Metric Based Automatic Event Segmentation

Yuwen Zhuang[1], Mikhail Belkin[1], and Simon Dennis[2]

[1] Department of Computer Science and Engineering, Ohio State University
Columbus, OH 43201, USA
zhuang.14@buckeyemail.osu.edu, mbelkin@cse.ohio-state.edu
[2] Department of Psychology, Ohio State University
Columbus, OH 43201, USA
simon.dennis@gmail.com

Abstract. This paper describes a metric-based model for event segmentation of sensor data recorded by a mobile phone worn around subjects' necks during their daily life. More specifically, we aim at detecting human daily event boundaries by analysing the recorded triaxial accelerometer signals and images sequence (lifelog data). In the experiments, different signal representations and three boundary detection models are evaluated on a corpus of 2 subjects over total 24 days. The contribution of this paper is three-fold. First, we find that using accelerometer signals can provide much more reliable and significantly better performance than using image signals with MPEG-7 low level features. Second, the models using the accelerometer data based on the world's coordinates system can provide equally or even much better performance than using the accelerometer data based on the device's coordinates system. Finally, our proposed model has a better performance than the state of the art system [1].

Keywords: Triaxial accelerometer, Event segmentation, Lifelog data.

1 Introduction and Motivation

Lifelogging, as a growing interest, is a term referring to people digitally capturing all the information produced by them in daily life. Lifelog is a data collection of records of an individual's daily activities in one or more media forms. This may contain huge volumes of data from different sensor sources. For example, in the study of [2], an average collection of 1,900 images per day per person leads to approximate 700,000 images per year per person. Hence, a main challenge is how to add utility to this huge and complex collection that is continuously captured and accumulated from multiple sensors [3].

Peoples' daily lives consist of different events that come in many varieties. An event is an organization of human experience, such that a dynamic and continuous experience is divided into stable entities, providing a structure for attention, memory and learning [4]. Event segmentation, as one of the most fundamental intelligent mechanisms that a human possess, is a process where people segment a continuous stream of experience into meaningful events [5].

D. Uhler, K. Mehta, and J.L. Wong (Eds.): MobiCase 2012, LNICST 110, pp. 129–148, 2013.
© Institute for Computer Sciences, Social Informatics and Telecommunications Engineering 2013

Recently, event segmentation [6,1] has been suggested as a method to organize lifelogs where an event boundary occurs when there is an end of one meaningful event and another event begins.

There are several psychologic foundations for using the event segmentation as a methodology for organizing lifelogs. First, supported by behavioral [7] and neuro-imaging data [8,9], it has been suggested that event segmentation is automatic [10,11], even as observers passively view activities. Second, segmenting activity into meaningful events is indicated as a core component of perception [12] and has consequences for memory and learning [13,14]. For example, first, [5] show that individuals who are better able to segment ongoing activity into events are better able to remember it. Second, by event segmentation, a terrific economy of representation for perception and memory can be achieved. Hence, organizing lifelogs into events will provide a more natural and pellucid way for organizing lifelogs, retrieval and interpretion.

However, several related and crucial questions for event segmentation still remain. First, are the event boundaries consistent across different people? In other words, is there good inter-subjective agreement on the event segmentation boundaries? Although the event boundaries can be **fuzzy** [12] indicating there is inevitable variability when and where a boundary occurs, they are **remarkably consistent** across participants [15] with **good reliability** among the same participants in test-retest [7].

Second, what kind of features are needed? [5] show that event boundaries can be identified by tracking significant changes in physical and social features. [12] indicates that the critical features may include sensory features, such as color, sound and movement, and conceptual features, such as cause-and-effect interactions and actors' goals. Evidences [12] are shown to demonstrate that both physical-movement features (such as change in location) and changes in actor's goals play strongly important roles in the segmentation of activity into events.

Finally, since previous studies [10,16,17] only study the movement features on event segmentation for the short time records which last only for a few seconds to several minutes, the final question is - What is the impact for movement features for event segmentation for long time records (which last more than 2 hours or even longer)?

In this paper, our study on natural and realistic daily event segmentation shows that, for daily event segmentation on long time records, the movement features can also play an important role and are important cues for event boundary detection. More specifically, our study focuses on the impact of movement feature and visual feature characterized by two independent sensor sources, namely, accelerometer and image on event segmentation. Furthermore, we will also follow the same way [1] to construct the ground truth boundaries, that is asking people to segment their events by viewing and marking event boundaries on their corresponding image records. But instead of associating the sensor data to images [1] to evaluate performance, we suggest a performance measure which is well developed from audio segmentation for daily event segmentation using sensor data. This measure considers the "fuzzy" effect [12] of event boundaries by assuming that the event boundary can have a small continuous time interval and evaluates the F-score (a measure of a test's accuracy based on precision and recall) directly on event boundaries. This is different from [1] which evaluates F score on images and [18] which evaluates F score on events (activities).

The specific contributions of this paper are the following:

1. Accelerometer signals can provide much more reliable and significantly better performance than using images signal with MPEG-7 low level features.
2. For the accelerometer signal, our proposed model using the Fourier Transform feature has a better performance than the state of the art system [1] using "the rate of change in motion" ([6,19]) feature for accelerometer signal and also their fusion method.
3. Using the "Behavior Text" [20] feature suggested by Carnegie Mellon University group doesn't give us a better performance than using the traditional Fast Fourier Transform (FFT) feature for event segmentation under our proposed model.

2 Related Work

In one of the early studies, by using a mobile phone equipped with a sensor box, [21] investigated several time series segmentation methods for segmenting context data sequences into discrete, non-overlapping and internally homogeneous segments. A cost function is defined for any segments on the time series. Hence, the segmentation problem has been converted into an optimization problem. In their particular study, they define the segmentation cost as a sum of the variance of the components of the segment. However, there are two drawbacks here. First, they don't have any detailed ground truth boundaries to compare with their methods. In other words, they lack some systematic metric to evaluate the performance. Second, the number of events needs to be predefined and they don't show how to get the optimal number.

An algorithm based on a hidden Markov model (HMM) is proposed by [22] for unsupervised clustering of free-living human activities on accelerometry. This algorithm iteratively trains the HMM whose state is a sub-HMM with minimum duration constraint using the Expectation-Maximisation (EM) algorithm. The topology of the HMM changes during the cluster merging step with a merging criterion. However, this model suffers from two main limitations as mentioned in the paper [22]. First, the varied durations of different activities complicates the selection of features and hyper-parameters. Second, only one hour data sequences have been tested in the experiments instead of a full range of daily human activities.

By using the SenseCam[1], [6] investigated 5 different sources of information, which are low-level image descriptors, audio, temperature, light and accelerometer readings and their combinations to segment the SenseCam images into discrete events. Their method mainly involves two steps. The first step is called score assignment. Generally, the computation of the score involves the distance computation between every pair of contiguous windows which contains a fixed number of data units (e.g. images) and slides along the data sequence ordered by time. In this step, only the time break which has an image timestamp associated with it has a score. A high score will indicate an event boundary. Since different types of data may have different captured times, interpolation techniques are used to get the score for the image capture time. For example, a Gaussian window centred at the capture time of the images is used for the sensor values. For the audio data, a linear interpolation is applied. The second step is about score normalization and threshold. That after score normalization, the segmentation algorithm determines a threshold in order to get 20 events for each day. Generally, the time breaks of the 20 top scorings are considered to be the event boundaries.

In their follow up works, [1] introduces a performance metric by providing ground truth boundaries such that some images are selected as event boundaries. Their performance measure is the F1 score (a measure of a test's accuracy based on precision and recall) between ground truth boundaries and algorithm outputs. In this study, they focus on the segmentation of images in conjunction with accelerometer readings and suggest to use “the rate of change in motion” [6,19] feature for accelerometer data representation.

More recently, [18] propose a framework of the lifelog system by using a smart phone. They investigate the activities segmentation and activities recognition on data collected from 2 users wearing the phone for 5 days. They propose a novel method called “behaviour text” [24,20] to represent the sensory data through quantizing them. In this method, a K-means clustering algorithm is applied to the raw sensor data as the first step. Each sensor record is then assigned a unique symbol sequence presenting its nearest cluster. After converting the raw sensor data into this “behaviour text”, they proposed two different methods, namely, top-down activity segmentation through activity change detection for event segmentation and smoothed Hidden Markov Model (HMM) for activities segmentation and annotation. By average over all activity types, the authors find that the top-down activity segmentation approach performs better than the smoothed HMM [18].

3 Smart Phone and Data Collection

Several research groups [25,26,3,20,18] have developed personal Lifelog systems to capture personal experiences by wearing various sensors and a wearable

[1] SenseCam [23] is a small device that people can wear around their neck. It has a digital camera and multiple sensors equipped, including: a light sensor, a thermometer, an accelerometer to detect motion and a passive infra-red sensor to detect the presence of a person.

computer. However, most of them need multiple devices to be carried around user's body. In our study, only a smart phone is needed for data collection which makes it more comfortable for users and encourages more natural interactions with them.

Generally, an Android smart phone contains a variety of sensors including a 3-axis accelerometer, a 3-axis orientation sensor, an light sensor, a magnetic field sensor, a temperature sensor, a pressure sensor or even gyroscope sensor and gravity sensor. It also has the function to take images, videos and record audio. Furthermore, an Android phone can also track the user's location by using a GPS device when the user is outdoors.

In this study, we collect data from 2 subjects who use an Android phone to capture images, audio, GPS locations and some sensor data as they engage in their every day activities. The phone is worn around their neck and is positioned in a case with a strap as shown in Figure 1. They are free to turn the application off when they want to protect their privacy. However, they are instructed to provide at least 6-7 hours worth of data each day.

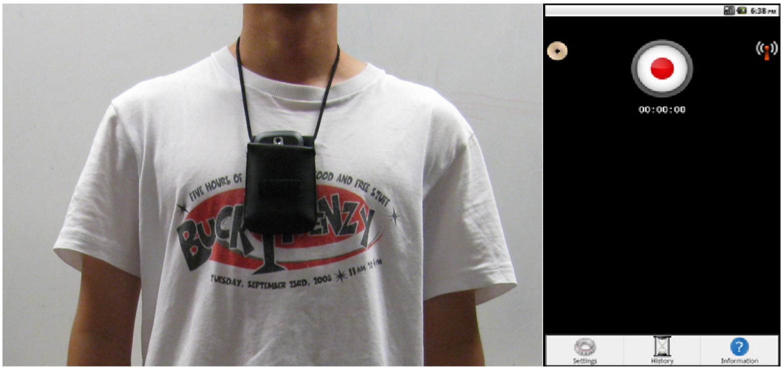

Fig. 1. Smart phone and software for collecting activity data

4 Metric-Based Event Segmentation

Generally, this model involves three steps, namely, "Feature Extraction", "Distance Metrics" and "Event Boundary Detection". The overall procedure for sensor data is depicted in Figure 2, the procedure for the image data is similar.

4.1 Feature Extraction

Sensor Data

1. Fast Fourier Transform (FFT) feature: The signal is first divided into a series of consecutive overlapping frames where each frame is a fragment of the signal - a fixed size of samples. At a sampling frequency of 15HZ, a sample

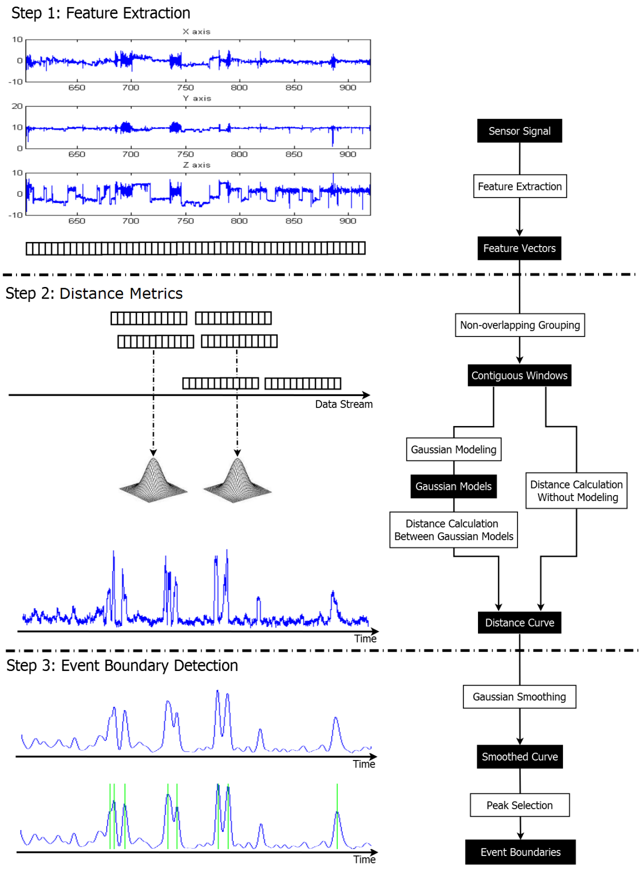

Fig. 2. Metric-based Event Segmentation for Sensor Data

size of 30 samples represents 2 seconds. FFT features are then extracted for each frame. In the experiment, the sample size is empirically set to 30 and the overlapping size is 10.

Suppose x is the input signal and y is the output features, the function implementing this transform is as follows,

$$y_k = \sum_{j=1}^{N} x_j \omega_N^{(j-1)(k-1)}$$

where $\omega_N = e^{(-2\pi i)/N}$ is an Nth root of unity and the length of x is N. And

$$Y_k = |y_k|$$

where $|\cdot|$ is the complex magnitude.

Since $Y_{\frac{n}{2}-k} = Y_{\frac{n}{2}+k}$, we can just keep half of the FFT features. Furthermore, the DC feature (the first one Y_1) is the total acceleration value of the signal over the window and is discarded in this study since from the experiment results we find excluding it can achieve a better performance. This (Excluding the DC component) is similar to how [27] handles activity recognition. Hence, the selected FFT features for each frame are $\{Y_k\}$ where $k \in \{2, 3, 4, \cdots, \frac{n}{2}\}$. Finally, since the accelerometer has three axis, the final feature vector is the concatenation of the FFT features for each axis.

2. Motion Change Rate (MCR) feature: "The rate of change of motion" is first introduced by [19] and suggested by [6,1] for accelerometer data captured by SenseCam on event segmentation.
3. "Behaviour Text" feature: This feature is suggested by [24,20,18] to represent the sensory data for event segmentation and activity recognition. A k-means clustering method is applied on the raw sensor data that results a sequence of symbols representing the cluster centres for each sample.

Image Data. The images are represented by MPEG-7 descriptors. The descriptors we select are following what [1] suggest which are: color layout, color structure, scalable color and edge histogram.

4.2 Distance Metrics

The distance metrics results in a curve of dissimilarity called "distance curve" with respect to time.

Sensor Data

1. For Fast Fourier Transform (FFT) feature: Feature vectors are grouped into a series of non-overlapping consecutive windows (sliding windows) whose size is fixed. A dissimilarity measure is then applied on pairwise sliding windows. The step length is one frame which means after the dissimilarity measure on two non-overlapping consecutive windows, we move both windows one frame in the direction of increasing time and compute the new dissimilarity and so on. In the experiment, we test different window sizes and show their corresponding performances.
   - For the **model-based** approach, a multivariate Gaussian distribution is applied for each window to describe the data. Several distance measurements between these two Gaussians are used to measure the dissimilarity of neighboring non-overlapping windows and the windows are then shifted by a fixed step (about 1 frame) along the whole signal. This process leads to the final distance curve. The following are several optional distance measurements:

     Kullback-Leibler diatance **(KL)**:

$$d_{KL} = \frac{1}{2}(\mu_1 - \mu_2)^T(\Sigma_1^{-1} + \Sigma_2^{-1})(\mu_1 - \mu_2) + \frac{1}{2}tr(\Sigma_1^{-1}\Sigma_2 + \Sigma_2^{-1}\Sigma_1 - 2I) \quad (1)$$

Bhattacharyya distance **(BHA)**:

$$d_{BHA} = \frac{1}{4}(\mu_1 - \mu_2)^T(\Sigma_1 + \Sigma_2)^{-1}(\mu_1 - \mu_2) + \frac{1}{2}log\frac{|\Sigma_1 + \Sigma_2|}{2\sqrt{|\Sigma_1||\Sigma_2|}} \quad (2)$$

Bayesian Information Criterion **(BIC)**[28]:

$$d_{BIC} = Nlog|\Sigma| - N_1log|\Sigma_1| - N_2log|\Sigma_2| - \lambda P \quad (3)$$

where Σ is the sample covariance matrix from samples of two windows and $P = \frac{1}{2}(d + \frac{1}{2}d(d+1))logN$. d is the dimension of the space, the penalty weight λ is equal to 1, N is the total number of samples in two windows and μ is the mean vector and I is identity matrix.

- For the **non-model-based** approach, we introduce the following method: Average Euclidean Distance **(AED)**:

$$d_{AED} = \frac{1}{|A||B|}\sum_{u\in A, v\in B} dist(u,v) \quad (4)$$

where $dist(u,v)$ is a Euclidean distance between vector u and vector v. Set A and B represent two sliding windows and $|\cdot|$ denotes the cardinality.
Mean Vector Distance **(MVD)**:

$$d_{MVD} = dist(\mu_1, \mu_2) \quad (5)$$

where $dist(\mu_1, \mu_2)$ is a Euclidean distance between the mean vector μ_1 and μ_2 for sliding windows.

2. For the Motion Change Rate (MCR) feature: In order to compare the system performance suggested by [6,1], the sensor motion values are associated with an image using a Gaussian window centred at the time the image is captured. The distance curve is then formed from a series of motion values where large motion values indicate event boundaries. A Min-Max normalisation technique [29] is applied followed by the event boundary detection [1]. In the experiment, we test different Gaussian window widths and show their corresponding performances.
3. For the "Behaviour Text" feature: Behaviour text string is grouped into a series of non-overlapping consecutive windows (sliding windows) whose size is fixed. A dissimilarity measure [20,18] is then applied on pairwise sliding windows. In the experiment, we test different window sizes and show their corresponding performances.

Image Data. Suggested by [1], images are grouped into a series of non-overlapping consecutive windows (sliding windows) whose size is fixed. An "average image representation" is derived for each window, histogram intersection is then applied on pairwise "average image representation". Since the histogram intersection results a similarity, the dissimilarity is derived by subtracting the similarity from 1. In the experiment, we test different window sizes (different number of images) and show their corresponding performances.

4.3 Event Boundary Detection

This step involves two sub steps, namely “Smoothing” and “Peak Selection”.

Smoothing. Since the event boundaries are “fuzzy” [12] and the distance curve may contain noise and events may have different levels of granularity or scale, it is necessary to smooth the curve to identify robust event boundaries that are invariant with respect to granularity or scaling, and are minimally affected by noise and small distortions. Suggested by [30], Gaussian kernel is used to handle these considerations. The following is the description:

To smooth the curve by the convolution of a variable-scale Gaussian, $G(t, \sigma)$, with an input curve, $I(t)$:

$$L(t, \sigma) = G(t, \sigma) * I(t), \tag{6}$$

where $*$ is the convolution operation in t, and

$$G(t, \sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{t^2}{2\sigma^2}} \tag{7}$$

The smoothing level for Gaussian kernel is denoted by parameter σ. In the experiment, the σ for cross validation is from $\{1, 2, 3, ..., 90\}$.

Peak Selection. Three different peak selection models are proposed in this section, namely, “All peak”, “Tall peak” and “Significant peak”. In the experiments, their corresponding free parameters are selected by cross validations.

Firstly, for the “All peak”, this model simply selects all the peaks (potentional boundaries) to form the final event boundaries. Since peaks can be selected from different smoothing levels, there is only one free parameter in this model - the smoothing level σ in “Gaussian Convolution”.

Secondly, a peak is “tall” when its height is greater than some *threshold* as Figure 3 depicts. In this model, all the boundaries corresponding to tall peaks are selected as the final event boundaries. In this model, besides the σ in “Gaussian Convolution”, there is another free parameter - the *threshold*. In the experiment, the σ we select for cross validation is from $\{1, 2, 3, ..., 90\}$ and the *threshold* is $i \times \frac{(max-min)}{50} + min$ where $i \in \{1, 2, 3, ..., 50\}$ and max and min are the maximum and minimum values for all the distance curves.

Finally, a peak is “significant” [31] if

$$\begin{aligned} |d(max) - d(min_{left})| &> \alpha\sigma' \\ or\ |d(max) - d(min_{right})| &> \alpha\sigma' \end{aligned} \tag{8}$$

where α is a parameter, σ' is the standard deviation of the distance curve. And min_{left} and min_{right} are the left and right minimas around the peak “max” as Figure 4 depicts.

This model selects all the “significant peak”. The boundaries associated to these peaks are selected to form the final event boundaries. This model involves two free parameters, σ in “Gaussian Convolution” and α in detecting the “significant peak”. In the experiment, the α we selected for cross validation is from $\{0.1, 0.2, 0.3, ..., 2.0\}$ and the σ for cross validation is from $\{1, 2, 3, ..., 90\}$.

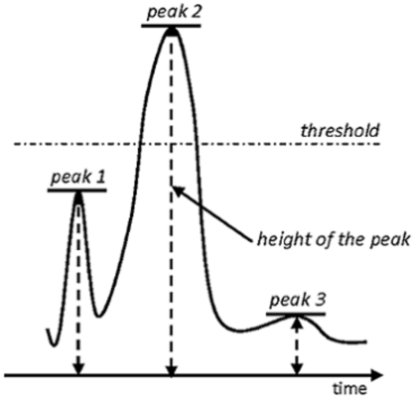

Fig. 3. Tall Peak: Peak 2 is tall peak since its height is greater than the threshold

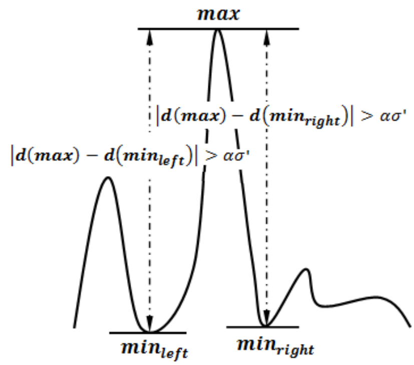
max
$|d(max) - d(min_{right})| > \alpha\sigma'$
$|d(max) - d(min_{left})| > \alpha\sigma'$
min_{left}
min_{right}

Fig. 4. Significant Peak: The middle peak is "Significant"

5 Segmentation Quality Measure

5.1 Evaluation Reference

In order to evaluate performance, subjects manually segment their daily experience into different events by viewing their corresponding sequential images. The timestamps of the images they choose are marked as ground truth boundaries. Hence, the ground truth event boundaries for daily experience is a set of timestamps.

5.2 Evaluation Metrics

An event segmentation system tries to detect changes in the sensor signals where the changes correspond to boundaries of different events. Such a system may have two possible types of error. $Type-1$ errors occur if a true boundary is not hit within a certain range in time (3 minutes either side in our case). $Type-2$ errors occur if a detected change does not correspond to any ground truth boundary (false alarm). Type 1 and 2 errors are also referred to as precision (PRC) and recall (RCL), respectively [32].

Let N_{hit} be the number of boundaries correctly detected (hit), N_{ref} be the total number of boundaries in the reference and N_f be the number of detected boundaries (system outputs).

The precision (PRC) and recall (RCL) can be defined as follows:

$$PRC = \frac{N_{hit}}{N_f}, RCL = \frac{N_{hit}}{N_{ref}} \tag{9}$$

Generally, the F-measure is often used to compare the performance of different algorithms as follows:

$$F = \frac{2 \times PRC \times RCL}{PRC + RCL} \tag{10}$$

The range of the F-measure is from 0 to 1 where a higher F-measure indicates a better performance.

5.3 Hits Counting: Search Region

In order to determine the number of hits, a fixed-size search region around each reference boundary is placed and verified whether there are some boundaries produced by a segmentation algorithm in these regions as Figure 5 shows.

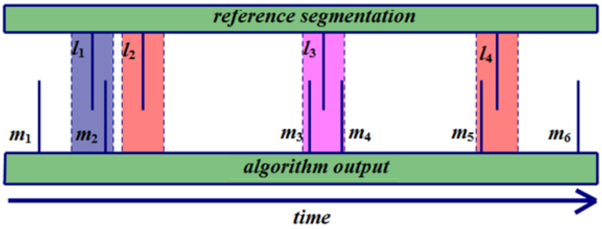

Fig. 5. In this example, for reference boundary l_1, there is an algorithm output m_2 in its search range (colored region with dotted lines as its both sides), hence, boundary l_1 is hit by output m_2. For boundary l_3, there are two outputs in its search range, and l_3 is also hit. The number of boundaries correctly detected is 3 in this case (N_{hit}=3). And the total number of boundaries in the reference is 4 (N_{ref}=4). Total number of detected boundaries (algorithm outputs) is 6 (N_f=6).

However, if there is some overlapping search region, this will cause an ambiguous situation in evaluation. This problem can be solved by removing the overlapping region by asymmetrically shrinking the search regions of its two sides to a common mid-point [33] (see Figure 6).

In the experiments, since we don't study segmentation of the events which last less than 3 minutes, the total search region is set to 6 minutes.

6 Experiments

6.1 Dataset

Sensor Data. Unlike [6] whose sensor data is captured every 2 seconds (0.5HZ), the sampling rate of our smart phone sensor is from 15HZ to 20HZ. This sample rate is sufficient for detecting human daily physical activity [34]. Since the **raw accelerometer** data is recorded using the **device's coordinate system**, we convert it into **world's coordinate system** (see Figure 7) by eliminating the force of the gravity and with the assistance of the magnetometer sensor. We call this converted data (which is also gravity eliminated) **adjusted accelerometer** data. In the experiments, we investigate the performance difference between these two different representations by using different distance measures and different models.

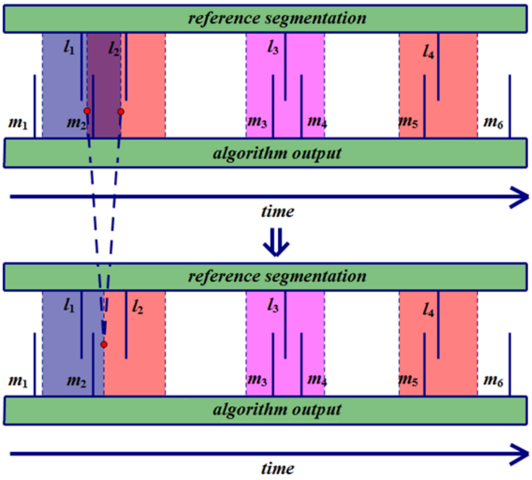

Fig. 6. The first plot shows an example of an overlapping search region causing a ambiguous situation in evaluation, namely a problem of how to define a matching boundary for each reference boundary. It is not clear whether l_1 or l_2 is hit by m_2 or both. The second plot removes the overlapping by asymmetrically shrinking the search regions of its two sides to a common mid-point. Hence, the matching becomes straightforward so that m_2 is only contributed to the hit of l_1.

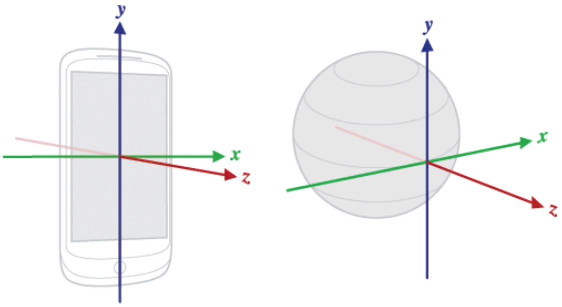

Fig. 7. The left figure indicates the device's coordinate system and the right figure represents the world's coordinate

Ground Truth Boundaries. In this experiment, we ask the wearers to manually create a ground truth of segmentations for all his/her own recorded images. There are several reasons that we ask the wearers to segment their own records. First, event segmentation has subjective and individual differences and is hard to characterize by normative criteria [9]. Second, we believe that the wearers have the best knowledge of their own intentions and goals for what they did when viewing their own image records. Third, considering the privacy issues with data highly personal to the user, it is desirable to mark event boundaries according to each wearers' own judgements suggested by [1].

6.2 Experimental Setup

In the experiments, the data set is first grouped into continuous chunks in time. Then all the chunks are divided into training chunks and test chunks. Each chunk represents records of activities per person per day. We leave one chunk out to select the parameters and test it on the test chunk. By grouping the data into chunks during the testing, this can guarantee that the models are tested on completely different days.

6.3 Understanding of the Performance Measure

In order to have a better intuition of the performance measure - F values, we do some experiments based on random boundary generation and ground truth boundary replacement.

Random Boundary Generation. Without knowing the number of event boundaries for each chunk, we use a fixed number to generate the random boundaries from a uniform distribution. Fig 8 shows the results, the number of boundaries range from 1 to 200 for all chunks. The F values come from the average results on 100 simulations (random boundary generation). Furthermore, knowing the number of event boundaries for each chunk, we can get a F value that is 0.22 ± 0.026, where 0.22 is the average and 0.026 is the standard deviation. The average suggests a "chance segmentation" and can be counted as a baseline.

Ground Truth Boundary Replacement. In this experiment, different proportion of ground truth boundaries are replaced by equal number of random boundaries with 100 different simulations. Fig 9 shows the results. For example, a F value of 0.65 indicates that nearly 38% of the ground truth boundaries are placed by random boundaries. 0.60 indicates 44%, 0.55 indicates 51% and 0.5 indicates 59%.

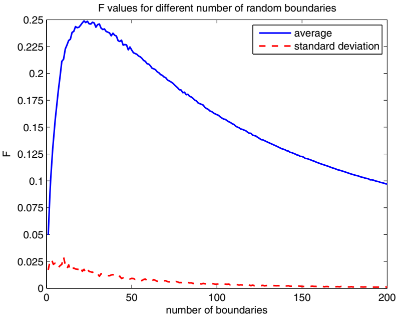

Fig. 8. F values for different number of boundaries

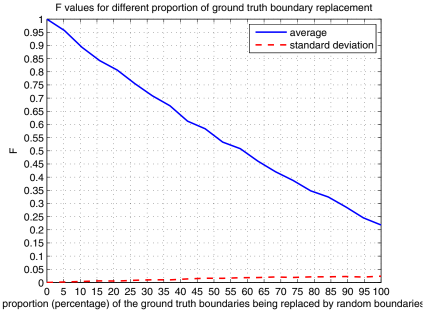

Fig. 9. F values for different proportion of ground truth boundary replacement

6.4 Experimental Results and Analysis

In the experiments, we are going to answer the follows: 1. How well can our proposed model do comparing with other systems on the event segmentation? 2. Which modality is better? Image data or accelerometer data? 3. What is the impact for movement features for event segmentation on long time records?

1. Our Proposed Models: Fig 10 and Fig 11 show the results. By using the "Tall Peak Detection" on adjusted accelerometer, the "*BHA*" and "*BIC*" distance measures can give us the best average F value around 0.65 using the window size in the range from 50 frames (equivalent to 75 seconds) to 90 frames (equaling 135 seconds) [2].

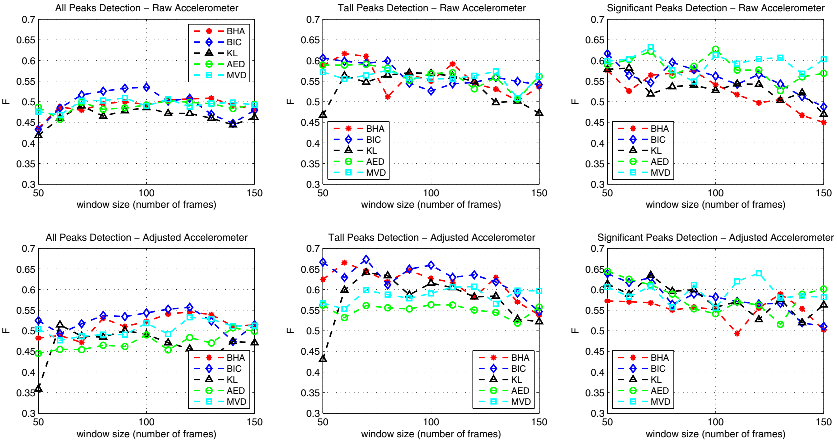

Fig. 10. Event segmentation on accelerometer data using our proposed model

The "Histogram Intersection" distance suggested by [35] is used in the distance curve computation for image signals with different representations. By the comparison of the event segmentation results between using accelerometer data (Fig 10) and image data (Fig 11), it is clear, the event segmentation performance from using accelerometer data is much higher than using image data.

2. Dublin City University's System: The Motion Change Rate (MCR) feature is used in this system. Since the performance measures are different, in order to make a fair comparison, the distance curves are processed with/without "Peak Scoring" technique before peak detection. Furthermore, a data normalisation method "Sum" is used for different signals before fusion. A fusion

[2] Although different models would prefer different window sizes, we would like to report the average F value in a reasonably good range instead of reporting the best one. Since the best one associated with a particular window size may be sensitive to the data and model.

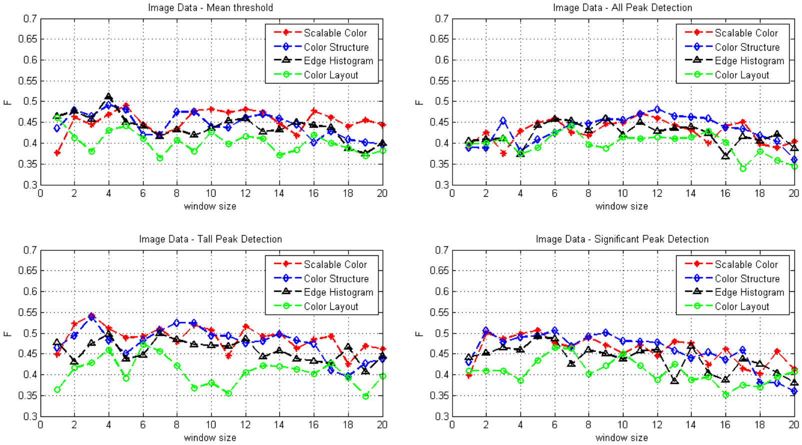

Fig. 11. Event segmentation on image data using our proposed model

method "CombMIN" suggested by [1] is used for different signals. For this fusion method, different distance curves are multiplied by different weights. For each time step, the minimal among these weighted distance curves is selected as the output. The weights we used in the fusion are suggested by [35].

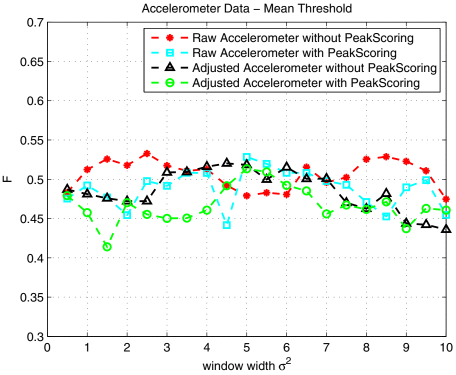

Fig. 12. Event segmentation using Dublin City University's System - Accelerometer Data Only

Fig 12 shows the results for accelerometer data using the "Mean Threshold" method suggested by [1] for peak detection. The best performance using the accelerometer data is around 0.53.

The early fusion method is suggested by [35] (left panel in Fig 13) so that different image representations are concatenated into a signal representation,

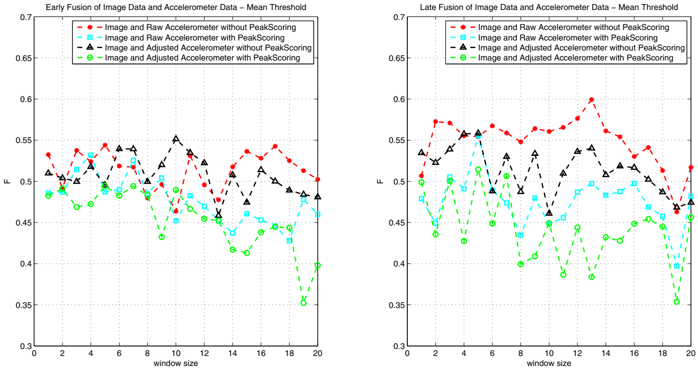

Fig. 13. Event segmentation using Dublin City University's System - Fusion

then a distance curve computed on that representation is fused with distance curve of the accelerometer signal later. The weight for the image distance curve is 0.65 and the weight for the accelerometer distance curve is 0.35. Furthermore, we suggested a completely late fusion method (right panel in Fig 13) where this fusion is on distance curves of different image representations and distance curves of accelerometer signal with equal weights. And it is clear, when using raw accelerometer data without "peak scoring", this method can achieve an F score which is around 0.58. Finally, with a comparison between Fig 13 and Fig 12, the fusion of image and accelerometer data can give us a better result than using accelerometer data only.

3. Carnegie Mellon University's system: The "Behaviour Text" feature is used in this system. In this experiment, in order to make a fair comparison[3] for the study of the "Behaviour Text" feature for event segmentation, we use our proposed peak detection methods to find the event boundaries. The hyper parameters for the feature extraction and dissimilarity measure are set empirically [24,20] according to the experimental results. Fig 14 shows the results. Using "Significant Peak Detection" on adjusted accelerometer data, the best average F values are around 0.60 whose window sizes are from 1 minute length to 2 minute length. And it is clear that the performances using adjusted accelerometer data are much better than the performances using raw accelerometer data.

[3] The papers [24,20] lack enough details for replicating their hierarchical segmentation method and through some email contacts with the main author, our attempt for replicating their hierarchical segmentation method still gives us some low performance.

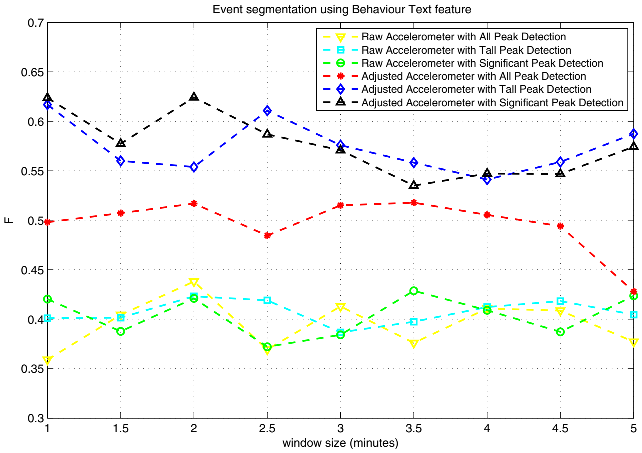

Fig. 14. Event segmentation using "Behaviour Text" feature

7 Conclusion and Discussion

The automatic event segmentation using accelerometer data appears to be more reliable than that using image data. This was somewhat unexpected considering the ground-truth boundaries were created by subjects from imagery. The main reason may be due to the fact that the camera can capture a totally different image signal with slightly different device orientation even within a similar context. Humans can understand the semantic meaning of these images and recognize that they have a similar context even though they may look very different. However, the distance measure using MPEG-7 low level features will tell the difference in vision and the metric based model with these features fails to understand the similarity in context.

Second, according to the comparison between our proposed model and the Dublin City University System [1] on event segmentation using accelerometer data, we find it is necessary to have a high sampling rate to capture the change of daily human activity. One reason that our proposed model can have a higher performance than the Dublin City University System may be due to their low sampling rate - sensor data is captured every 2 seconds (0.5HZ). We think such low sampling rate may be insufficient for detecting daily human physical activity [34] and may fail to detect some event boundaries.

Third, our proposed peak selection methods using the bag of feature representation such as "Behavior Text" [20] suggested by Carnegie Mellon University group has a much better performance on the adjusted accelerometer data than the raw accelerometer data. Furthermore, the best performance from our proposed model is also from the adjusted accelerometer data. All these may suggest that, for the accelerometer data, the gravity impact combined with the orientation of the phone (implied by the accelerometer data based on a device's local coordinate system) is useless or even harmful for event segmentation.

Since the gravity can be decomposed as adding numbers into the three axis of the accelerometer data, they vary with different phone's positions. This usually overwhelm the acceleration introduced by activity change and make what the algorithm detected majorly becomes the change of the device's position.

Finally, we believe that the movement features can play an important role for event segmentation on the long time records.

Acknowledgment. This work was supported by the Air Force Office of Scientific Research under AFOSR FA9550-09-1-0614 and National Science Foundation under IIS-1117707.

References

1. Doherty, A.R., Smeaton, A.F.: Automatically Segmenting Lifelog Data into Events. In: WIAMIS 2008 - 9th International Workshop on Image Analysis for Multimedia Interactive Services (2008)
2. Doherty, A.R., Byrne, D., Smeaton, A.F., Jones, G.J.F., Hughes, M.: Investigating Keyframe Selection Methods in the Novel Domain of Passively Captured Visual Lifelogs. In: CIVR 2008: Proceedings of the 2008 International Conference on Content-based Image and Video Retrieval, Niagara Falls, pp. 259–268. ACM (2008)
3. Takata, K., Ma, J., Apduhan, B.O., Huang, R., Jin, Q.: Modeling and Analyzing Individuals Daily Activities Using Lifelog. In: Second International Conference on Embedded Software and Systems, pp. 503–510. IEEE Computer Society, Los Alamitos (2008)
4. Swallow, K.M., Zacks, J.M., Abrams, R.A.: Event Boundaries in Perception Affect Memory Encoding and Updating. Journal of Experimental Psychology. General 138(2), 236–257 (2009)
5. Zacks, J.M., Swallow, K.M.: Event Segmentation. Current Directions in Psychological Science 16(2), 80–84 (2007)
6. Doherty, A.R., Smeaton, A.F., Lee, K., Ellis, D.P.W.: Multimodal Segmentation of Lifelog Data. In: Proc. of RIAO 2007 (2007)
7. Nicole, J.M.Z., Speer, K., Swallow, K.M.: Activation of Human Motion Processing Areas During Event Perception. Cogn. Affect. Behav. Neurosci. 3, 335–345 (2003)
8. Chen, C.: Information Visualisation and Virtual Environments. Springer, London (1999)
9. Zacks, J.M., Braver, T.S., Sheridan, M.A., Donaldson, D.I., Snyder, A.Z., Ollinger, J.M., Buckner, R.L., Raichle, M.E.: Human Brain Activity Time-locked to Perceptual Event Boundaries. Nat. Neurosci. 4(6), 651–655 (2001)
10. Zacks, J., Tversky, B., Iyer, G.: Perceiving, Remembering, and Communicating Structure in Events. Journal of Experimental Psychology: General 130, 29–58 (2001)
11. Zacks, J.M., Swallow, K.M., Vettel, J.M., McAvoy, M.P.: Visual Motion and the Neural Correlates of Event Perception. Brain Research 1076(1), 150–162 (2006)
12. Zacks, J.M., Speer, N.K., Swallow, K.M., Braver, T.S., Reynolds, J.R.: Event perception: A mind-brain perspective. Psychological Bulletin 133, 273–293 (2007)
13. Zacks, J.M., Speer, N.K., Vettel, J.M., Jacoby, L.L.: Event Understanding and Memory in Healthy Aging and Dementia of the Alzheimer Type. Psychology and Aging 21(3), 466–482 (2006)

14. Newtson, D., Engquist, G.: Foundations of attribution: The Perceptual Organization of Ongoing Behavior. Journal of Experimental Social Psychology 12(5), 436–450 (1976)
15. Newtson, D.: Foundations of attribution: The perception of ongoing behavior. New Directions in Attribution Research, 223–248 (1976)
16. Zacks, J.M., Kumar, S., Abrams, R.A., Mehta, R.: Using Movement and Intentions to Understand Human Activity. Cognition 112(2), 201–216 (2009)
17. Newtson, D., Engquist, G.A., Bois, J.: The Objective Basis of Behavior Units. Journal of Personality and Social Psychology 35(12), 847–862 (1977)
18. Wu, P., Peng, H.-K., Zhu, J., Zhang, Y.: SensCare: Semi-automatic Activity Summarization System for Elderly Care. In: Zhang, J.Y., Wilkiewicz, J., Nahapetian, A. (eds.) MobiCASE 2011. LNICST, vol. 95, pp. 1–19. Springer, Heidelberg (2012)
19. ÓConaire, C., O'Connor, N.E., Smeaton, A.F., Jones, G.J.F.: Organising a Daily Visual Diary Using Multifeature Clustering. Optical Society of America (2007)
20. Chennuru, S., Chen, P.-W., Zhu, J., Zhang, J.Y.: Mobile Lifelogger – Recording, Indexing, and Understanding a Mobile User's Life. In: Gris, M., Yang, G. (eds.) MobiCASE 2010. LNICST, vol. 76, pp. 263–281. Springer, Heidelberg (2012)
21. Himberg, J., Korpiaho, K., Mannila, H., Tikanmäki, J., Toivonen, H.T.: Time Series Segmentation for Context Recognition in Mobile Devices, pp. 203–210 (2001)
22. Nguyen, A., Moore, D., McCowan, I.: Unsupervised Clustering of Free-living Human Activities Using Ambulatory Accelerometry. In: 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS 2007, pp. 4895–4898. IEEE (2007)
23. Hodges, S., Williams, L., Berry, E., Izadi, S., Srinivasan, J., Butler, A., Smyth, G., Kapur, N., Wood, K.: SenseCam: A Retrospective Memory Aid. In: Dourish, P., Friday, A. (eds.) UbiComp 2006. LNCS, vol. 4206, pp. 177–193. Springer, Heidelberg (2006)
24. Chen, P., Chennuru, S., Zhang, Y.: A Language Approach to Modeling Human Behavior. In: Proc. Seventh Intl. Conf. Language Resources and Evaluation (2010)
25. Hori, T., Aizawa, K.: Context-based Video Retrieval System for the Life-log Applications. In: Proceedings of the 5th ACM SIGMM International Workshop on Multimedia Information Retrieval, MIR 2003, New York, NY, USA, pp. 31–38 (2003)
26. Kim, I.-J., Ahn, S.C., Ko, H., Kim, H.-G.: Automatic Lifelog Media Annotation based on Heterogeneous Sensor Fusion. In: IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems, MFI 2008, pp. 703–708 (August 2008)
27. Bao, L., Intille, S.S.: Activity Recognition from User-annotated Acceleration Data, pp. 1–17. Springer (2004)
28. Chen, S.S., Gopalakrishnan, P.S.: Speaker, Environment and Channel Change Detection and Clustering via the Bayesian Information Criterion, pp. 127–132 (1998)
29. Montague, M., Aslam, J.: Relevance Score Normalization for Metasearch. In: Proceedings of the Tenth International Conference on Information and Knowledge Management, pp. 427–433. ACM (2001)
30. Lindeberg, T.: Scale-space theory: A Basic Tool for Analysing Structures at Different Scales. Journal of Applied Statistics, 224–270 (1994)
31. Zochová, P., Radová, V.: Modified Distbic Algorithm for Speaker Change Detection. In: Eurospeech, pp 1:3073–1:3076 (2005)
32. Ajmera, J., Mccowan, I., Bourlard, H.: Robust Speaker Change Detection. IEEE Signal Process. Lett. 11, 649–651 (2004)

33. Räsänen, O.J., Laine, U.K., Altosaar, T.: An Improved Speech Segmentation Quality Measure: the R-Value. In: INTERSPEECH, pp. 1851–1854 (September 2009)
34. Bouten, C., Koekkoek, K., Verduin, M., Kodde, R., Janssen, J.: A Triaxial Accelerometer and Portable Data Processing Unit for the Assessment of Daily Physical Activity. IEEE Transactions on Biomedical Engineering 44(3), 136–147 (1997)
35. Doherty, A.: Providing Effective Memory Retrieval Cues through Automatic Structuring and Augmentation of a Lifelog of Images (2008)