an unsupervised method for multi-level segmentation, which could be used for a preprocess of non-sequential activity recognition, and could construct a high-level activity recognition using accelerometers on mobile phones. After the experiment and multi-level segmentation, we confirmed several phenomena to imply the validity of multi-level segmentation.

ALKAN data are open and free to use. The users of ALKAN have already agreed with opening the data to public. Open data is necessary since several techniques have been proposed for activity recognition. Activity recognition methodologies must be evaluated using the same data set. ALKAN could be the platform for evaluating existing or future activity recognition methodologies.

VII. Acknowledgements

This work is supported by Grant-in-Aid for Young Scientists (A) (21680009) of JSPS. The authors would like to thank their support. We also appreciate the cooperation for experiment by Prof. Naoki Nakashima, Prof. Yasunobu Nohara, Hiroko Tsuruta, and the staff of Saiseikai Kumamoto Hospital, Japan.

REFERENCES

[1] L. Bao and S. Intille, ”Activity Recognition from User-Annotated Acceleration Data”, Proc. Pervasive 2004, pp. 1–17.

[2] ALKAN web site, http://alkan.jp/

[3] Y. Hattori, S. Inoue, G. Hirakawa, ”A Large Scale Gathering System for Activity Data with Mobile Sensors”, Proc. ISWC2011, pp. 97-100.

[4] H. Bayati, J. Millan, R. Chavarriaga, ”Unsupervised adaptation to on-body sensor displacement in acceleration-based activity recognition”, Proc. ISWC2011, pp.71-78.

[5] T. Maekawa, S. Watanabe, ”Unsupervised Activity Recognition with User's Physical Characteristics Data”, Proc. ISWC2011, pp. 89-96.

[6] Go Hirakawa, Yuichi Hattori, Masato Nakamura, Sozo Inoue, ”Activity Information Sharing System with Video and Acceleration Data”, Proc. Int'l Conf. Pervasive and Embedded Computing and Communication Systems, pp. 557-561, March 5, 2011, Algarve, Portugal.

[7] Sorin Dusan and Lawrence Rabiner, ”On the relation between maximum spectral transition positions and phone boundaries”, In INTERSPEECH-2006, pp.645-648, 2006.

[8] Suutala, J., Pirttikangas, S. and Roning, J., ”Discriminative Temporal Smoothing for Activity Recognition from Wearable Sensors”, Ubiquitous Computing Systems 2007, Lecture Notes in Computer Science, Vol.4836, Springer Berlin / Heidelberg, pp.182?195.

[9] Brent Longstaff, Sasank Reddy, Deborah Estrin, ”Improving Activity Classifcation for Health Applications on Mobile Devices using Active and Semi-Supervised Learning”, 4th International Conference on Pervasive Computing Technologies for Healthcare (PervasiveHealth) 2010, pp.1-7, 2010.