METRIC BASED AUTOMATIC EVENT SEGMENTATION AND NETWORK PROPERTIES OF EXPERIENCE GRAPHS

THESIS

Presented in Partial Fulfillment of the Requirements for the Degree Master of Science in the Graduate School of The Ohio State University

By

Yuwen Zhuang

Graduate Program in Computer Science and Engineering

The Ohio State University

2012

Master’s Examination Committee:

Mikhail Belkin, Advisor
Simon Dennis, Co-Advisor
Deliang Wang, Committee Member
ABSTRACT

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. In this thesis, we collect lifelog data by using a mobile phone or a Microsoft Research SenseCam worn around subjects’ necks during their daily life. We then propose a way to organize the lifelog data - a metric-based model for event segmentation. Further more, we analyse the data properties through constructing the experience graphs from the recorded images. This thesis involves two parts, the details are as follows:

First we describe 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 work 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 [13].

Second, we investigate data obtained from subjects wearing a Microsoft Research SenseCam as they engaged in their every day activities. We construct experience graphs for each subject from their corresponding images by using two different image representation methods - color histogram and color correlogram. The statistical analyses of these graphs show that they have a small-world structure which is characterized by high proximity ratios and sparse connectivity, independent of the representation used. However, the degree distribution analyses show that they are not scale-free, broad-scale or even single-scale that [3] studied. Furthermore, we also find that the graphs constructed based on the color correlogram representation, which is better than the color histogram in many content-based image retrieval systems [25], have shorter average path lengths and higher global clustering coefficients than the graphs constructed based on the color histogram representation.

**Keywords:** Triaxial accelerometer; Event segmentation; Lifelog data; Microsoft Research SenseCam; Experience Graphs; Color Histogram; Color Correlogram; Small World; Proximity Ratio; Scale-Free; Global Clustering Coefficient;
This document is dedicated to my family.
ACKNOWLEDGEMENTS

This work was supported by the Air Force Office of Scientific Research under grant number AFOSR FA9550-09-1-0614. We also thank Microsoft for providing us the Sense-Cam devices.
VITA

2007 ................................. Bachelor Degree in Software Engineering

2009-2011 ........................... Graduate Research Associate in The Ohio State University

2011-2012 ........................... Graduate Teaching Associate in The Ohio State University

WORKS AND PUBLICATIONS


Yuwen Zhuang, Vishnu Sreekumar, Mikhail Belkin, Simon J.Dennis, ”The network properties of episodic graphs”

Yuwen Zhuang, Vishnu Sreekumar, Mikhail Belkin, Simon J.Dennis, ”Metric Based Automatic Event Segmentation”
FIELDS OF STUDY

Major Field: Computer Science and Engineering

Specialization: Machine Learning and Lifelog Data Analysis
# TABLE OF CONTENTS

Abstract ................................................................. ii
Dedication ............................................................... iii
Acknowledgements ...................................................... v
Vita ........................................................................... vi
List of Figures ............................................................. x
List of Tables ............................................................. xii

<table>
<thead>
<tr>
<th>CHAPTER</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Introduction and Background</td>
</tr>
<tr>
<td>1.1</td>
<td>Metric Based Automatic Event Segmentation</td>
</tr>
<tr>
<td>1.1.1</td>
<td>Lifelog</td>
</tr>
<tr>
<td>1.1.2</td>
<td>The Reason For Event Segmentation</td>
</tr>
<tr>
<td>1.1.3</td>
<td>Questions For Event Segmentation</td>
</tr>
<tr>
<td>1.2</td>
<td>The Network Properties Of Experience Graphs</td>
</tr>
<tr>
<td>1.3</td>
<td>Contributions of Thesis</td>
</tr>
<tr>
<td>1.3.1</td>
<td>Automatic Event Segmentation</td>
</tr>
<tr>
<td>1.3.2</td>
<td>Network Properties Of Experience Graphs</td>
</tr>
<tr>
<td>2</td>
<td>Metric Based Automatic Event Segmentation</td>
</tr>
<tr>
<td>2.1</td>
<td>Related Work</td>
</tr>
<tr>
<td>2.2</td>
<td>Smart Phone and Data Collection</td>
</tr>
<tr>
<td>2.3</td>
<td>Metric-based Event Segmentation</td>
</tr>
<tr>
<td>2.3.1</td>
<td>Feature Extraction</td>
</tr>
<tr>
<td>2.3.2</td>
<td>Dissimilarity Measure</td>
</tr>
<tr>
<td>2.3.3</td>
<td>Event Boundary Detection</td>
</tr>
<tr>
<td>2.4</td>
<td>Segmentation Quality Measure</td>
</tr>
<tr>
<td>2.4.1</td>
<td>Evaluation Reference</td>
</tr>
<tr>
<td>2.4.2</td>
<td>Quality Measure</td>
</tr>
</tbody>
</table>
## LIST OF FIGURES

<table>
<thead>
<tr>
<th>FIGURE</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Smart phone and software for collecting activity data.</td>
</tr>
<tr>
<td>2.2</td>
<td>Metric-based Event Segmentation for Sensor Data</td>
</tr>
<tr>
<td>2.3</td>
<td>Tall Peak: In this example, peak 2 is tall peak since its height is greater than the threshold.</td>
</tr>
<tr>
<td>2.4</td>
<td>Significant Peak</td>
</tr>
<tr>
<td>2.5</td>
<td>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 ($N_f=6$).</td>
</tr>
<tr>
<td>2.6</td>
<td>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 $l_1$.</td>
</tr>
<tr>
<td>2.7</td>
<td>The left figure indicates the device’s coordinate system and the right figure represents the world’s coordinate.</td>
</tr>
<tr>
<td>2.8</td>
<td>F values for different number of boundaries</td>
</tr>
<tr>
<td>2.9</td>
<td>F values for different proportion of ground truth boundary replacement</td>
</tr>
<tr>
<td>2.10</td>
<td>Event segmentation on accelerometer data using our proposed model</td>
</tr>
</tbody>
</table>
2.11 Event segmentation on image data using our proposed model . . . . . . . . 26
2.12 Event segmentation using Dublin City University’s System - Accelerome-
ter Data Only . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 27
2.13 Event segmentation using Dublin City University’s System - Fusion . . . 27
2.14 Event segmentation using “Behaviour Text” feature . . . . . . . . . . . . 29
3.1 The means of Global Clustering Coefficients, Average shortest path lengths
and diameters for all subjects . . . . . . . . . . . . . . . . . . . . . . . . . . 39
3.2 Log-log plots of the cumulative degree distributions for experience graphs. 41
3.3 Linear-log plots of the cumulative degree distributions for experience graphs. 42
LIST OF TABLES

<table>
<thead>
<tr>
<th>TABLE</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Characteristic path lengths, clustering coefficients, and proximity ratios for graphs studied in [48] with a small world topology. In this table, the “film actors” corresponds to a collaboration graph of film actors, “power grid” corresponds to the power grid of the western United States and “C.elegans” is the neural network of the worm Caenorhabditis elegans [48].</td>
</tr>
<tr>
<td>3.2</td>
<td>Proximity Ratios</td>
</tr>
</tbody>
</table>
CHAPTER 1
INTRODUCTION AND BACKGROUND

1.1 Metric Based Automatic Event Segmentation

1.1.1 Lifelog

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 [12], 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 [46].

1.1.2 The Reason For Event Segmentation

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 [45]. 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 [54]. Recently, event segmentation [14, 13] 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.

### 1.1.3 Questions For Event Segmentation

There are several psychologic foundations for using the event segmentation as a methodology for organizing lifelogs. First, supported by behavioral [33] and neuro-imaging data [8, 50], it has been suggested that event segmentation is automatic [49, 55], even as observers passively view activities. Second, segmenting activity into meaningful events is indicated as a core component of perception [52] and has consequences for memory and learning [53, 30]. For example, first, [54] 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 interpretation.

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** [52] indicating there is inevitable variability when and where a boundary occurs, they are **remarkably consistent** across participants [29] with **good reliability** among the same participants in test-retest [33].

Second, what kind of features are needed? [54] show that event boundaries can be identified by tracking significant changes in physical and social features. [52] 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 [52] 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 time-scales for everyday events can vary from a few seconds to tens of minutes, previous studies [49, 51] suggest that movement features play a stronger role on fine grained [31] events (the smallest meaningful units) than coarse grained [31] events (the largest meaningful units) while coarse grained events are more dependent on conceptual features such as goals and causes. Hence, the final question is - is it the case that only fine-grained event boundaries are segmented using movement features? Or what is the impact for movement features on coarse-grained event boundaries?

1.2 The Network Properties Of Experience Graphs

Real world networks such as the Internet, the World Wide Web, and social and biological networks of various kinds have been the subject of intense study in recent years. From physics and computer science to biology, social sciences and even cognitive science, researchers have found that a great variety of systems can be represented as networks.

Small-world networks, as potential models of complex systems, is the focus of recent interest [48, 47, 28]. The network topology of a semantic network, which represents semantic relations among concepts and is often used as a form of knowledge representation, has been studied in detail [43]. Three different types of semantic networks: word associations, WordNet [26], and Rogets Thesaurus are shown to have a small-world structure characterized by sparse connectivity, short average path lengths between words, and strong local clustering [43].

For modeling semantic knowledge and inference, a semantic network is an organization of abstracted items instead of concrete realities, specific objects, or actual instances which are highly spatial-temporal related. However, it would be valuable to investigate human experiences which are also highly spatial-temporal related.
Furthermore, human dynamics is an emerging field of research interest [38, 41, 7]. There are many methods tried to model human activity. For example, [41] show the limits of predictability in human dynamics by studying 3-month-long location record during the mobile phone usage. [7] use Lévy-walk model to describing human mobility. [17] study stochastic queuing models for capturing human communication patterns and etc. Hence, it would be also interesting to investigate human activity pattern by studying human experience.

1.3 Contributions of Thesis

1.3.1 Automatic Event Segmentation

In this thesis, our study on natural and realistic daily event segmentation shows that, for long daily event segmentation (events that last at least 3 minutes and generally more than 10 minutes), even for coarse-grained events, the movement features can 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 [13] 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 [13] 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 [52] 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 [13]
which evaluates F score on images and [36] which evaluates F score on events (activities). More specifically, the contributions of this thesis are the following:

1. We show that using accelerometer signals can provide much more reliable and significantly better performance than using images signal with MPEG-7 low level features.

2. The experiment results suggest that for the accelerometer signal, our proposed model using the Fourier Transform feature has a better performance than the state of the art system [13] using “the rate of change in motion” [14, 34] feature for accelerometer signal and also their fusion method. Therefore, the model we suggested has a better performance than the model proposed by [14] for the segmentation of lifelog data.

3. Our proposed peak selection methods using the bag of feature representation such as “Behavior Text” [40] suggested by Carnegie Mellon University group has a much better performance on accelerometer data based on world’s (global) coordinate system than the accelerometer data based on a device’s local coordinate system. And using the “Behavior Text” feature doesn’t give a better performance than using the traditional Fast Fourier Transform (FFT) feature for event segmentation under our proposed model.

1.3.2 Network Properties Of Experience Graphs

We collect data from subjects and construct experience graphs for each subject then analyze the network properties of these graphs. A brief overview of how we do it is the following: We have subjects wear a SenseCam device that records images of every 10 seconds for an average of 6-7 hours a day for about a week. In order to capture the similarity relation between any two images from the data, we introduce two content-based image retrieval (CBIR) techniques - color histogram and color correlogram [21] to represent each image and measure the similarity relationships between pairs of images. Here,
by “content-based”, we mean that the search will only analyze the actual content of each image rather than other metadata such as tags, keywords, or descriptions associated with the image. Hence, we get a graph with each node representing an image and each edge having a similarity value associated with it. After the construction of these “experience graphs”, we do a graph-theoretic analysis. The results show that the network properties of the experience graphs are independent of the image representation that is chosen. Image data from all subjects show “small world” properties like the semantic networks studied by [43]. More specifically, the contributions of this part are the following:

1. We show that the experience graphs have a small-world structure which is characterized by high proximity ratios and sparse connectivity, independent of the representation used.

2. The degree distribution analyses show that they are not scale-free, broad-scale or even single-scale that [3] studied.

3. We also find that the graphs constructed based on the color correlogram representation, which is better than the color histogram in many content-based image retrieval systems [25], have shorter average path lengths and higher global clustering coefficients than the graphs constructed based on the color histogram representation.
CHAPTER 2

METRIC BASED AUTOMATIC EVENT SEGMENTATION

2.1 Related Work

In one of the early studies, by using a mobile phone equipped with a sensor box, [18] 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 [32] 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 [32]. 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\textsuperscript{1}, [14] 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, [13] 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” [14, 34] feature for accelerometer data representation.

More recently, [36] propose a framework of the lifelog system by using a smart phone.

\textsuperscript{1}SenseCam[19] 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 infrared sensor to detect the presence of a person.
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” [9, 40] 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 [36].

2.2 Smart Phone and Data Collection

Several research groups [20, 23, 46, 40, 36] have developed personal Lifelog systems to capture personal experiences by wearing various sensors and a wearable 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 2.1. 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.

![Figure 2.1: Smart phone and software for collecting activity data.](image)

2.3 Metric-based Event Segmentation

Generally, this model involves three steps, namely, “Feature Extraction”, “Dissimilarity Measure” and “Event Boundary Detection”. The overall procedure is depicted in Figure 2.2.

2.3.1 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 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,

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

(2.3.1)

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

And

$$Y_k = |y_k|$$

(2.3.2)

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

Since $Y_{2^{-k}} = Y_{2^{k}+k}$, 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 [4] 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 [34] and suggested by [14, 13] for accelerometer data captured by SenseCam on event segmentation.

3. “Behaviour Text” feature: This feature is suggested by [9, 40, 36] 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 [13] suggest which are: color layout, color structure, scalable color and edge histogram.

**2.3.2 Dissimilarity Measure**

The dissimilarity measure 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, the 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 distance (KL):

\[ d_{KL} = \frac{1}{2} (\mu_1 - \mu_2)^T (\Sigma_1^{-1} + \Sigma_2^{-1}) (\mu_1 - \mu_2) + \frac{1}{2} tr(\Sigma_1^{-1} \Sigma_2 + \Sigma_2^{-1} \Sigma_1 - 2I) \]  
\hspace{1cm} (2.3.3)

Bhattacharyya distance (BHA):

\[ d_{BHA} = \frac{1}{4} (\mu_1 - \mu_2)^T (\Sigma_1 + \Sigma_2)^{-1} (\mu_1 - \mu_2) + \frac{1}{2} \log \frac{|\Sigma_1 + \Sigma_2|}{2 \sqrt{|\Sigma_1| |\Sigma_2|}} \]  
\hspace{1cm} (2.3.4)

Bayesian Information Criterion (BIC)[10]:

\[ d_{BIC} = N \log |\Sigma| - N_1 \log |\Sigma_1| - N_2 \log |\Sigma_2| - \lambda P \]  
\hspace{1cm} (2.3.5)

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

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

\[ d_{AED} = \frac{1}{|A||B|} \sum_{u \in A, v \in B} \text{dist}(u, v) \]  
\hspace{1cm} (2.3.6)

where \( \text{dist}(u, v) \) 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):

\[ d_{MVD} = \text{dist}(\mu_1, \mu_2) \]  
\hspace{1cm} (2.3.7)

where \( \text{dist}(\mu_1, \mu_2) \) is a Euclidean distance between the mean vector \( \mu_1 \) and \( \mu_2 \) for sliding windows.
2. For the Motion Change Rate (MCR) feature: In order to compare the system performance suggested by [14, 13], 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 [27] is applied followed by the event boundary detection [13]. 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 [40, 36] is then applied on pairwise sliding windows. In the experiment, we test different window sizes and show their corresponding performances.

**Image data**

Suggested by [13], 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.

2.3.3 Event Boundary Detection

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

**Smoothing**

Since the event boundaries are “fuzzy”[52] 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 [24], 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, $I(t)$:

$$L(t, \sigma) = G(t, \sigma) \ast I(t),$$  \hspace{1cm} (2.3.8)

where $\ast$ is the convolution operation in $t$, and

$$G(t, \sigma) = \frac{1}{2\pi\sigma^2}e^{-\frac{t^2}{2\sigma^2}}$$  \hspace{1cm} (2.3.9)

The smoothing level for Gaussian kernel is denoted by parameter $\sigma$. In the experiment, the $\sigma$ 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 $\sigma$ in “Gaussian Convolution”.

Secondly, a peak is “tall” when its height is greater than some threshold as Figure 2.3 depicts. In this model, all the boundaries corresponding to tall peaks are selected as the final event boundaries. In this model, besides the $\sigma$ in “Gaussian Convolution”, there is another free parameter - the threshold. In the experiment, the $\sigma$ we select for cross validation is from $\{1, 2, 3, ..., 90\}$ and the threshold is $i \times \frac{(\text{max} - \text{min})}{50} + \text{min}$ 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”[56] if

\[ |d(\text{max}) - d(\text{min}_{left})| > \alpha \sigma' \]

or

\[ |d(\text{max}) - d(\text{min}_{right})| > \alpha \sigma' \tag{2.3.10} \]

where \( \alpha \) is a parameter, \( \sigma' \) is the standard deviation of the distance curve. And \( \text{min}_{left} \) and \( \text{min}_{right} \) are the left and right minimas around the peak “max” as Figure 2.4 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, \( \sigma \) in “Gaussian Convolution” and \( \alpha \) in detecting the “significant peak”. In the experiment, the \( \alpha \) we selected for cross validation is from \( \{0.1, 0.2, 0.3, \ldots, 2.0\} \) and the \( \sigma \) for cross validation is from \( \{1, 2, 3, \ldots, 90\} \).
2.4 Segmentation Quality Measure

2.4.1 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.

2.4.2 Quality Measure

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 [1].

Figure 2.4: Significant Peak
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:

$$PRC = \frac{N_{hit}}{N_f}$$  \hspace{2cm} (2.4.1)

$$RCL = \frac{N_{hit}}{N_{ref}}$$  \hspace{2cm} (2.4.2)

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

$$F = \frac{2 \times PRC \times RCL}{PRC + RCL}$$  \hspace{2cm} (2.4.3)

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

### 2.4.3 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 2.5 shows.

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 [39] (see Figure 2.6).

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.
Figure 2.5: 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 ($N_f=6$).

Figure 2.6: The first plot shows an example of an overlapping search region causing an 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 $l_1$. 

20
2.5 Experiments

2.5.1 Dataset

Sensor Data

Unlike [14] 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 [6]. Since the raw accelerometer data is recorded using the device’s coordinate system, we convert it into world’s coordinate system (see Figure 2.7) and eliminate the impact of the gravity by using a low pass filter to isolate the force of gravity. We call this converted data adjusted accelerometer data. In the experiments, we investigate the performance difference between these two different representations by using different distance measures and different models.

Figure 2.7: The left figure indicates the device’s coordinate system and the right figure represents the world’s coordinate.
Ground Truth Boundaries

The definition of an event is largely subjective and also has many different levels of granularity, for example, [54] shows that event boundaries are hierarchically structured, such that fine-grained events are clustered into larger coarse-grained events. And they also define two different types of boundaries [31], “fine” boundaries and “coarse” boundaries where “fine” boundaries are related to smallest events and “coarse” boundaries are related to largest events. In this experiment, we ask the wearers to manually create a ground truth of segmentations for all his/her own recorded images. They are asked to make event boundaries based on the changes of their own intentions or goals. In this study, they are not instructed to specify “fine” boundaries and “coarse” boundaries.

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 [50]. 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 [13].

2.5.2 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.
2.5.3 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 2.8 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 $0.22 \pm 0.026$, 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.

![Graph showing F values for different number of random boundaries](image)

Figure 2.8: F values for different number of boundaries
**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 2.9 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%.

![F values for different proportion of ground truth boundary replacement](image)

Figure 2.9: F values for different proportion of ground truth boundary replacement

### 2.5.4 Experimental Results and Analysis

1. Our Proposed Models: Fig 2.10 and Fig 2.11 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) and 90 frames (equaling 135 seconds).  

Figure 2.10: Event segmentation on accelerometer data using our proposed model

The “Histogram Intersection” distance suggested by [11] 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 2.10) and image data (Fig 2.11), it is clear, the event segmentation performance from using accelerometer data is much higher than using image data.

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.
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 method “CombMIN” suggested by [13] 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 [11].

Fig 2.12 shows the results for accelerometer data using the “Mean Threshold” method suggested by [13] for peak detection. The best performance using the accelerometer data is around 0.53.

The early fusion method is suggested by [11] (left panel in Fig 2.13) so that different image representations are concatenated into a signal representation, then a distance
Figure 2.12: Event segmentation using Dublin City University’s System - Accelerometer Data Only

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 2.13) 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 2.13 and Fig 2.12, 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 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 [9, 40] according to the experimental results. Fig 2.14 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 [9, 40] 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.
Figure 2.14: Event segmentation using “Behaviour Text” feature
CHAPTER 3
THE NETWORK PROPERTIES OF EXPERIENCE GRAPHS

3.1 Data Collection

We collect data from 4 subjects who use Microsoft Research SenseCams to capture images every 10 seconds as they engage in their everyday activities. They are free to turn the camera off when they want to protect their privacy. However, they are instructed to provide at least 6-7 hours worth of data each day. In addition to the images clicked at regular intervals, images are also captured every time the sensors in the camera detect changes in color, light-intensity and temperature. We assume that the data thus collected sufficiently approximates the visual environmental input that an individual receives over the course of one week.

3.2 Basic Concepts of Graph Theory and Small-World Topology

We will define some terminology of graph theory and introduce small – world topology in this section.

3.2.1 Terminology of Graph Theory

The experience network is represented as a graph, consisting of a set of nodes or vertices V and a set of edges E that join pairs of nodes. The size (number of nodes) of the network is denoted by n. An undirected arc is an edge for which the ends are equivalent – there
is no head or tail. An *undirected graph* is a graph in which the nodes are connected by undirected arcs.

*Neighbors* are pairs of nodes which are connected by an arc or edge. In an undirected graph, a *path* is a sequence of vertices such that from each of its vertices there is an edge to the next vertex in the sequence. For a particular path from node x to node y, the *path length* is the number of edges (for an undirected graph) along that path. In an *undirected connected* graph, there exists a path between any pair of nodes. A *connected component* is a subset of nodes that is connected.

### 3.2.2 Small-World Topology

A small world model, which was introduced by [48], has a “small world” topology. More precisely, a small world network possesses the following properties: i) *“Short average path length”*, meaning most pairs of vertices are connected by a short path through the network. ii) *“High ’clustering’ or ’transitivity’”*, meaning that if two vertices have another neighboring vertex in common, there is a high probability that these two vertices will be connected directly to each other.

Given a network, it is easy to measure the *average path length*: Find the distances between all pairs of vertices in the network (not counting vertex pairs that are not connected at all) and compute their average. However, measuring clustering is a little more complicated. [48] proposed a measure of clustering, the *clustering coefficient* (or *local clustering coefficient*). The clustering coefficient of a network is the average clustering over all n vertices:

\[
C_{ws} = \frac{1}{n} \sum_{i=1}^{n} C_i = \frac{1}{n} \sum_{i=1}^{n} \frac{\text{number of connected neighbor pairs}}{\frac{1}{2}k_i(k_i - 1)}
\]

(3.2.1)

where \(k_i\) is the degree of vertex \(i\) and \(C_i\) is the clustering coefficient for vertex \(i\).

However, the definition here has some problems. Because of the factor \(k(k-1)\) in the
denominator which contributes significantly to the value of $C$, the main problem for Eq. (3.2.1) is that it is heavily biased in favor of vertices with low degree.

Thus, [28] proposed a *global clustering coefficient* defined as follows:

$$C = \frac{3 \times \text{(number of triangles on a graph)}}{\text{(number of connected triples of vertices)}}$$

(3.2.2)

where a triangle refers to a group of three vertices that are connected to both of the others, and a connected triple refers to a group in which a vertex is connected to an (unordered) pair of other vertices. Each triangle contributes to three separate connected triples. The factor of 3 in the numerator guarantees the value of $C$ lies strictly in the range from zero to one. A fully connected graph has $C = 1$. In the analysis below, we use Eq.(3.2.2) instead of Eq.(3.2.1) to measure the clustering.

### 3.2.3 Random Graphs

The Erdős Rényi model $G(n, m)$ [16] is a standard model to generate random graphs. In the $G(n, m)$ model, a graph is chosen uniformly randomly from the collection of all graphs that have exactly $n$ nodes and $m$ edges. For example, in the $G(4, 2)$ model, each of the six possible graphs on four vertices and two edges are chosen with probability $1/6$.

### 3.2.4 Testing for a Small-World Topology

[48] defined a network to be a small world network if mean vertex-vertex distance $L$ is comparable with that on a random graph with the same number of nodes and edges ($L \approx L_{\text{rand}}$) and the clustering coefficient is much greater than that for a random graph with the same number of nodes and edges ($C \gg C_{\text{rand}}$). In order to give a quantitative measure of “small worldliness”, [47] defined a *proximity ratio* as follows:

$$\mu = \frac{C/C_{\text{rand}}}{L/L_{\text{rand}}}$$

(3.2.3)
which is of order 1 on a random graph model $G(n,m)$ [16] with the same number of nodes $n$ and edges $m$, but $\mu \gg 1$ on a small world graph. The following table 3.2.4 shows us some examples from [47] based on their analysis of the graphs studied by [48].

<table>
<thead>
<tr>
<th></th>
<th>$L$</th>
<th>$L_{rand}$</th>
<th>$C$</th>
<th>$C_{rand}$</th>
<th>$\mu$</th>
</tr>
</thead>
<tbody>
<tr>
<td>film actors</td>
<td>3.65</td>
<td>2.99</td>
<td>0.79</td>
<td>0.00027</td>
<td>2396</td>
</tr>
<tr>
<td>power grid</td>
<td>18.7</td>
<td>12.4</td>
<td>0.08</td>
<td>0.005</td>
<td>10.61</td>
</tr>
<tr>
<td>C.elegans</td>
<td>2.65</td>
<td>2.25</td>
<td>0.28</td>
<td>0.05</td>
<td>4.755</td>
</tr>
</tbody>
</table>

Table 3.1: Characteristic path lengths, clustering coefficients, and proximity ratios for graphs studied in [48] with a small world topology. In this table, the “film actors” corresponds to a collaboration graph of film actors, “power grid” corresponds to the power grid of the western United States and “C.elegans” is the neural network of the worm *Caenorhabditis elegans* [48].

### 3.3 Experience Graph Construction

Each image can be regarded as a snapshot of visual experience. With these snapshots of experience as nodes, we construct an undirected graph based on similarity calculated between pairs of images. In order to compute the similarities, we introduce some techniques from content-based image retrieval (CBIR). In this step, two different image representation methods (color histogram and color correlogram) are used and the corresponding between-item similarity measure is applied to capture image (dis)similarity for all pairs of images. These experience graphs are fully-connected initially. Then we apply between-item similarity thresholds to obtain partially-connected graphs for further analysis. The details are as follows:
3.3.1 Color Histogram

For image retrieval problems, the color histogram method is a simple way of representing global image features. This representation is invariant under image rotation and translation, and has been used extensively [22].

Similarity Measure

In this part of the analysis, we use the histogram intersection distance to measure similarity among images. The intersection of histograms $h$ and $g$ is given by:

$$
    d(h, g) = \frac{\sum_A \sum_B \sum_C \min(h(a,b,c), g(a,b,c))}{\min(|h|, |g|)}
$$

(3.3.1)

where $|h|$ and $|g|$ give the magnitudes of the histogram height corresponding to a certain color, which is equal to the number of samples (i.e., number of pixels) containing that particular color. The sum is normalized by the histogram with the fewest samples [22]. In our experiment, all the images have the same size which is 640 by 480, hence, $|h| = |g| = 640 \times 480$.

3.3.2 Color Correlogram

The color histogram suffers from limited discriminative power due to the fact that images with completely different spatial organization of colors can produce identical color histograms as long as the color content is similar. To overcome this limitation, [21] introduced the color correlogram which describes how the spatial correlation of colors changes with spatial distance. [25] carried out a detailed comparison of a number of commonly used color features for designing a content-based image retrieval system based on a large collection of image data. The color features they compared include color histogram, color moments [44], color coherence vector [37] and color correlogram controlling for different color spaces and quantizations. Their experimental results indicate that the color correlogram provides the best retrieval performance.
Notation

Let $I$ be an $n \times m$ image. The colors in $I$ are quantized into $k$ colors $c_1, c_2, \ldots, c_k$. For a pixel $p = (x, y) \in I$, let $I(p)$ denote its color. Let $I_c \triangleq \{ p | I(p) = c \}$ where $c \in \{c_1, c_2, \ldots, c_k\}$. The notation $p \in I_c$ is equivalent to $p \in I, I(p) = c$. For pixels $p_1 = (x_1, y_1), p_2 = (x_2, y_2)$, we define $L_\infty$ norm to measure the distance between them, such that $|p_1 - p_2| \triangleq \max\{|x_1 - x_2|, |y_1 - y_2|\}$. Hence, we denote $I_c^d \triangleq \{ p_2 | p_1 \in I_c \land |p_2 - p_1| = d \}$, where $d \in \{1, 2, 3, \ldots, l\}$ is a distance.

**Color Correlogram**

In practical terms, a correlogram of an image corresponds to a 3 dimensional table whose entry $(c_i, c_j, d)$ is the probability of finding a pixel of color $c_j$ at a distance $d$ from a pixel of color $c_i$ in the image. More formally, the correlogram of $I$ is defined for $i, j \in \{1, 2, 3, \ldots, k\}, d \in \{1, 2, 3, \ldots, l\}$ where distance $d$ is a priori, such that

$$\gamma_{c_i, c_j}^{(d)}(I) \triangleq \Pr_{p_1 \in I_c, p_2 \in I} [p_2 \in I_{c_j} \land |p_2 - p_1| = d] \triangleq \frac{|I_{c_j} \cap I_c^d|}{|I_c^d|} \quad (3.3.2)$$

Given any pixel of color $c_i$ in the image, $\gamma_{c_i, c_j}^{(d)}(I)$ gives the probability that a pixel at distance $d$ away from the given pixel is of color $c_j$. Hence, the color correlogram is a three dimensional table indexed by color and distance between pixels and the size of the correlogram is $O(k^2l)$.

**Banded Color Correlogram**

The banded correlogram is for reducing storage. Given $b$ which $b$ divides $l$, for $1 \leq d \leq l/b$,

$$\overline{\gamma}^{(d)}_{c_i, c_j}(I) \triangleq \sum_{d'=(d-1)b+1}^{db} \gamma_{c_i, c_j}^{(d')}(I) \quad (3.3.3)$$

For each color pair $(c_i, c_j)$, the probability values for the distances in the selected distance set whose cardinality is $b$ are summed as a single number. Hence, a banded color
correlogram is a restricted version of the color correlogram. The following equation is an example where \( d \in \{1, 3, 5, 7\} \).

\[
\gamma_{c_i, c_j}(I) \triangleq \sum_{d \in \{1, 3, 5, 7\}} \gamma_{c_i, c_j}^{(d)}(I)
\]  

(3.3.4)

**Similarity Measure**

We use the \( L_1 \) distance measure for correlograms of images \( I \) and \( I' \) as follow:

\[
|I - I'|_{\gamma L_1} \triangleq \sum_{i, j \in \{1, 2, 3, \ldots, k\}, d \in \{1, 2, 3, \ldots, l\}} \frac{\left| \gamma_{c_i, c_j}^{(d)}(I) - \gamma_{c_i, c_j}^{(d)}(I') \right|}{1 + \gamma_{c_i, c_j}^{(d)}(I) + \gamma_{c_i, c_j}^{(d)}(I')}
\]  

(3.3.5)

**3.3.3 Representation in HSV Space**

The HSV (hue, saturation, value) color space is thought to provide better correspondence with human visual perception of color (dis)similarities than the RGB color space [25]. The efficiency of properly quantized HSV color correlograms was also demonstrated against RGB correlograms [35]. For these reasons, we work in the HSV space when constructing the color histogram and correlogram representations for our images.

In the HSV (hue, saturation, value) color space, the hue circle consists of the primaries red, green and blue separated by 120 degrees. This requires the hue to be quantized more finely than the others. For constructing the color histogram, \( H \) is quantized to 30 levels and \( S \) & \( V \) are quantized to 10 and 3 levels respectively. Thus, the quantized HSV space has \( 30 \times 10 \times 3 = 900 \) histogram bins.

For the color correlogram, the HSV color space is quantized into \( 18 \times 3 \times 3 \). As [21] did, we let \( d \in \{1, 3, 5, 7\} \) Eq.(3.3.4). We use the banded color correlogram as in Eq.(3.3.4) instead of the full color correlogram for storage reasons.

**3.3.4 Edge Selection**

After the similarity measure between all pairs of images, we get a fully connected graph \( G \). However, we are only interested in connecting up nodes that have some degree of
similarity between them. We are interested in the similarity structure of these experience nodes and are not too concerned with highly dissimilar nodes. Therefore, we select the edges whose similarity value ranks in the top x% (16 proportions from 0.001% to 1.0%) of all the distances calculated over the entire data set to construct the experience graphs. This translates to selecting edges that are above a certain similarity threshold such that two nodes associated with the same edge are highly similar. Hence, we end up with undirected graphs that is constructed by connecting experience nodes that are not highly dissimilar and these graphs exhibit sparse connectivity patterns.

3.4 Graph-theoretic Analyses of Experience Graphs

We use several statistical features defined above to characterize the structure of experience graphs.

3.4.1 Proximity Ratios

The first statistical feature is the proximity ratio \( \mu \) [47]. In table 3.2, we show that the proximity ratio \( \mu \), is large (\( \mu \gg 1 \)) in all the experience graphs. This demonstrates that all the experience graphs have a small world topology. Secondly, the proximity ratios decrease as the number of edges increases (from 0.03% to 1.0%).

If we compare graphs constructed with the same proportion of edges using the color histogram and color correlogram representations for the same subject, in general, the proximity ratios for the color correlogram graphs are higher. Figure 3.1 also demonstrates that the color correlogram graphs have a higher global clustering coefficient but lower average shortest path lengths and diameters than the corresponding color histogram graphs. Hence, the graphs that use color correlogram seem “denser” or have more “small world” properties than the color histogram graphs.
Table 3.2: Proximity Ratios

<table>
<thead>
<tr>
<th>Top X%</th>
<th>Subject1</th>
<th>Subject2</th>
<th>Subject3</th>
<th>Subject4</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Color</td>
<td>Histogram</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.03%</td>
<td>1095.38</td>
<td>2054.03</td>
<td>958.27</td>
<td>320.41</td>
</tr>
<tr>
<td>0.04%</td>
<td>637.86</td>
<td>572.59</td>
<td>594.18</td>
<td>322.78</td>
</tr>
<tr>
<td>0.05%</td>
<td>681.17</td>
<td>281.99</td>
<td>219.75</td>
<td>279.36</td>
</tr>
<tr>
<td>0.08%</td>
<td>151.93</td>
<td>389.41</td>
<td>137.53</td>
<td>125.93</td>
</tr>
<tr>
<td>0.1%</td>
<td>123.50</td>
<td>348.77</td>
<td>128.43</td>
<td>88.61</td>
</tr>
<tr>
<td>0.2%</td>
<td>63.05</td>
<td>107.88</td>
<td>41.50</td>
<td>56.99</td>
</tr>
<tr>
<td>0.5%</td>
<td>25.41</td>
<td>17.26</td>
<td>16.24</td>
<td>19.01</td>
</tr>
<tr>
<td>1.0%</td>
<td>10.44</td>
<td>13.37</td>
<td>9.58</td>
<td>12.42</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Color</th>
<th>Correlogram</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.03%</td>
<td>2114.89</td>
<td>542.04</td>
</tr>
<tr>
<td>0.04%</td>
<td>1154.51</td>
<td>592.26</td>
</tr>
<tr>
<td>0.05%</td>
<td>1312.96</td>
<td>312.09</td>
</tr>
<tr>
<td>0.08%</td>
<td>233.58</td>
<td>106.03</td>
</tr>
<tr>
<td>0.1%</td>
<td>199.06</td>
<td>81.67</td>
</tr>
<tr>
<td>0.2%</td>
<td>91.25</td>
<td>44.98</td>
</tr>
<tr>
<td>0.5%</td>
<td>19.48</td>
<td>20.51</td>
</tr>
<tr>
<td>1.0%</td>
<td>12.44</td>
<td>11.37</td>
</tr>
</tbody>
</table>

3.4.2 Degree Distribution Analysis

Small world networks are classified [3] into (a) scale free networks, characterized by a vertex connectivity distribution that decays as a power law (e.g. the World-Wide Web).
Figure 3.1: The means of Global Clustering Coefficients, Average shortest path lengths and diameters for all subjects

(b) *broad-scale* networks, characterized by a connectivity distribution that has a power law regime followed by a sharp cutoff (e.g. the network of movie-actor collaborations) or (c) *single-scale* networks, characterized by a connectivity distribution with a fast decaying tail (e.g. the electric power grid for Southern California, the neuronal network of the worm *Caenorhabditis elegans*). In order to specify the class that our experience graphs fall into, we analyze the degree (connectivity) distributions.

**Notation**

The degree of a vertex in a network is the number of edges incident on or connected to that vertex. If there are $n$ nodes in a network and $n_k$ of them have degree $k$, then the degree distribution $p(k)$ of a network is defined to be $p(k) = n_k/n$. A cumulative degree
distribution $P(k)$ is the fraction of nodes with degree greater than or equal to $k$. More formally,

$$P(k) = p(k) + p(k+1) + \ldots$$

(3.4.1)

Power-law degree distributions have been studied in detail and have gained a great deal of attention [2, 15]. The power law degree distribution has the following form:

$$p(k) \sim ck^{-\gamma}$$

(3.4.2)

where $c$ is a constant.

**Scale-free networks**

*Scale-free networks* emerge in the context of a growing network where new vertices connect preferentially to the vertices are already well connected [5].

The graphs with a power-law degree distribution are also referred to as *scale-free networks* [5]. The distribution of the local connectivities is scale free. Generally, a scale-free network is a network whose distribution of connectivities decays with a power law tail.

**Analysis Results**

Similar to what [3] did for the analysis of the classes of small world networks, we compute the cumulative degree distributions for all the subjects and present log-log and linear-log plots of these distributions. For each subject, the top 1.0% edges are used to create the color histogram and correlogram graphs.

Figure 3.2 displays the log-log plots of cumulative degree distributions for the experience graphs of 4 subjects. If the distribution had a power law tail, it would fall on a straight line in this log-log plot. Clearly, these distributions do not follow a power law decay. However, these plots suggest that the tails of the distributions decay faster than a power law would. Hence, our experience graph connectivities are not scale-free.
Figure 3.2: Log-log plots of the cumulative degree distributions for experience graphs.

Figure 3.3 displays the linear-log plots of the cumulative degree distributions for the experience graphs. The distributions seem to fall on multiple straight lines partially, suggesting multiple exponential decays of the distribution of connectivities. However, instead of suggesting multiple exponential decays, we prefer to conclude that they do not fall into one of the classes that [3] studied and hence need further investigation. Interestingly, we find that the correlogram graphs tend to have higher degree vertices than the histogram graphs in general.
Figure 3.3: Linear-log plots of the cumulative degree distributions for experience graphs.
CHAPTER 4
CONCLUSIONS AND FUTURE WORK

4.1 Automatic Event Segmentation

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 [13] 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 [6] and may fail to detect some event boundaries.

Third, our proposed peak selection methods using the bag of feature representation
such as “Behavior Text” [40] 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 the gravity impact for the accelerometer
data is useless or even harmful for event segmentation. And the orientation of the phone
(implied by the accelerometer data based on a device’s local coordinate system) may not
be very important for event segmentation.

Finally, we believe that the movement features can play an important role on coarse-
grained event boundary generation.

4.2 The Network Properties Of Experience Graphs

We show that experience networks, like many other natural networks [48], have a small-
world topology which is characterized by a high clustering coefficient and a short average
shortest path length. Furthermore, we find that the graphs constructed based on the color
correlogram representation (and the corresponding distance measure) have a higher prox-
imity ratio [47] than the graphs constructed based on the color histogram representation
and similarity measure. In other words, they have shorter average path lengths and higher
global clustering coefficients which means that their structures are more “small world” than
the graph constructed based on the histogram representation.

The study of degree distributions shows that they are not scale-free graphs and that they
do not fall into any of the other classes that [3] have studied. This suggests that experience
graphs may be a new class of small-world graphs and needs further exploration.

It has also been demonstrated [42] that the color correlogram image representation
is the preferred representation in a study of episodic memory and context segmentation
(where compression in time is important in addition to spatial considerations). This is a
desirable property if such a model is to be applied in a study of episodic memory and contextual reinstatement as a search over networks. In such a framework, episodic memory can be regarded as information stored in multiple networks of visual experience, time information, semantic information about various events, etc and contextual reinstatement can potentially be modeled as a search over those networks.


