Semantic Interpretation of Events in Lifelogging

Peng Wang

B.A., M.A.

A Dissertation submitted in fulfilment of the requirements for the award of

Doctor of Philosophy (Ph.D.)

to the

Dublin City University

School of Computing

CLARITY: Centre for Sensor Web Technologies

Supervisor: Prof. Alan F. Smeaton

October 2011

Declaration

I hereby certify that this material, which I now submit for assessment on the programme of study leading to the award of Doctor of Philosophy is entirely my own work and has not been taken from the work of others save and to the extent that such work has been cited and acknowledged within the text of my work.

Signed: Peng Wang

Student ID: 58103261

Date:

Contents

Abstract **viii**

Acknowledgements **ix**

List of Figures **xi**

List of Tables **xiv**

1 Introduction **1**

1.1 Introducing Lifelogging 2

1.1.1 Lifelogging Based on Context-Sensing 2

1.1.2 Typical Applications of Lifelogging 5

1.2 Research Questions 9

1.3 Thesis Structure 14

2 Background to Semantic Interpretation of Lifelogging **17**

2.1 Introduction 17

2.2 Multimodal Context-Awareness 21

2.3 Multimedia Semantic Retrieval 23

2.3.1 Feature Extraction and Representation 24

2.3.1.1 Low-Level Features 24

2.3.1.2 High-Level Features 27

2.3.2 Content-based and Concept-based Retrieval 28

2.3.3 Query Expansion / Concept Selection 29

2.3.4 TRECVid vs. Lifelogging 30

2.3.5 Difficulties in Lifelogging Retrieval 32

2.4 Event-Centric Media Processing 34

2.5 Summary 36

3 Semantic Density-based Concept Selection 38

3.1 State-of-the-Art Concept-based Retrieval 39

3.2 Event Semantic Space (ESS) 41

3.2.1 Everyday Activities: Exploring and Selection 42

3.2.2 Topic Related Concepts 44

3.2.3 Constructing the Event Semantic Space (ESS) 46

3.3 Investigating ESS Concept Relationships 48

3.3.1 Definitions 48

3.3.2 Lexical Similarity Based on Taxonomy 48

3.3.3 Contextual Ontological Similarity and Relatedness 51

3.4 Concept Selection Based on Semantic Density 53

3.4.1 Text Pre-Processing 53

3.4.2 Conjunctive Concept Similarity 55

3.4.3 Density-based Concept Selection 56

3.5 Leveraging Similarity for Concept Ranking 59

3.5.1 Concept Similarity Model 60

3.5.2 Similarity Rank 60

3.6 Summary 62

4 Evaluating Concept-based Similarity and Selection 64

4.1 User Experiment 64

4.2 Experimental Evaluation Methodology 67

4.3 Evaluation Setup 70

4.4 Result Evaluation 74

4.4.1 Evaluating the Clustering Algorithm 74

4.4.2 Similarity Ranking Assessment 78

4.5 Evaluation on TRECVid 81

4.6 Summary 87

5 Fusion of Event Semantics for Lifelogging 89

5.1 Event-based Visual Processing 93

5.2 Multi-Concept Event Semantic Aggregation 95

5.2.1 Ontology-based Multi-Concept Classification 95

5.2.2 Interestingness-based Concept Aggregation 104

5.2.2.1 Event Concept Interestingness 105

5.2.2.2 Semantic Aggregation of Concepts 107

5.2.3 Vector Similarity in Semantic Space 108

5.3 High-Level Semantic Activity Detection 110

5.3.1 Problem Description 111

5.3.2 Vocabulary Construction for SenseCam Images 112

5.3.3 Concept-Spatial HMM Activity Classification 115

5.3.3.1 HMM Model Structure 115

5.3.3.2 Parameter Training 116

5.4 Semantic Representation: a VSM-Like Paradigm 117

5.5 Experiment and Evaluation 120

5.5.1 Activity Classification Evaluation 120

5.5.1.1 Evaluation Data set 120

5.5.1.2 Evaluation on Clean Concept Annotation 121

5.5.1.3 Erroneous Concept Annotation 127

5.5.1.4 Evaluation on Erroneous Concept Annotation 128
5.5.2 Ontology-based Multi-Concept Detection Evaluation 131
5.5.3 Event Representation Evaluation 137
5.6 Summary 143
6 Event Modeling and Semantic Enhancement 144
6.1 Semantic Representation and Model Language 145
6.1.1 Ontologies 146
6.1.2 Resource Description Framework (RDF/RDFS) 148
6.1.3 OWL 150
6.2 Contextual Event Enhancement Architecture 151
6.2.1 An Illustrative Scenario 151
6.2.2 Event Ontology Based on Multi-Context 152
6.2.3 EventCube: an Enhanced Album of Events 155
6.3 Event Semantic Enhancement and Query 159
6.3.1 Linked Open Data (LOD) and SPARQL 159
6.3.1.1 LOD-based Event Enhancement 160
6.3.1.2 Enable the Semantic Query 161
6.3.2 Location Enhancement 162
6.3.3 Social Context Enhancement 166
6.4 Event Semantic Enhancement Experiment 168
6.4.1 Experiment Setup 168
6.4.1.1 Event Recording Setup 168
6.4.1.2 Retrieval Environment Setup 169
6.4.2 Aligning Semantics for Social Enhancement 171
6.4.3 Event-Centric Enhancement Application Overview 173
6.4.4 Assessing Context Enhancement 175
6.5 Summary 182

7 Conclusions and Future Work 183

7.1 Main Contributions 185

7.2 Future Work 187

Bibliography 188

A TRECVid Topic Set 215

B EventCube Ontology (in Turtle) 220

C Event Enhancement Record 224

D Location Clustering Examples 230

D.1 Clustering for a Whole Month of One Lifelogger 230

D.2 Clustering on Another User for a Day 231

Abstract

The topic of this thesis is lifelogging, the automatic, passive recording of a person's daily activities and in particular, on performing a semantic analysis and enrichment of lifelogged data. Our work centers on visual lifelogged data, such as taken from wearable cameras. Such wearable cameras generate an archive of a person's day taken from a first-person viewpoint but one of the problems with this is the sheer volume of information that can be generated. In order to make this potentially very large volume of information more manageable, our analysis of this data is based on segmenting each day's lifelog data into discrete and non-overlapping events corresponding to activities in the wearer's day. To manage lifelog data at an event level, we define a set of concepts using an ontology which is appropriate to the wearer, applying automatic detection of concepts to these events and then semantically enriching each of the detected lifelog events making them an index into the events. Once this enrichment is complete we can use the lifelog to support semantic search for everyday media management, as a memory aid, or as part of medical analysis on the activities of daily living (ADL), and so on. In the thesis, we address the problem of how to select the concepts to be used for indexing events and we propose a semantic, density-based algorithm to cope with concept selection issues for lifelogging. We then apply activity detection to classify everyday activities by employing the selected concepts as high-level semantic features. Finally, the activity is modeled by multi-context representations and enriched by Semantic Web technologies. The thesis includes an experimental evaluation using real data from users and shows the performance of our algorithms in capturing the semantics of everyday concepts and their efficacy in activity recognition and semantic enrichment.

Acknowledgements

I would like first to give sincere thanks to my supervisor Prof. Alan Smeaton who gave me so much valuable guidance not only on my research directions but also on my research methodologies. Every time I have discussions with him, I am impressed by his profound knowledge and his innovative guidance. I have deep gratitude to him for all he has done for me! I would also thank Deirdre Sheridan, Ann Marie Sweeney and Prof. Noel O'Connor who helped a lot in various aspects in my daily research.

Many thanks to my office mates Zhengwei Qiu, Eoin Hurrell, Graham Healy, Niamh Caprani, Adam Bermingham and Edel O'Connor who accompany me in my work every day. To work with them is really great! During my PhD, I also got much expertise from Cathal Gurrin, Gareth Jones, Prof. Songyang Lao, Aiden Doherty, Daragh Byrne, Yi Chen, James Lanagan, Neil O'Hare and Mark Hughes. Without their knowledge and suggestions, I could not solve the problems I met within my research. Special thanks to Zhenxin Zhang, Milan Redžić and Hyowon Lee who provided me with access to their SenseCam data. I also appreciate the help for my experiments from Jogile Kuklyte, Damien Connaghan, Dian Zhang, David Scott, Jinlin Guo and Paulina Piasek.

I can not fully express my gratitude to my Mum and Dad who support me all the time. Their endless care and unselfish love encouraged me to conquer whatever difficulties I come across in my life and research. I am such a lucky guy because I have my beloved girlfriend Ling Zou in company with me during the most critical period, the last year of my PhD. I owe many thanks to her for her encouragement, support and understanding.

The work of my PhD is supported by China Scholarship Council (File No. 2008611102) and by Science Foundation Ireland under grant number 07/CE/I1147 (CLARITY CSET). Many thanks to them! I also wish to thank the Digital Enterprise Research

Institute (DERI) in Galway, Ireland for hosting me during a research visit from March to May in 2010. Their expertise and practical skills in Semantic Web have been of great help for carrying out my research.

List of Figures

2.1 The Microsoft SenseCam (right as worn by a user). 22

2.2 Pipeline overview of semantic fusion. 33

2.3 Event model and layered structure. 35

3.1 Framework for concept-based retrieval 39

3.2 Concept space and event concept vector. 47

3.3 Event semantic density. 54

3.4 Inconsistency coefficient and PCA. 58

3.5 Number of clusters. 58

3.6 Relationship dendrogram for concepts. 59

3.7 Concept similarity link. 61

4.1 Concept number vs. agreement. 71

4.2 Concept distribution. 73

4.3 Average number of concepts selected per event topic. 73

4.4 Proportion of selected concepts in the concept set. 74

4.5 Group Consistency comparison. 75

4.6 Comparison of precision. 76

4.7 Comparison of Set Agreement. 77

4.8 Comparison of Rank Correlation. 78

4.9 Comparison of Pairwise Orderedness (small set). 80

4.10 Comparison of Pairwise Orderedness (larger concept set). 80

4.11 Comparison for Top-5 ranked concepts (smaller concept set). 81

4.12 Comparison for Top-10 ranked concepts (smaller concept set). 82

4.13 Comparison of Set Agreement (left: human benchmark, right: collection benchmark) in TRECVid. 83

4.14 Comparison of Group Consistency (MediaMill 101 concepts). 84

4.15 Comparison of Top-K concepts (human benchmark). 85

4.16 Comparison of Top-K concepts (collection benchmark). 86

5.1 Concept distribution among SenseCam images. 90

5.2 Concept distribution – logarithmic scale plot. 91

5.3 Ontology for multi-concept lexicon (indoor highlighted). 98

5.4 Ontology for multi-concept lexicon (outdoor highlighted). 98

5.5 Concept classification results (Conf vs. Multi-class Margin). 102

5.6 Correlation of accuracy and confidence/multi-class margin (left: indoor; right: outdoor). 103

5.7 Event-level concept aggregation. 108

5.8 Vocabulary construction example in $2D$ space. 115

5.9 HMM structure for activity modeling. 116

5.10 Two states transition model. 116

5.11 Semantic representation for events. 119

5.12 Search graph for MAP optimizing (3 states). 122

5.13 Search graph for MAP optimizing (2 states). 123

5.14 Comparison of two sampling methods (clean data). 126

5.15 Averaged concept MAP with different positive class means. 129

5.16 Averaged activity MAP with different positive class means. 130

5.17 Comparison of two sampling methods (simulated data). 131

5.18 Accuracy improvement for ‘Indoor’ concept by single concepts. 133

5.19 Precision-recall curve ('Outdoor'). 134

5.20 Effect of adjusted confidence ('Outdoor'). 135

5.21 AP improvement by Disjoint Confidence Adjusting (DCA). 136

5.22 MAP improvement by Disjoint Confidence Adjusting (DCA). 137

5.23 Contrast difference (SS-COS). 140

5.24 Saliency difference (SS-COS). 140

5.25 Contrast vs. Saliency Measure. 141

5.26 Correlation of quality with Top-k. 141

5.27 Concept number in single representation. 142

6.1 Lifelogging event ontology. 153

6.2 Event enhancement architecture. 156

6.3 Location clustering diagram (from left to right: shop, accommodation and lab). 164

6.4 Event enhancement interface (left: event viewer; right: map and enhancement browser). 174

6.5 Day samples of place detection. (left: routine working days; right: one day in city center) 178

List of Tables

3.1 Everyday activities from [83] in decreasing order of enjoyment 43

3.2 Target activities for our lifelogging work 44

3.3 List of concept similarity matrices 51

4.1 Demographic information of participants 65

4.2 Experimental data set for pooling 66

4.3 Examples of everyday concepts 67

4.4 Experimental concept sets 72

5.1 Experimental data set for activity classification 121

5.2 State number searching (by MAP) 124

5.3 Event detection results 125

5.4 SenseCam concept sets for keyframe evaluation 138

5.5 Experimental data set 139

6.1 Example results of SPARQL query (Listing 6.2) 166

6.2 The whereabouts log 170

6.3 Online data sources employed 171

6.4 Evaluating location clusters 176

6.5 Enhanced samples for places 179

6.6 Enhanced samples for social context 181

Chapter 1

Introduction

The idea of recording our everyday lives is not new to us. The writing diary is one way we used to record the experiences of individuals and the diary has been handed down from generation to generation for centuries. With the pervasive application of computing technology, the form we use to record our daily experiences is changing. Digital blogging is an example of a new form of diary which has become very popular recently. While the traditional diary is usually private and intended for our own use, blogging is the opposite in that it is usually open to the public and used for sharing one's experiences, feelings, opinions, comments and so on. Blogging is a type of on-line-based recording of experiences/memories intended for sharing and reliving where everyday activities can be shared. Most blogging today is text only although some are posted with multimedia information such as digital photos and video clips. With diaries and blogging, these can only record or document a small part of one's activities by manual selection and editing of content whereas the idea of automatic life recording tries to record every detail of our everyday lives. Can we efficiently record the aspects of our lives with the advanced sensing devices? Can we efficiently access the content of such recordings and find useful information from a large volume of life logs? The current research area of lifelogging is trying to answer these two questions.

1.1 Introducing Lifelogging

The earliest motivation behind automatic generation of personal digital archives can be traced back to 1945 when Bush expressed his vision [32] that our lives can be recorded with the help of the technology and the access can be made easier to these 'digital memories'. This new way of autobiography generation has become more and more realistic recently, with the advances of lightweight computing devices and highly accurate sensors. Mobile devices are approaching a more capable computing ability, dwarfing the most powerful computers in the past. The low price and the embedded nature of smaller and lightweight sensors (cameras, GPS, Bluetooth, accelerometers, etc.) make computing devices portable or even wearable to enable life recording to be done unobtrusively. The large volume of data storage and high speed wireless networks needed for this help the mobile platform to turn into people-centric sensors capturing multidimensional sensory inputs besides spatial and temporal data. Lifelogging is the term describing this notion of digitally recording aspects of our lives, where the recorded multimedia content is the reflection of activities which we subsequently use to obtain the meaning of daily events by browsing, searching, or querying.

1.1.1 Lifelogging Based on Context-Sensing

To build a mapping between the real world and the digital world, various contexts can be recorded for the capture of the true meaning of daily activities. Here, contexts refer to the information which can be used to characterize a situation. A large variety of contexts can be used in lifelogging such as textual information, photos, audio and video clips, environment information (light, temperature, pressure, etc.), bio-information (heart rate, galvanic response, etc.) and spacial information (location, acceleration, co-presence, etc.). These contexts are changing dynamically and if captured then they can be used as cues to our activities and thus help with accessing information in our personal digital libraries.

To develop this further, a large number of digital devices with sensors can be applied to capture the above-mentioned contexts. Among all the devices emerging, the digital camera is the most widely-used lifelogging device. Within the lifelogging community, cameras are often used as wearable devices to record still images [148] or videos [77, 110, 27]. Audio-based capture devices are also employed in some research like [165], [27] and [164]. In order to reduce the listening time spend trying to find a relevant segment of sound, in [164], the authors conducted an experiment combining speech recognition transcripts and time-compressed audio without sacrificing user comprehension.

Though location is not sufficient to fully reflect the semantics of events, it still attracts much interest in research like that reported in [13], [70], [71] and [60], to name a few. Among the tools used for location-awareness, Global Positioning System (GPS) is preferred as the first choice due to its accuracy and independence from infrastructure. Besides, GPS offers a wider range of location sensing than other fixed sensors which are infrastructure-dependent such as UbiSense [1]. To deal with the problem of GPS dropout when satellite signals are not visible such as when inside buildings, WiFi-based and GSM-based localizations have also been introduced. An alternate location scheme is used in [60], by combining different ways of localization to solve the application issue of energy consumption. [55] fused information from cell tower and discovered Bluetooth IDs to support localization for both outdoors and indoors. In addition to indoor localization, friendly names and MAC addresses of Bluetooth devices are often used as a context of people in proximity. [36] employed Bluetooth

[1] http://www.ubisense.net

devices to measure event similarity by analyzing Bluetooth presence, duration and familiarity. The experiment was carried out in MIT, using Bluetooth-enabled mobile telephones to identify the deep social patterns in user activities [55].

The accelerometer is another popular sensor which can easily be embedded in mobile devices to sense part of our physical contexts. [27] uses two triaxial accelerometers worn on the left side of the hip and the wrist of the dominant hand respectively, for activity classification. A similar application of accelerometers is carried out in [16]. In [133], activity classification is done using lower sampling rate accelerometer only, at the frequency of 1 Hz to facilitate longer battery usage.

Besides the sensors we mentioned above, some other sensing devices are also available for capturing user context such as heart rate, galvanic skin response and core body temperature. BodyMedia is an off-the-shelf device from BoydMedia Inc.[2] and ActiHeart from Cambridge Neurotechnology has the sensing capability of the combination of heart rate and motion [3].

In terms of the deployment of sensing devices, modern lifelogging can be categorized roughly into in-situ lifelogging and wearable lifelogging. In-situ lifelogging can also be simply described as lifelogging in instrumented environments. This means the activities can only be captured through installed sensors in the local infrastructure, therefore the recording is highly dependent on instrumented environments [116]. In wearable lifelogging, the sensing devices are portable and carried by the wearers. This is usually done by harnessing the wearers with head-mounted cameras [77, 110] or cameras mounted in front of chests [27, 148]. It's not hard for us to notice that, using digital cameras or camera-enabled mobile devices forms the main stream of this kind of lifelogging. This is because visual information contains more semantics of events which can be used to infer other contextual information like 'Who', 'What', 'Where' and 'When'. Visual lifelogging is the term used to describe

[2] http://www.bodymedia.com

[3] http://www.camntech.com/

both image-based and video-based lifelogging. Example visual lifelogging projects are Steve Mann's WearCam [108, 109, 110], the DietSense project at UCLA [136], the WayMarkr project at New York University [30], the inSense system at MIT [27] and the SenseCam [74, 148] developed at Microsoft Research Cambridge. Though these projects use various mobile devices for digital logging, they have the common feature of using cameras to capture still images or videos, to resemble the views of wearers. Note that camera-embedded mobile phones are employed in both the DietSense and WayMarkr projects for diet monitoring and experience recall. The SenseCam device is a sensor-augmented wearable camera designed to capture a digital record of the wearer's day by recording a series of images and capturing a log of sensor data. SenseCam has two of the main components of its operation which are sensing its environment and using a built-in still camera to record images. It has been shown recently to be effective in supporting recall of memory from the past for memory-impaired individuals [148]. Due to its advantages of sensing capabilities, light weight and unobtrusive logging with long battery life, we employ SenseCam as the visual recording device in our work, as shown in Figure 2.1. More details about SenseCam will be given in Section 2.2, Chapter 2.

1.1.2 Typical Applications of Lifelogging

Due to its various advantages, lifelogging may be needed in many areas to satisfy the needs of different groups. The typical applications of lifelogging, especially visual lifelogging can be summarized as an automatic diary, a tourism guide, a memory aid, for diet monitoring, for ADL analysis, or for work-related recording and so on. The details on some of these are as follows:

Digital diary: As we described above, in traditional diary writing or blogging, the documentation is usually carried out manually and involves material choices. The selection of contents and inclusion of value choices need to be considered to decide what is important and worth recording in the diary. An efficient lifelogging recording and summarizing tool could fulfill this task automatically, in addition, with heterogenous multimedia data. To deal with the very large personal data collection, intelligent techniques are necessary for structuring, searching and browsing of this collection for locating important or significant events in a person's life. In [96], three stages are identified for the construction of a digital diary as the processes of capturing and structuring SenseCam images, for example, and then displaying them to an end user to review. In [61], an animated slideshow composed of SenseCam images is presented as a form of a lightweight story telling, along with associated location information recorded by GPS. The main challenges and considerations are also discussed in [35] to archive meaningful autobiographical digital information from lifelog collections. In [47, 53], image features are explored in conjunction with sensor readings such as accelerometer data to cluster a day's worth of SenseCam images into meaningful events allowing quick digital diary browsing.

Tourism Guide: Lifelogging technologies can be adopted in tourism applications as many lifelogging systems have enabled the capability of location sensing. Real-time location tracing can be used to provide many services depending on the recognition of wearer's context semantics. [60] presented an architecture and implementation of a mobile system, called Micro-Blog, for global information sharing, browsing, and querying. A scenario is also illustrated in [60] for the interaction with the system in the application of tourism, by playing audio-visual experiences shared by tourists. In [172], the area of tourism for SenseCam is highlighted, which is then followed by [26], in which museum experience enhancement is explored with museum artifact images taken by SenseCam.

Memory aid: Memory aid is a potential medical benefit which can be supported by lifelogging technologies. By recording various aspects about our recent daily activities, lifelogging will offer an approach for wearers to re-experience, recall or look back through recent past events. In [74], a user study with a patient suffering from amnesia is conducted with SenseCam images and highlights the usefulness of these images in reminiscing about recent events by the patient. In [148], evidence is found that SenseCam images do facilitate people's ability to connect to their recent past. The authors argued that lifelogging systems capture a set of cues (data) which can trigger the remembering of human experience, rather than capture the human experience. In [65], the challenges faced with an extensive period of Human Digital Memory (HDM) generation (2 years and 2 million images) are presented and architectural requirements for managing such archives are also illustrated. Similar applications of turning lifelogging into a short-term memory aid can also be found in [23], [165] and [164].

Diet monitoring: Diet monitoring is another application of lifelogging for medical purposes. Though dietary patterns have been proved as a critical contributing factor to many chronic diseases [136], traditional strategies based on self-reported information do not fulfill the task of accurate diet reporting. More usable and accurate ways to analyze dietary information about an individual's daily food intake are badly needed. Visual media like images and videos provide hugely increased sources of sensory observations about human activities among which food intake can be monitored for diet analysis. The application of visual lifelogging in diet monitoring can support both patients with obesity and health care professionals analysing diets. DietSense [136] is an example of such a lifelogging software system using mobile devices to support automatic multimedia documentation of dietary choices. The captured images can be *post facto* audited by users and researchers with easy authoring and dissemination of data collection protocols [136]. Professional researchers can also benefit in performing diet intake studies with the help of lifelog browsing and annotation tools. Both audio recorders and cameras are combined in [82]. Their usual practices suffer from under-reporting because some subjects were not confident with the use of a tape recorder and camera. Other research into diet recording by employing camera-equipped mobile devices, such as personal digital assistants (PDAs) or mobile phones, can be found in [166], [57], [88], etc.

ADL analysis: The analysis of activities of daily living (ADL) is another application of lifelogging. More concerns is now being shown in modern society about the individual health and well-being of everyday life. However, any long-term investigation into daily life comes across lots of difficulties in both research and the medical treatment area. Occupational therapy aims to analyze the correlation between time spent and our actual health, and there is a growing body of evidence indicating the relationship [94, 111]. Observational assessment tools are needed to correctly establish care needs and identify potential risks. Long-term daily routines and activity engagement assessments are necessary to evaluate the impact on activities of daily living caused by diseases or old age, hence to provide a proper programme towards the needs of each patient. While traditional self-reporting or observational measures are time-consuming and have limited granularity, lifelogging can provide an efficient approach to providing broader insights into activity engagement. Lifelogging is a technology to automatically record everything happening to us, hence it can provide an accurate way to measure activity engagement and affecting factors. Project IMMED [112] is a typical application of lifelogging to ADL, the goal of which is assessing the cognitive decline caused by dementia. Audio and video data of the instrumented activities of a patient are both recorded in [112] and indexed for medical specialists' later analysis. In [84], a wearable camera is used to capture videos of patients' activities of daily living. A method for indexing human activities is presented for studies of progression of the dementia diseases. The indexes can then be used for doctors to navigate throughout the individual video recordings in order to find early signs of the dementia in everyday activities. The same rationale is also reported in [113].

Besides the above described areas, lifelogging can also be applied in others areas like education [17, 58], work-related task observation [33, 91], accessibility within business [85], and so on. The main challenge of applying lifelogging to all above areas is how to access and manage everyday activity media, that is how to build an efficient index for activity retrieval and interpretation. This is discussed in the following sections.

1.2 Research Questions

The application of lifelogging, especially visual lifelogging, in activity analysis imposes challenging problems to multimedia data retrieval due to the large volume of lifelogging data. In addition, a large part of the data are repetitive due to the nature of activity engagement with repeated images of the same or nearly the same thing. Undersampled visual images such as traditional stills camera images have the drawback that the resulting photographs may end up being quite staged rather than forming a simple record of events as they happened [74]. While digital video can be applied in lifelogging for activity recording, continuously recording of digital video will come up with more issues like large data volume and privacy problem. Unlike a regular digital camera, SenseCam has a number of different electronic sensors built in, which can be used to automatically trigger a photograph to be taken when certain changes in sensor readings are detected. The internal timer will also be used to trigger photograph capture at the rate of every 30 seconds. This rate helps to decrease data volume while the details of activity engagement can also be recorded. More important, more interesting changes in the wearer's environment like a significant change in light level, or the detection of body heat in front of the camera can be used to trigger the capture. This capturing rate has been demonstrated to be efficient in various application such as memory aid [148, 23], life trait analysis [52], etc. and we will also employ this capturing rate control mechanism. It is important to realize that to record every activity at this rate will also generate a large amount of data for a single typical day, not to say for a longer term, for example, a month or even a year. Here, a typical day's digital log means to record every activity the wear carries out on that day, for example, from going to work until before preparing to go to bed. Without efficient indexing and retrieval tools, the user might have to look through these images one by one, just to find the event of interest ! Definitely, nobody can afford such a huge and tedious effort. Besides, the movement of the wearer and the resulting poor quality visual data make it difficult to automatically categorize and index the media.

Text retrieval is a large branch of information retrieval and traditional text-based searching principles have been well founded since they started in the early 1960s. The task of text-based retrieval is to match the user query against a set of free-text records, which are organized as documents like newspaper articles, web pages, video manuscripts and so on. The very successful technologies in text retrieval like term weighting [8], the Vector Space Model [146], the Language Model [131], PageRank for assigning importance based on links [126], to name a few, are adopted in many applications. Furthermore, text retrieval has been proved to be efficient on a large scale by current Web search engines such as Google [4], Yahoo! [5], Baidu [6], Bing [7], etc., in which text-based retrieval is the fundamental basis. In multimedia retrieval, image or video data is still indexed by text fields which are called metadata. One way to add metadata is by user manual annotation. However, this approach is not realistic for large volumes because it is tedious and time consuming. Besides, consistent manual annotation for unstructured daily media is impossible and the text-based retrieval technologies can not provide search engines with high quality for multimedia data. Another way to add metadata is to associate textual descriptions with the multimedia

[4] http://www.google.com

[5] http://search.yahoo.com

[6] http://www.baidu.com

[7] http://www.bing.com

content by automatic approaches such as recognition and classification, automatic speech recognition (ASR), closed captions, and text in video (using optical character recognition) OCR text. However, for lifelogging, these indexing technologies can not be directly applied or at least can not perform as well as in the TV news broadcasting domain for example, when the multimedia data are not well edited and are affected by poor image quality and high visual diversity. Therefore, the textual metadata extracted from multimedia is usually scarce and noisy, and so is far from being enough to satisfy the retrieval use in lifelogging.

Due to the explosion of multimedia quantities such as archived TV broadcast videos, various multimedia resources released on the Internet, intense work in multimedia retrieval domain has aimed to provide efficient and accurate functionality for users to access the desired information. Content-based multimedia retrieval utilizes the low-level perceptual features for multimedia queries. These low-level features used can be extracted from different modalities, for example, textual features [29, 7] obtained from closed captions, speech recognition which can be applied to videos, or images features [62, 105, 24] like color, texture, edges, etc. In content-based multimedia retrieval, these low-level features are extracted from multimedia objects and mapped directly to user queries. The notion of concepts is handled implicitly in content-based retrieval as described by [156]. The semantic gap between low-level features and user expectation still exists and turns out to be the focus of concept-based multimedia retrieval. In concept-based retrieval, a set of concepts are first detected by statistical approaches which build mappings between low-level features and concepts. Then the detected concepts are fused for more complex retrieval topics [37, 124, 156].

The performance of automatic detection of concepts in image and video data has been improved to a satisfactory level for some generic concepts like indoor, outdoor, faces, etc. on high quality data from broadcast TV or movies. The progress in the development of semantic concept detection for videos has been witnessed in the annual TRECVid benchmark [152]. Every year, TRECVid benchmark activities not only provide a large set of annotated video samples, but also provide an evaluation campaign in which dozens of research groups can measure the performance of their retrieval systems using the same metrics and data collection. As reported in [99], the automatic detected concept in the TV news broadcasting domain can already be scaled to 1000+, for which 101 concepts are defined in [159] and 834 in [119]; 491 concepts are detected in [157], 374 in [39] and 311 in [99].

However, the large effort in the news broadcasting domain can not be applied directly, at least not very well, to the everyday activity retrieval domain. Among the above mentioned semantic lexicons, the Large-Scale Concept Ontology for Multimedia (LSCOM) is the most comprehensive taxonomy developed for standardizing multimedia semantics in the broadcast TV news domain [119]. The construction of LSCOM tries to answer the question: What set of semantic concepts should the community focus on as it explores new automated tagging techniques ? [119] In the concept selection procedure, the LSCOM consortium tried to bring together experts from multiple communities like multimedia, ontology engineering and others with domain expertise. Multiple criteria are also considered which are utility, coverage, feasibility, and observability [119]. As a framework, the LSCOM effort also produced a set of use cases and queries along with a large annotated data set of broadcast news video. But many of the LSCOM concepts, for example weapon, government leader, etc., are never useful or even encountered in the lifelogging domain so while the hierarchical structure of LSCOM might have been useful, the actual concepts were not. This gives rise to our first research question in this thesis.

(RQ1) What concept ontology needs to be defined to satisfy the needs for indexing everyday multimedia in lifelogging ?

Modern multimedia retrieval approaches index data with a predefined lexicon and enable semantic reasoning on the concept set to facilitate high-level user queries. To bridge the sensory gap between raw data and user expectations, a set of concept detectors is usually developed to represent the high-level metadata. In the retrieval procedure, user's query is broken down into a group of concepts which can reflect the query semantics. A ranked list of results, for example shots for news broadcasting, are returned based on the confidence of the concept detector. In order to provide satisfactory retrieval performance, we need to solve the problem of mapping ambiguity between everyday activity and concepts. This leads us to another research question:

(RQ2) How can we automatically select proper concepts for a given activity topic ? How can we perform semantic reasoning in the lifelogging domain ?

It is important to realise that a single lifelog event such as sitting on a bus, walking to a restaurant, eating a meal, watching TV, etc. consists of many, usually hundreds, of individual SenseCam images. In the case of sitting on a bus, where there is little movement by the wearer, most SenseCam images are the same whereas cooking, for example, where the wearer is moving around, generates a larger range of dissimilar images. This is very different from pre-edited multimedia such as broadcast TV news video or movies for which the frames in the same shot are visually very similar. The visual diversity of lifelog media gives rise to the difficulty of accurate concept detection, and furthermore the burden of activity detection and semantic representation. The corresponding research question which frames this is:

(RQ3) How can we classify different activities and represent them when there are severe visual diversities ?

The semantics we can infer from lifelog media is usually limited compared to the proliferation of online knowledge resources such as Wikipedia, Facebook and Flickr to name a few. The development of modern Semantic Web technologies makes it easier to use the large amount of online data repositories. The Resource Description Framework (RDF) is the Semantic Web formalization language optimized for information sharing and interchange. RDF models each statement as a triple consisting of three parts: subject, predicate, and object. While the web is extended with a data commons by publishing various open datasets as RDF on the Web and by setting RDF links between data items from different data sources, by September 2010, these datasets consisted of over 25 billion RDF triples, which were interlinked by around 395 million RDF links. The standardized data representation could facilitate the enrichment of lifelogging activities. Before we build efficient semantic enhancement application, we have to address another research question:

(RQ4) How can we enhance the semantics of lifelogging activities using Semantic Web technologies ?

These four research questions help us to formulate an overall hypothesis for our work, namely that "Semantic Web technologies can support the interpretation of event semantics in lifelogging". This hypothesis reflects the notion that we will use Semantic Web technologies in our work of mining lifelog event semantics. However, this does not mean we will only use the technologies from the Semantic Web domain to address our research questions. On the contrary, Semantic Web technologies will be assimilated with other multimedia retrieval technologies in our work. We use the word 'support' in our hypothesis with the meaning that Semantic Web technologies can be brought into the process of event analysis in lifelogging and achieve satisfactory performance in semantic interpretation.

1.3 Thesis Structure

The above proposed four research questions and overall hypothesis are addressed in the following chapters in the thesis. The thesis expands the research questions with an overview of current research methodologies on lifelogging and multimedia information retrieval. Then the development of new algorithms and the modeling of research problems are described in detail as well as the demonstration of our experiment results and application performance.

Chapter 2 gives a brief background description of state-of-the-art methodologies in lifelogging research and multimedia retrieval. The prevailing semantic indexing and annotation procedures are discussed to illustrate the potential benefit of concept-based multimedia retrieval applied to lifelogging. In addition, the difference between the lifelogging domain and traditional multimedia are compared to realize the new challenges in lifelogging retrieval. The hypothetical semantic interpretation hierarchy underlying our research is also briefly introduced at the end of the chapter.

Chapter 3 first investigates everyday activities and elaborates the selection of target activities for our lifelogging semantic analysis. Then a density-based semantic concept selection algorithm is introduced to utilize concept similarity reasoned from ontologies. The concepts are then re-ranked with candidate concepts selected by agglomerative clustering, used as seeds. In this chapter, semantic reasoning on prevalent lexical and contextual ontologies are also discussed.

Chapter 4 elaborates our user experiment to generate a set of concepts in regard to everyday activities, and then demonstrates experiments on semantic density-based concept selection algorithm. Various ontological similarity measures are compared based on the performance of concept selection. The evaluation is first carried out on everyday concept selection in lifelogging. To test the generality of our algorithm, we also assessed the performance on a concept set defined in the TRECVid benchmark, which focuses on the TV news broadcasting domain. The efficacy of our algorithm in semantic reasoning by ontologies and selection of relevant concepts, is shown by experiment results.

Chapter 5 addresses the issues of everyday activity detection and event-level concept fusion. A concept-based activity detection algorithm is proposed in this chapter, modeling the temporal dynamics of concept appearance with a HMM-based approach.

The performance of our activity detection algorithm is demonstrated by assessing on concept detectors with various levels of detection accuracy. To utilize the concept relationships for semantic fusion, the ontological multi-concept classification is also explored, followed with the interestingness-based concept aggregation for events. Semantic concept interestingness is calculated by fusing image-level concepts which are then exploited to select a representation for the semantic event correlated to various event topics. The efficacy of our algorithm is shown in fusing semantics at the event level, and in selecting event representations in visual lifelogging.

Chapter 6 starts with the modeling of events as an ontology in a multi-context point of view. Each event is modeled as an instance of event ontology and formalized with prevailing ontologies to incorporate context semantics extracted from raw sensor readings. Event semantic enhancement in this chapter is based on this lifelogging event model to query most relevant semantics from online knowledge repositories of linked open data through Semantic Web technologies. The enriched event semantics is demonstrated and evaluated in this chapter to show the efficacy of this enhancement methodology.

Chapter 7 ends this thesis with some conclusions as well as future avenues for later research.

Chapter 2

Background to Semantic Interpretation of Lifelogging

2.1 Introduction

It has become more and more practical for researchers to investigate the underlying patterns of our daily lives following the development of computer networks, large volume databases, machine learning technologies and the wide deployment of computing devices. Especially, many lightweight devices such as the mobile phone are endowed with sensing capabilities through built-in cameras and other heterogeneous sensors. These widespread mobile devices have already formed an infrastructure to gather data and allow us to mine the patterns of human life and social characteristics.

The vision of using technology to record everything that happens to us is called lifelogging. Steve Mann is a pioneer who tried to capture what he saw through video cameras mounted on his head [108] and these have evolved from 'chunky' head-mounted cameras to discreet recorders built into eyeglasses. Microsoft Research in Cambridge have used the SenseCam to capture everyday life and have evidence that these images can improve peoples' memory abilities [148]. In MIT, an experiment was carried out using Bluetooth-enabled mobile telephones to measure information context in order to identify the deep social patterns in user activities [55]. In [164], Vemuri and Bender presented a memory re-finding use of lifelogging which is called "iRemember". In their research they recorded audio clips as the main information used to navigate memory. In [127] this kind of technology is also employed to provide real-time transportation information to individuals with mild cognitive disabilities and improve efficiency and safety as well. Mobile phones and other kinds of digital devices are very popular nowadays and form a large computing resource and an ubiquitous infrastructure for our digital life. The DietSense project [136] at UCLA makes use of a mobile phone with a camera embedded to capture pictures automatically. The images collected as the log of a wearer's mealtimes are used to analyze the diet intake in order to give feedback and to improve diet choices. The WayMarkr project at New York University also makes use of a mobile phone affixed to a strap to take pictures automatically [30]. Furthermore, social dynamics are studied in [55] by using mobile Bluetooth as the measure in lifelogging. Although they are successful in solving some design considerations, the algorithm for detecting contexts lacks flexibility which can not adapt to the semantics of contexts dynamically with the limited use of context-awareness. Context information is not fully used to receive more flexible approaches of context classification and recognition for labeling the semantic meaning of the user events.

What all this literature points to is a very active community in lifelogging, exploring a range of techniques and using a variety of lifelogging devices. Yet lifelogging needs to be about more than just the capture technology used to capture the lifelogs, it needs to be about the techniques used to analyse the lifelogs and provide search and browsing and navigation through those lifelogs. Thus indexing and retrieval are just as important as the lifelog capture devices.

In order to manage accumulated lifelogs we need clever information management, and much of the related work has been done in multimedia retrieval where low-level feature-based multimedia queries using image features such as color, texture, edges and other attributes have been studied extensively. However, there is no means to reflect the coincidence between features extracted from visual data and the interpretation that they have for the user in a given situation [154]. Bridging the gaps between different levels of semantics is the challenge for researchers in content-based information retrieval. In multimedia information retrieval, state-of-the-art techniques use statistical approaches to map low-level features to concepts which are then fused to relate to high-level query topics [156]. The whole task is generally broken down into two steps: the detection of a set of concepts and the association of concepts with queries. This modern methodology facilitates an understanding of topic queries and low-level features by analyzing the mapping in a semantic way. To build a large-scale ontology and lexicon for semantic gap filling, large efforts are done for activities like LSCOM (Large-Scale Concept Ontology for Multimedia) [119, 6], TRECVid [152] and MediaMill's 101 concepts [159]. According to the TRECVid benchmark [152], acceptable results have been achieved already in many cases particularly for concepts where there exist enough annotated training data. Based on concept detection, encouraging improvement has been reported showing the efficiency and the effectiveness of concepts for higher level retrieval [156, 124].

Semantic Web technologies have developed in recent years with the goal of modeling the semantics in a machine understandable approach. Due to the standardized format and capacity of efficient semantic description, ontology modeling is employed in providing a concrete semantics for information retrieval. In [117], conceptual model and annotation ontology are used for video representation and retrieval. A large-scale concept ontology has been developed for standardizing multimedia semantics in the broadcast news domain. As a framework, the LSCOM effort also produced a set of use cases and queries along with a large annotated data set of broadcast news video [119]. In topic-related retrieval, Yang [179] has tested different measures in video shot retrieval. The results shows that difference in topics/tasks can vary the measured performance. The concepts detected by classifiers are usually fused for topic-related filtering. However, the accuracy of detection will affect the utility of filtering which shows the high demand from classification accuracy [43]. In [104] and [75], an ontology for video retrieval is addressed while an image retrieval ontology is investigated in [167]. As a hierarchical ontology database, WordNet is used in [76] and [75] to couple image analysis and concept detection in the real world by creating links between visual and general concepts. The entities in WordNet are thus extended with image properties to build a mapping of perceptual elements and concepts. Ontologies which contains visual information can then be formed to facilitate annotation for broad domain requirements. Similarly, Snoek [156] tries to build a direct link between generic concept detectors and WordNet synsets. Besides general purpose ontologies like WordNet, some specific domain ontologies have also been built and show effect in representing domain concepts and relations in a formalization of a semantic network. The usefulness is also investigated in image or video retrieval by importing domain ontologies into information matchmaking which involves the combination of text description and image features [167]. In [104], a domain-dependent concept ontology is built to enable multi-level modeling of semantic video concepts for medical video retrieval.

What all this work represents is a considerable effort in building and using ontologies in the task of (visual) multimedia information search. Mostly, ontologies have been useful assets in the search task but their drawbacks are in the large efforts needed in constructing them, and the fact that there isn't a single best way to use them in retrieval.

In the rest of the thesis, we will provide details of our work in developing our approaches to lifelogging and dealing with such issues as concept selection, concept detection, semantic event interpretation as well as enhancement of user events. During the description, state-of-the-art technologies will be compared and our further working plans are also discussed in company with details for experiments and evaluation.

2.2 Multimodal Context-Awareness

As an integrated part of our lives, our contexts are changing dynamically and if we can capture some parts of these contexts then these can be used as cues for our activities. By ‘contexts’ we mean the features of where we are, who we are with, what we are doing and when we are doing it. Since the context includes various aspects of the environment in which the user interacts with digital devices, the plurality of context can be applied intelligently to detect meaningful changes in the environment. The increasing adoption of sensors for mobile phones makes it possible to gather more context information on handset devices which is important raw material for creating an automatic diary for example. This kind of application of heterogenous sensors in context sensing is named as multimodel context-awareness. Based on the collection of low-level sensor information we can infer cues about the host and the environment. The contexts can then be derived from cues to compose the diary.

To present a meaningful reflection of daily life, we must detect and interpret implicit semantics of lifelogging data from heterogeneous contexts. To determine contexts, a large body of information is needed. We believe that the location is not sufficient in the analysis of a dairy because it can not fully explain the Who, What, Where and When questions which is the common form of everyday events. We adopt the four primary types of context information raised by Dey *et al.* [49] as the fundamental information to generate a diary, namely location, identity, time and activity. This four-dimensional context structure can well depict the Who, What, Where and When application of the diary. Additionally, a more detailed understanding of a situation can be retrieved or generated by integrating these contexts.

Motivated by the above issues, we use SenseCam (shown in Figure 2.1) as the main wearable device in our research. SenseCam is a lightweight passive camera with several sensors built-in. It captures the view of the wearer with its fisheye lens which helps to capture more in the view than the normal lens. The pictures are taken at the rate of about one every 50 seconds without the trigger of other sensors. The onboard sensors can help to trigger the capture of pictures when sudden changes are detected in the environment of the wearer.

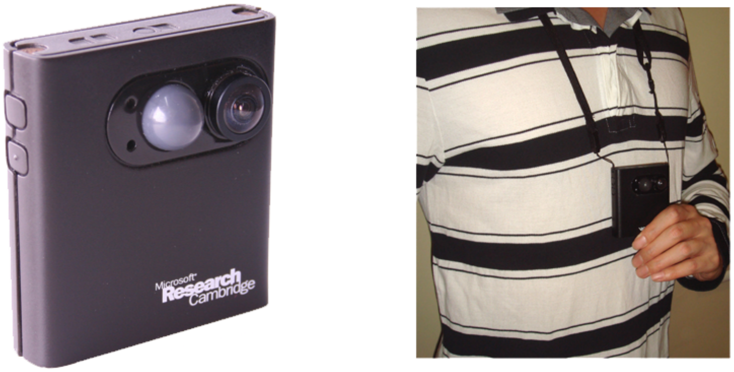


Figure 2.1: The Microsoft SenseCam (right as worn by a user).

Quite different from traditional video and image processing, processing lifelogged data involves numbers of sensors which can generate a large amount of heterogeneous data. Take SenseCam for example, temperature and acceleration are sensed and stored together with the images captured by the built-in camera. Using SenseCam one can collect up to 2,500 images in a typical day. With a detection frequency of 0.1Hz, each user could gather about 6,000 GPS records and 3,000 Bluetooth detections each day, as well as about 16,000 accelerometer records. This large amount of multi-source data poses a challenge to detect more important and interesting events in life. Besides, different sensor data reflect different aspects of a user's life and the low-level characters do not have an explicit relationship with high-level event semantics. How to fuse the contexts to get a meaningful representation of daily events is a challenge. Moreover, in lifelogging, devices and sensors continuously capture and store the context of the wearer, making event detection more difficult. In a SenseCam for instance, visual information for a user will be collected every about 30 seconds without interruption after the device is on. This is quite different from common personal digital photos taken by ordinary cameras. Using an ordinary digital camera, the time gap or long physical distance between consecutive photos can often be used as an indication of a new event, which works very well in work by Platt *et al.* [130] and by Naaman [118]. However, these kinds of cues are scarce in lifelogging.

2.3 Multimedia Semantic Retrieval

Due to the generally unstructured characteristics of multimedia data, there are more challenges in returning satisfactory result according to a user's expectation. Annotation and indexing are both necessary for flexible retrieval. One efficient way of adding information that describes the semantics of multimedia objects, is to use information *metadata* [25]. Rather than searching the raw media, searching on the metadata using standardized word-based retrieval makes thing much easier. Besides, the storage of metadata is much reduced compared to, say raw video.

In lifelogging, there are two ways to obtain descriptive metadata for everyday logged media: manual annotation and automatic indexing. Manual annotation is an non-automatic way to add textual information for media. Considering the fact than there might be up to 2,500 SenseCam images captured in a single day, it is not possible for a user to annotate such a large volume of data. In addition, manual annotation also suffers from it subjectivity, inconsistency and incompleteness, making it's later usage difficult and unpredictable. The automatic construction of metadata for multimedia is quite desirable and is now described.

2.3.1 Feature Extraction and Representation

In the multimedia domain, features are used to derive metadata from raw media data. The process to capture features from a multimedia object is called feature extraction [25]. Feature extraction and derivation of metadata from features are often carried out automatically, therefore are preferred in our semantic interpretation of lifelogging. Two levels of features are usually distinguished to reflect the extent to which the feature is related to media semantics, namely low-level features and high-level features.

2.3.1.1 Low-Level Features

Generally speaking, low-level features refer to data patterns and statistics which contain less meanings than textual description about media content. Because low-level feature extraction is a totally mathematical computation, it can be done automatically. Take text documents for example, where low-level features can be derived from the frequency of each word appearing in the documents, removing stop words like 'the', 'a', 'it', etc., which do not contribute to expressing the semantics of the document. The equivalent widely used low-level features for image and video include average energy, zero crossing rate ZCR, and silence ratio, etc. for audio; color, texture, shape, etc. [25].

Although low-level features are not usually directly used for retrieval, more meaningful features can be built on top of them by further analysis. The advantages using low-level features can be summarised as:

- they are representative: Compared to raw image input, the low-level features can represent aspects of the characteristics of image more accurately.
- they require lower storage: The storage of already extracted low-level features requires much less space than that of raw image pixels.

- they could be used for dimensionality reduction: The extraction of low-level features can help to reduce the computational dimensionality since the processing of raw image pixel array is usually high-dimensional. Meanwhile, some other technologies like Latent Semantic Indexing (LSI), Principle Component Analysis (PCA), etc. can also be applied on top of extracted features to assist dimensionality reduction.

- they have less computational expense: The reduction on dimensions allows the comparison of two feature values much easier and quicker.

Image media is the most important source for SenseCam-based lifelogging and we now elaborate on the kind of features used in image representation.

Color Features

Each image is constructed from a specific number of pixels. As each pixel has a color value (gray-scale for black-white image) within a range of color, color features [62] can characterize the content of images.

Color Histogram: *Color Histograms* reflect pixel distribution across discrete color values. The histogram is calculated by simply counting the number of pixels having a color value within a given set of color ranges. Color histograms are widely used for distinguishing images by visual similarity.

Scalable Color: *Scalable Color* is another descriptor which measures color distribution over an entire image. The color space is fixed to HSV to calculate Scalable Color, quantized uniformly to 256 bins, including 16 levels in H, four levels in S, and four levels in V. The histograms of Scalable Color are encoded based on Haar-transform in order to reduce the large size of this representation, while allowing scalable coding [160].

Color Layout: Like Colour Histograms and Scalable Colour, *Color Layout* is designed as an MPEG-7 visual descriptor to capture the spatial distribution of color in an image or an arbitrary-shaped region. It is a compact and resolution-invariant color descriptor defined in the YCbCr color space. Color Layout uses representative colors on an 8×8 grid followed by a DCT and encoding of the resulting coefficients. A few low-frequency coefficients are selected using zigzag scanning and only 6 coefficients for luminance and 3 for each chrominance are kept, forming a 12-dimensional vector for Color Layout [160, 107].

Texture Features

Similar as color features, texture features are another kind of low-level descriptors for image search and retrieval which can be extracted automatically. Texture descriptors consider an image as a mosaic of different texture regions [106], and the image features associated with these regions are then used for image search. Three texture descriptors are considered in MPEG-7, which are *Texture Browsing*, *Homogeneous Texture* and *Edge Histogram* respectively. As described in [107], all of these descriptors are calculated when there exist patterns such as homogeneous regions, dominant orientations, etc. in an image.

Shape Features

Image shapes are usually represented by a set of point samples extracted from shape contours for example about 100 pixel locations sampled from the output of an edge detector. No special requirements are needed for these representative points, that is, they are not necessarily required to be landmarks or curvature extrema, etc [20]. Shape-based features utilize shape boundary or entire shape regions to capture local geometric characteristics within an image. A Fourier descriptor is a representative of a boundary-based shape feature while moment invariants use region-based moments which are invariant to transformations [143]. Shape context is another shape descriptor used to describe the coarse distribution of the rest of shape points with respect to a given point. It has been adopted recently in such applications as human action recognition [46], trademark retrieval [144], etc. The comparison of two shapes can then be extended to finding sample points with similar shape contexts from both shapes.

2.3.1.2 High-Level Features

High-level features refer to features which are semantically meaningful for the end user. While low-level features are never readable by the end user, high-level features can express the semantics of media in a more acceptable way as ‘concepts’, such as ‘indoor’, ‘outdoor’, ‘vegetation’, ‘computer screen’, etc. These features can provide a meaningful link between low-level features, and user expectations. The extraction of high-level features demands filling the gap between low-level features and high-level features, which is called the *semantic gap* in multimedia retrieval.

Semantic concepts are usually automatically detected in a mathematical way by mapping low-level features to high-level features. The state-of-the-art approach is to apply discriminative machine learning algorithms such as Support Vector Machines (SVMs) to decide the most likely concepts given the extracted features [156]. Compared to a discriminative model which is more task-oriented, generative statistical models such as Markov model try to analyze the joint probability of variables, which are also proposed in concept annotations [98]. Both generative and discriminative approaches have their own pros and cons. A generative model is a full probabilistic model of all variables whereas a discriminative model has limited modeling capability. This is because a discriminative model provides a model only for the target variable(s) conditional on the observed variables hence can not generally express more complex relationships between the observed and target variables. However, discriminative models are often easier to learn and perform faster than generative models. Besides, it has been shown that discriminative classifiers often get better classification performance than generative classifiers with large training volume (usually including positive and negative samples). Among these machine learning algorithms, SVM is an efficient discriminative approach with strong theoretical foundations and excellent empirical successes in many tasks like handwritten digit recognition, image retrieval and text classification, etc. [97] It has been demonstrated to be an efficient framework by many research groups in concept detection [99, 38, 157] and we will also employ SVM as the base classification algorithm to perform the task of concept indexing. The *learning for classification* models using these technologies always involve a large corpus of annotated datasets. It is impossible to build concept detectors for all possible concepts and it is still challenging to build detectors which can cross application domains. Current solutions for multimedia content retrieval focus on specific domains. For instance, the LSCOM concept ontology and MediaMill's 101 concept detectors mentioned earlier in Section 2.1 are all focused on the TV news broadcasting retrieval domain. In this thesis we will analyze high-level features needed in an everyday visual recording domain, for example SenseCam images.

2.3.2 Content-based and Concept-based Retrieval

Since low-level features can be extracted automatically from media objects, the comparison between media objects based on these features leads to content-based retrieval. Using a content-based query, a multimedia system handles the notion of concepts implicitly. The low-level features are assumed to correspond to the semantics of the query while the mapping is not modeled. Color, texture, shape, etc. are the features widely employed for content-based retrieval [62, 105, 24]. For video retrieval, the text features from spoken dialogue, closed captions, etc. [29, 7, 171, 44, 176, 123, 87] are also employed in combination with image features for content retrieval.

More recently, much research has shown the limitations of content-based retrieval which fails in conquering the semantic gap purely using low-level features. The introduction of high-level features shows the advantages in filling, or at least reducing the semantic gap. Retrieval based on high-level features is called concept-based retrieval.

Concept-based retrieval handles the notion of concepts explicitly by expressing user queries in terms of high-level concepts rather than low-level features [156].

There are two categories of high-level feature detections, namely *dedicated approaches* and *generic approaches*. Dedicated approaches aim to grasp the direct mapping from low-level features to high-level concepts specifically used in different domains [90, 155, 142]. These approaches are rule-based, hence for a new concept a new mapping rule needs to be developed. The diversity of too many specific methods or dedicated approaches is addressed by adopting generic approaches [120, 12, 56, 158, 162]. A pool of concept detectors can be learned for concept-based retrieval as lexicons enriched with general-purpose vocabularies such as WordNet or domain-specific ontologies. Satisfactory results have been reported in research using the generic machine learning paradigm, particularly for parts of concepts and related tasks when there exist enough annotated training data [152]. Based on high-level features, retrieval has been proved to be successful in recent research through the correct description of user query with the appropriate concept detectors, which is also called concept selection [156, 124].

2.3.3 Query Expansion / Concept Selection

Query expansion is an approach to searching which is well developed in the application of document retrieval. The intuition of query expansion is to improve retrieval performance by adding more words to the query thus making it more explicit. In concept-based multimedia retrieval, a similar intuition holds, namely that by automatically expanding the set of concepts which are detected in a video clip or query through adding more which are "about" the same material, the expansion usually results in a more precise representation. In theory, the query-concept mapping is responsible for translating user expectation to a set of concepts, in a process called concept selection. Though recent trends show that the generic methods can learn concepts from a large manually annotated corpus, it is still unrealistic to build a group of concept detectors which have a comparable number to human vocabularies. Textual approaches [156] and collection-based statistical methods [101, 69] can be used for concept selection while recent research shows favoring ontologies to select relevant concepts [156, 169]. More details about the selection of concepts will be given in Chapter 3.

2.3.4 TRECVid vs. Lifelogging

In analyzing concept classifiers to be constructed, attempts have been made to deal with the issue of classifier scalability. LSCOM [6] was developed as a popular multimedia ontology for concept classifiers in the TV news video domain. In [6], the concepts were narrowed down to a set of 449 unique concepts to construct a lexicon for multimedia. The concepts selected cover events, objects, locations, people, and programs. In [121], Naphade *et al.*, the LSCOM concepts are broken down into a 7 orthogonal dimensional space. Finally 39 concepts are chosen in this lightweight lexicon known as LSCOM-lite. They are selected by analyzing their utility in tasks such as searching and detection. The searching terms are mapped to the WordNet hierarchy to find the proper nodes with the right balance of specificity and generality [121]. Concluding from the above mentioned literature, the ontology selection by LSCOM mainly considers the following four requirements [119]:

- **Utility**: The concepts selected should have high practical value in supporting tasks such as semantic searching and queries;

- **Coverage**: The overall semantic space of interest should be covered by the selected concepts;

- **Feasibility**: In defining LSCOM, concept feasibility is examined technically to make sure the concepts can be extracted automatically;

- **Observability**: There must be a large amount of training samples for the selected concepts i.e. a high occurrence frequency of semantic concepts is needed.

In TRECVid, the LSCOM lexicon and ontology is used for the evaluation of the high-level feature (concepts) detection task. Christel *et al.* [42] investigated using oracle selection which means ideal selection approach (manual selection by user experiment) for concept-based strategies and showed its utility for retrieval for topics. In their work, 39 LSCOM-lite concepts and 24 TRECVid 2006 topics were examined based on pooled truth data. Due to the limited number of LSCOM-lite concepts used, more concepts related to specific topics did not work any better than a baseline while non-related concepts were selected for some topics. Only 2 among the 24 topics benefitted from incorporating more than 2 concepts in the retrieval process [42] due to the lack of a sufficient ontological framework and what this shows is the need for more than just a flat set of concepts, i.e. a need for a concept *structure*.

Though the lexicon defined by LSCOM is efficient for concept classification in the video domain, it is still different from the concepts needed in the lifelogging domain, due to the following reasons:

1. **Data structure**: In video processing, each shot is usually represented by a keyframe. The concepts within the specific shot are also detected by classifying the keyframe, as is usually done in the TRECVid evaluation. Compared with video shots, lifelog visual media such as SenseCam image streams are less structured. To make the SenseCam image streams more manageable, automatic event segmentation [53] can be applied first before further indexing of large chunks of SenseCam images. Here we simply define lifelogging event as the occurrence in real world at specific time and place and more detailed definition of event will be given in Chapter 5. By applying event segmentation algorithm proposed in [53], a particular event might be represented by a series of images with longer time intervals between them since on average about 30

seconds elapse between images when using a SenseCam.

2. **Visual diversity**: Compared to frames in a video shot, the successive images in visual lifelogs have greater dissimilarity for many events in terms of visual features. The sets of concepts detected from successive images might therefore have significant differences so the event concepts can not easily be detected just by the keyframe.

3. **Semantic focus**: While the ontology used in video classification is focused on TV news or sports etc., the semantics of lifelogging should be more related to the activities from which we construct our life experience, such as the activities of meeting, shopping, socialising and even travelling.

4. **User Context**: Different users will have different notions in interpreting their event semantics due to the different contexts and users' experiences. There will be more disagreement on semantics in lifelogging than in TV videos for example. Besides this, users will have different preferences, different lifestyles, different activities, which also imposes difficulties on the selection of concepts for lifelogging. One semantic interpretation might not make sense for a particular user if his context is unknown.

2.3.5 Difficulties in Lifelogging Retrieval

For lifelogging, we describe the architecture overview of the semantic retrieval task as in Figure 2.2, from raw data collected for lifelogging events. To generate the high-level concepts for events, the classifiers are employed in the pipeline for context/concept extraction from multimodal data. Many statistical and discriminative models are proposed to seek accurate multimedia information annotation and organization, among which SVMs [31] might be the most popular machine learning algorithm especially within the multimedia community. Even though the classifiers built by machine learning algorithms gain satisfying results in the TV news broadcasting domain, they do have the following limitations:

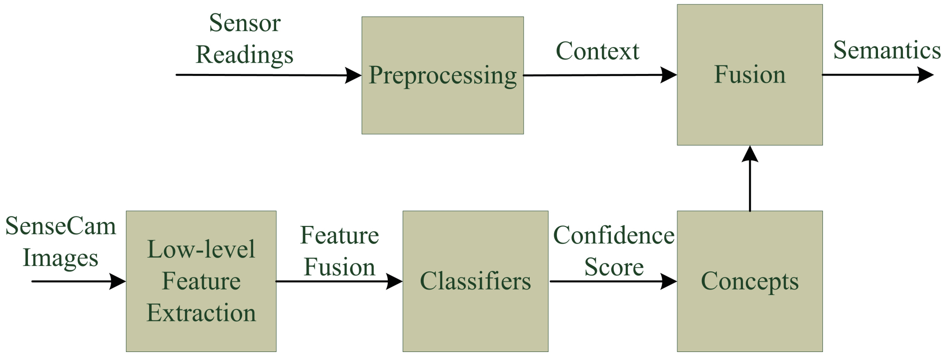


Figure 2.2: Pipeline overview of semantic fusion.

- *Classifier Learnability*: Before providing satisfactory label recommendations it is still sometimes difficult to find many positive training instances for each concept classifiers. Even given lots of positive instances, these must then be highly visually diverse, making the classifiers difficult to construct well if they are to faithfully detect all the wide variety of visual instances for a given concept. Insufficient data/annotation for concepts restricts the classifiers in having good learnability.

- *Classifier Scalability*: To detect a large number of concepts many classifiers are needed, which is not only computationally expensive but also leads to difficult model training problems. Concept selection can help to find the most useful concepts to reduce and minimize the concept set. It is analyzed to deal with the issue of number of useful classifiers.

- *Classifier Disambiguation*: Besides the large amount for concept classifiers, even one concept has multiple meanings which users might use under different context. In [149], disagreement among users on concepts consistency is observed especially on abstract concepts. This also makes the concept selection and classification difficult.

2.4 Event-Centric Media Processing

It's widely accepted that events are the basic and elementary units for humans to organize our memories [170]. Recent research on personal photo organization also show that people often think of their own photos in terms of events corresponding to a certain loosely defined theme such as a wedding, vacation, birthday, etc. [59], [141], also [118]. In modern multimedia processing, events are represented using different presentation forms such as text, images, videos and even some other sensor readings. Across all of these, there is no common model of what makes up an event which is accepted across the field, though it is receiving attention. For example, the ACM International Workshop on Events in Multimedia (EiMM09) which is held in conjunction with the ACM Multimedia conference each year.

Events play an important role in lifelogging because our daily lives are organized as events in our memories, and in addition we also plan and foresee our future life in the form of events. A consistent event model and an event-centric notion in lifelogging are needed to serve as a guide in processing lifelogs and in semantic interpretation of those lifelogs.

In a traditional diary, we write down the meaningful or significant activities or comments from our lives for later review. To generate a digital diary reflecting aspects of users' lives, the main events and especially the most interesting or the most unusual events should be detected and represented as parts of the diary. Harnessed through wearable sensors, data can be used not to only record the main activities in each day but also such data can cover details of events such as the location, people around and the images from the event allowing a reconstruction of the most important of the events in our lives. The reliable and accurate detection and understanding of everyday events and event boundaries can also facilitate better event management and retrieval in a digital dairy.

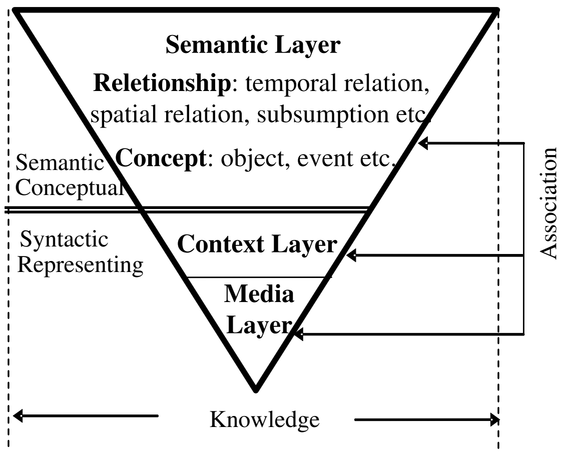


Figure 2.3: Event model and layered structure.

We present our view of lifelogging events and their media contents in the form of a layered structure as depicted in Figure 2.3. The structure includes three layers:

- Semantic layer: This represents the semantic meaning of data. In the semantic layer, concept semantics such as objects, activities, event topics and relationship semantics such as temporal/spatial relationships, equivalence and subsumption etc. are interpreted into higher understanding.

- Context layer: This layer includes the contexts which represent facets associated with events. The temporal and spatial aspects are the basic physical contexts in describing events, i.e. events are spread over the temporal and spatial axes. These two contexts are related to the temporal property and to location awareness in lifelogging. The people involved and the entities related to events and further information about these events are also included in the context layer to answer “Who, What, Where and When” questions about the event together with spatial and temporal contexts.

- Media layer: The physical and formal contents are represented in this layer, such as pixels, sensor data values and coding mechanisms etc. Although semantic meaning exists independently beyond any kind of media, rich media documents are necessary for users to explore a series of events in their lifelogs.

The media layer and the context layer emphasize the sensing and syntactic aspects in lifelogging. However, the semantic layer represents the meaningful aspects which are suitable for our understanding of a logged life. In Figure 2.3, knowledge (depicted by horizontal span) covers all three layers in the model to represent the importance of knowledge content and reasoning in each layer. It is not hard to notice that the amount of knowledge decreases when going from the top to the bottom of the model. The semantic layer on top of the model is richest in knowledge which is reflected by broader knowledge span in Figure 2.3. This means that the semantic layer contains more concepts and relationships inferred from explicit ones. Semantics, abstracted from context, are more decisive than the original pieces of context in understanding the user's situation. These three layers are associated together to provide the structural and experiential needs in generating a digital event lifelog. We can conclude that, to obtain rich semantics for event understanding, the fusion of contexts detected from the media layer is the crucial step and we base the rest of our processing of lifelog data on this premise.

2.5 Summary

In this chapter we present a high-level knowledge background for multimedia information retrieval as well as its application to lifelogging. The overview of aspects for lifelogging is also discussed in this chapter together with related work. As a new form of multimedia, lifelogging media has it own characteristics compared to traditional media such as broadcast TV, in modality, image quality, visual diversity, etc. We also take SenseCam as an exemplar lifelogging device and we analyzed the corresponding difficulties induced by lifelogging retrieval. Finally, our layered event interpretation conceptual model is proposed for context-aware lifelogging retrieval to be used later in the thesis.

In the next chapter we will further elaborate how semantic concepts contribute to understanding events in lifelogs and in particular how the combinations of concepts and the density of those combinations can be used to index events.

Chapter 3

Semantic Density-based Concept Selection

Traditional content-based methodologies for retrieval of image or video try to map low-level features to high-level semantics without bridging the semantic gap. This kind of approach has limitations because of lack of coincidence between low-level features and query semantics as we saw in Chapter 2. This makes concept-based high-level semantic reasoning an attractive solution to satisfy user expectations. Recent research tries to bridge low-level features and semantics with the fusion of concepts to provide better understanding of user expectations, which is known as a concept-based approach to multimedia indexing. In this approach, concepts are first detected by a mapping from low-level features using generic methods from training data. The concepts are then fused together to reason or deduce the final set of concepts which may be used as user query or a representation for multimedia information, whatever the application. The concept-based retrieval framework is illustrated in Figure 3.1.

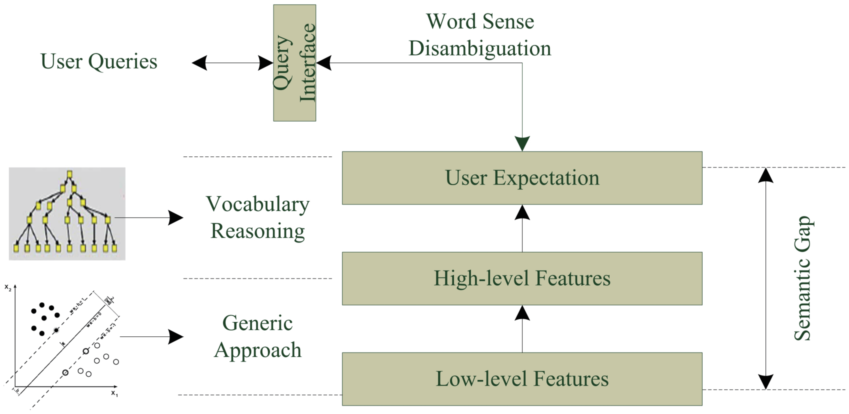


Figure 3.1: Framework for concept-based retrieval

3.1 State-of-the-Art Concept-based Retrieval

Concept-based information retrieval has received much interest from among the multimedia retrieval community due to its potential in filling the semantic gap and its semantic reasoning capability. In concept-based video retrieval, for example, there are methods to expand query terms into a range of concepts and user judgments and feedback can be used to reveal the correlation between concepts. In concept-based retrieval, subjects can be asked to choose the concepts they think are relevant to specific queries. This kind of approach, however, is time-consuming and difficult for a user. It is fine to test on a small number of concepts and queries as is the case in work by Christel *et al.* [41] for which two collections including 23 queries and 10 concepts together with 24 queries and 17 topics are used. Furthermore, the approach tends to suffer from low inter-annotator agreement, as depicted in [41] and [124].

The main automatic approaches to selecting appropriate concepts for semantic querying fall into two categories: lexical approaches and statistical approaches [122]. Lexical approaches leverage the linguistic relationships between semantic concepts in deciding the most related and most useful concepts for the particular application. Statistical approaches apply occurrence patterns from a corpus to reveal concept correlations. Statistical approaches also make use of specific, collection-specific associations driven by the corpus set while lexical approaches depend on global linguistic knowledge. These approaches can be summarised as follows:

Lexical approaches Semantic similarity is used as a measurement to rank the relevance of concepts to a given query text. WordNet is one popular source of such a lexical knowledge base. One straightforward solution to this is selecting the concepts based on minimizing the semantic distance between the concepts and query terms. WordNet-based semantic similarity between query terms and concepts are calculated as the weight of concepts using semantic similarity scores and some of the work in the area goes back many years, e.g. [139] and [137]. In more recent work, the Lesk-based similarity measure [15] [128] is demonstrated as one of the best measures for lexical relatedness and is employed in [67] for lexical query expansion. WordNet-based concept extraction is also investigated in [68] to evaluate the effectiveness of high-level concepts used in video retrieval. [68] shows the algorithm achieved comparable results to user created query concepts. The issue with concept selection when using a lexicon ontology such as WordNet is that the local similarities across branches are not uniform. This could lead to incomparable similarity values obtained from local ontology branches, as argued in [169]. In [156], Information Content is used to calculate similarity in order to deal with the problem of similarity inconsistency caused by non-uniform distances within the WordNet hierarchy. In [11], the names and definitions of concepts as well as relevant Wikipedeia articles are aggregated to generate the text collection for IR system whose output scores are later used in determining the probability of concepts given relevance.

Statistical approaches A large amount of manual annotation effort in the annual TRECVid benchmarking activity for video retrieval [125], and in LSCOM [1] the concept ontology for broadcast TV news, enables the analysis of static patterns for video retrieval. The groundtruth of hundreds of individual concepts and dozens of query annotations is used in comparing retrieval systems as well as selecting and analyzing the relevant concepts associated with particular queries. Mutual information (MI) is used effectively in feature selection especially for choosing discrete-valued features. MI is used in [101] for choosing concepts with high utility in retrieval from the information-theoretic point of view. The probability of a shot being relevant to a query is calculated given the prior probability of the shot already being relevant to the concept.

More recent work by Wei and Ngo [169] proposed an ontology-enriched semantic space model to cope with concept selection in a linear space. The ontological space is constructed with a minimal set of concepts and plays the role as a computable platform to define the necessary concept sets used in video searching. This linear space guarantees the uniform and consistent comparison of concept scores for query-to-concept mapping [169]. We call this concept selection approach an ontological space approach.

Besides the above-mentioned concept selection strategies, "oracle" selection is also investigated in [42] to select the concepts which are most suitable for TRECVid topics. Two benchmarks for concept selection are presented in [78] for video retrieval systems, which are either created by a human association of queries to concepts or are generated from a tagged collection. A user assessment is performed to validate the reliability, captured semantics and retrieval performance and mutual information is used as a measure for ranking the concepts according to their retrieval contribution [78].

3.2 Event Semantic Space (ESS)

A limitation for building classifiers is for them to reveal the higher level semantics of images when they have multiple concepts with high correlation. The concepts involved in lifelogging cover numerous aspects of our daily lives and the choice of concepts is very broad. According to the statistics in our investigation into SenseCam images which is described in more detail later, about 40 concepts have high frequency appearing in lifelog images which typify more than 10 significant high level activities. The detection of all concepts not only increases computational expense but also reduces the annotation accuracy. Meanwhile, the large number of concepts to be annotated incurs a large annotation effort. The interpretation of lifelogging events thus demands a strategy which helps to select the most useful concepts for event representation rather than just using all possible concepts.

3.2.1 Everyday Activities: Exploring and Selection

It's believed that there exists a relationship between everyday activity engagement and well-being for individuals. For a long time research has already shown increasing evidence indicating the association of personal health with activity engagement for various age groups [94, 111]. Everyday activity patterns are investigated in different areas such as occupational therapy, diet monitoring, etc., to improve subjects' physical and mental health by understanding how they use their time with various activity occupations. A lot of investigations and surveys have shown that most of time is spent on some of the activities such as sleeping and resting (34%), domestic activities (13%), TV/radio/music/computers (11%), eating and drinking (9%), which almost count for nearly 70% of the time in a typical day.

In [83], the most frequently-occurring everyday activities are explored to rate the enjoyment when people experience these activities. The 16 activities are listed in Table 3.1, ordered decreasingly by enjoyment rating. The impact of everyday activities on humans' feelings of enjoyment will also affect human health, which makes these activities important in well-being analysis and lifelogging.

Similar patterns of activity are also shown in [2], [3] and [40] with sleeping being

Table 3.1: Everyday activities from [83] in decreasing order of enjoyment

1	2	3	4
Intimate relations	Socializing	Relaxing	Pray/worship/meditate
5	6	7	8
Eating	Exercising	Watching TV	Shopping
9	10	11	12
Preparing food	On the phone	Napping	Taking care of Children
13	14	15	16
Computer/Internet	Housework	Working	Commuting

the most dominant activity followed by other activities like housework, watching TV, employment/study, etc. [2] and [3] also show that time distribution on activities varies with age groups. However, some activities achieve high participation agreement among all people investigated in the survey. High agreements across all age groups are obtained on activities such as sleeping, eating and drinking, personal care, travel, etc.

In our interpretation of lifelogging events, we select our target activities from the candidates with the following criteria:

- Time dominance: As described above, a small number of activities occupy a large amount of our time. The analysis of these activities can maximize the analysis of the relationship between time spent and human health. The selected activities should cover most of the time spent in a day.

- Generality: Even though the time spent on activities varies from age group to age group, there are some activities that are engaged in by different age groups. The selection of activities with high group agreement will increase the generality of activity analysis in lifelogging. Therefore, the output can be suitable for a wider range of age groups.

- High frequency: This criteria helps to select the activities which have enough sample data in lifelogging records. High sample frequency can improve the detection and other processing qualities, such as classification and interpretation. The activities with high time dominance are not necessarily have high frequency. For example, ‘sleeping’ covers a large part of time in a day but its frequency is low.

With these criteria in mind, we combined the activities investigated in literatures like [83], [2], [3], etc. and selected the following activities as targets for our further analysis. They are listed in Table 3.2. Note that these activities listed in Table 3.2 are still far from covering all activities in daily life analysis but we believe that they are representative and can be applied for further activity of daily living analysis. These activities will be used in testing the ideas and algorithms later in this thesis. Besides, the selection of activities and our algorithms to be proposed are generic. When more activities are chosen for various purposes of analysis, our algorithms can be applied in similar manner without loosing capabilities of generality.

Table 3.2: Target activities for our lifelogging work

1	2	3	4
Eating	Drinking	Cooking	Clean/Tidy/Wash
5	6	7	8
Washing clothes	Using computer	Watching TV	Children care
9	10	11	12
Food shopping	General Shopping	Bar/Pub	Using phone
13	14	15	16
Reading	Cycling	Pet care	Going to cinema
17	18	19	20
Driving	Taking bus	Walking	Meeting
21	22	23	
Presentation (give)	Presentation (listen)	Talking	

3.2.2 Topic Related Concepts

As accepted in the multimedia retrieval community, the term ‘topic’ is used to represent a given query task which has higher level semantic meaning. Similarly in our work, we use the term 'topic' to refer to a specific event type, i.e. an 'activity' in lifelogging. Without specific discrimination, an 'event' can refer to a specific case of an everyday 'activity' and vice-versa.

How to decide the possible concepts related to the event topics above is still an issue in our work. In state-of-the-art everyday concept detection and validation [34], concepts are suggested by several SenseCam users after they have gone through and studied several days' lifelogged events of their own. Then, being more familiar with their own lifestyles through reviewing their lifelogs, the concepts are discussed and filtered according to the criterion that the concept can be detected with satisfying accuracy. During this procedure, the concepts are not selected in a way the related event topics are considered. Some concepts are selected but they might not be helpful in interpreting specific event semantics. In addition, some concepts which might be of great help in recognizing and interpreting a specific event type may be ignored in the selection procedure. This limits the performance of event detection and semantic interpretation especially when particular concepts relevant to the event are missed. Given the fact that concept detection is not perfect, it is still a problem when a non-relevant concept is selected to be used in a query. The non-relevant concept here will reduce the performance by incurring high noise in the query step.

To find a set of candidate concepts related to each of the activities described in Section 3.2.1, we carried out user experiments on concept selection where candidate concepts related to each of the activities above were pooled based on user investigation. Although individuals may have different contexts and personal characteristics, the common understanding of concepts that is already socially constructed and allows people to communicate according to [92] and [78], also makes it possible for users to choose suitable concepts relevant to activities. User experiments were carried out to find out candidate concepts which potentially have high correlation with activity semantics. Details of the experimental methodology will be described in Chapter 4.

The user experiments give us a set of candidate concepts with regard to the activities we explored in Section 3.2.1. These concepts are used to construct an event-based semantic space for every activity engagement being logged. The concept space is expanded by each concept as one dimension, as shown in Figure 3.2 and events are represented by groups of images which have their own concept vectors. One group of images has the same topic describing the event. The semantic interpretation makes full use of the concept vectors of images constructing the event to infer higher-level semantics. Compared to current algorithms of concept selection, we propose a semantic density-based concept selection algorithm to find the most useful concepts in the following sections. While existing algorithms are not a good match for the particular problems of detecting the most appropriate semantic concepts for lifelog events and are not tested in lifelogging lexicons, our algorithm has the advantage of selecting concepts from a global point of view and is tested to be effective for everyday concept selection and ranking. A preliminary experiment is described to illustrate the algorithm.

3.2.3 Constructing the Event Semantic Space (ESS)

To select concepts to represent the semantics for events, we need to define the concept space. Intuitively, every concept representing any event should be one dimension, and the projection of an event onto the concept space is the co-occurrence information in between. However, different concepts have different impacts on event interpretation. Concepts which are neither too general nor too specific should be selected in the semantic space to reduce dimensionality and noise for concept detection. In a nutshell, we should include topic-related concepts with decent frequency, and exclude general and over-specific senses.

The Event Semantic Space (ESS) is defined as a linear space with a set of concepts as the basis as depicted in Figure 3.2. In order to ensure high coverage of the space, we

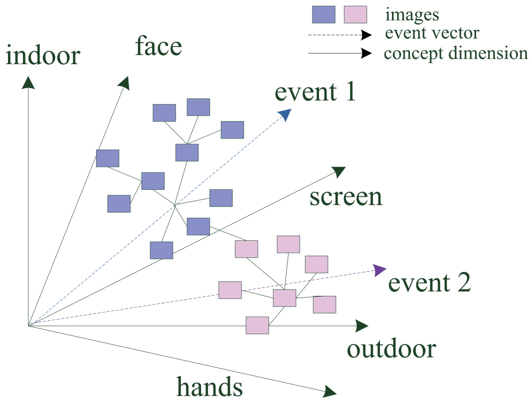


Figure 3.2: Concept space and event concept vector.

elaborate the selection of a minimum concept basis set according to the generalization of entities in the semantic space. Ideally, any semantic query can be represented as a coordinate in the semantic space. According to Wei and Ngo, [169], “The basis concepts provide a high coverage of semantic space, and are probably the ones that should be developed if they are feasible to be built with the current technology.”

We denote the semantic space as $\mathcal{S}$ spanned by a set of concept bases $\{\mathbf{c_1}, \mathbf{c_2}...\mathbf{c_N}\}$, where $\mathbf{c_i} \in \mathcal{S}$ is a basis concept. Then, the semantic space is constructed as:

$$\mathbf{c_1} \times \mathbf{c_2} \times ... \times \mathbf{c_N} \rightarrow \mathcal{S}$$

Let us assume that a concept detector d_i can be learned from low-level features for concept $\mathbf{c_i}$. We will have a concept detector set $D = \{d_1, d_2...d_N\}$ available to transform from the low-level feature space $\mathcal{L}$ to the semantic space $\mathcal{S}$. Then the relation between two spaces can be represented by:

$$D(\cdot) \otimes \mathcal{L} \rightarrow \mathcal{S}$$

where $D(\cdot) = \{d_1(\cdot), d_2(\cdot)...d_N(\cdot)\}$ is the corresponding transformation of concept detector set D.

3.3 Investigating ESS Concept Relationships

3.3.1 Definitions

An ontology is used to represent the concepts and concept relations within a domain. Usually ontologies are considered as graphs, where nodes represent concepts and edges represent relations between concepts. In much of the research dealing with discrete objects and binary relations, a graphical representation of the objects and the binary relations between them is a very convenient form of representation which can use well-established graph theory for algorithms to manipulate them [135]. As part of domain knowledge, an ontology structure contains the semantics of concepts, such as a child/descendant concept being a sub-concept of its parents/ancestors, which is reflected in an hierarchical ontology. The structure also decides the heritage of concept properties. For example, a *car* will inherit the features of its superordinate, probably a *vehicle*. Ontology-based similarity or relatedness measures can exploit the ontology structure or additional information to quantify the likeness or correlation between two concepts. To show the difference between similarity and relatedness, let's see an example of three concepts, *teacher*, *professor* and *school*. In this example, *teacher* and *professor* are similar concepts whereas *professor* and *school* are related to each other. In different application domains, similarity and relatedness might be treated separately.

3.3.2 Lexical Similarity Based on Taxonomy

Concepts are clustered according to their distribution in the semantic space. With a lack of features or coordinates in this semantic space, concepts can only be clustered in terms of their ontology relationships between each other. As a popular English lexical ontology, WordNet [115] is widely used as a semantic knowledge base. Synsets are basic elements in WordNet representing the sense of words. The current version

(3.0) of WordNet contains 155,327 words grouped into 117,597 synsets. The *is-a* relationship is modeled as hypernymy in WordNet where one concept is more general than another. Hyponymy represents the characteristic that one concept is more specific than another. The meronymy/holonymy connection is the semantics representing a *part-of* relationship. This comprehensive coverage and explicit representation of concept relationships make WordNet useful in analyzing the concepts relationship within the semantic space.

Path-based Methods Semantic similarity has been explored in previous research to define a matric for concept relationship analysis. Rada [135] was first to develop the basis for edge-based measures for concept similarity by defining the distance in a semantic network as the length of the shortest path between the two concept nodes. Richardson and Smeaton [139] built on the work of Resnik, reported in the survey article in [138] to further refine the similarity measures. The Hirst and St-Onge [73] similarity measure, takes path direction into account and the idea is that the concepts are semantically close if their WordNet synsets are connected by a short path which does not change direction too often. Another similarity definition is proposed in [174] by Wu and Palmer for verb similarity calculation since most of the other work is built upon noun concepts, and applied in machine translation. The formula extended by Leacock and Chodorow [95] is also a path-based similarity algorithm which determines similarity with regard to the maximum depth of the taxonomy.

Information-based Methods Semantic similarity based on information content is also an important branch in lexical relationship analysis. This kind of approach relies on the hypothesis that the more information two concepts share, the more similarity they have. The informativeness of a concept is quantified by the notion of its Information Content (IC), which is calculated based on the occurrence probability of concepts in given corpus. IC is obtained by negative likelihood of encountering a concept in a given corpus [137]. The basic intuition of using negative likelihood assumes that the more likely a concept appears in a corpus the less information it conveys.

Based on the IC formula, the concept will contain less information if the probability of its occurrence in a corpus is high. The advantage if using information content is that, once given a properly constructed corpus, the information content can be adapted in different domains because the information content is included in a statistical way according to occurrences of the concept, its subconcepts and sub-sumers.

In [138], Resnik applied information content to semantic similarity calculation by the information of Most Specific Common Abstract ($msca(c_1, c_2)$) as the amount of information that concepts c_1 and c_2 have in common. In this approach, only the 'is-a' relationship is applied because only the information of the sub-suming concept of the two concepts being compared, is used. In [153], this similarity measurement is also employed by Quigley and Smeaton to compute word-word similarity in image caption retrieval. Jiang and Conrath in [79] and Lin in [100] also both extended Resnik's measure by taking even more factors into account. Table 3.3 summarizes these semantic similarity relationships.

Hybrid Methods Some hybrid methods also attracted a spate of research interest recently, which try to make use of the WordNet hierarchy and IC measure to calculate semantic similarity. In [147], authors proposed similarity measures in taxonomy that use Information Content. However, the IC value is concluded from WordNet taxonomy hierarchy rather than deriving statistics from given corpus. Experiments tested on human judgements showed that it performs well compared to prevailing semantic measures. The measure is easier to calculate with the application of an ontological hierarchy in IC obtaining. [129] extends the intrinsic information content and take into account the whole set of semantic relations defined in ontology to conclude a new framework of relatedness calculation. The framework, which is called FaITH (Feature and Information THeoretic), maps the feature-based model of similarity into the information theoretic domain and also considers ontology link structure in its relatedness calculation [129].

Table 3.3: List of concept similarity matrices

Similarity measures	Function definition	Path-based	Infor.-based
Rada	$sim(c_1, c_2) = 1/len(c_1, c_2)$	√	×
Hist & St-Onge	$rel(c_1, c_2) = C - len(c_1, c_2) - k \times d$	√	×
Wu & Palmer	$sim(c_1, c_2) = \frac{2 \cdot depth(LCS)}{len(c_1,c_2)+2 \cdot depth(LCS)}$	√	×
Leacock & Chodorow	$sim(c_1, c_2) = -log\frac{len(c_1,c_2)}{2D}$	√	×
Resnik	$sim(c_1, c_2) = -logp(LCS)$	×	√
Jiang & Conrath	$sim(c_1, c_2) = \frac{1}{2 \cdot logp(LCS) - (logp(c_1)+logp(c_2))}$	×	√
Lin	$sim(c_1, c_2) = \frac{2 \cdot logp(LCS)}{logp(c_1)+logp(c_2)}$	×	√

3.3.3 Contextual Ontological Similarity and Relatedness

WordNet is a small ontology of primarily taxonomic semantic relations. ConceptNet extended WordNet to include a richer set of relations appropriate to concept-level nodes [102]. In the version of ConceptNet we use later, the relational ontology consists of 20 relation types falling into categories like K-lines, Things, Agents, Event, Spatial, Causal, Functional and Affective [103].

In ConceptNet, all concepts are linked with the above-mentioned relations which can reflect the correlations between concepts. We apply a link-based relatedness measure to maximize the concept relations in measuring concept correlation. This differs from WordNet which uses mainly taxonomic relationships, while ConceptNet employs more context relationships. While WordNet similarities only consider subsumption relations to assess how two objects are alike lexically, relatedness takes into account a broader range of relations which can be measured using ConceptNet.

According to [135], superordinate (*is-a*) links are assigned high importance tags in Quillian's model of semantic memory in which concepts are represented by nodes and relationships by links. When an ontology contains *is-a* links only, short paths will significantly contribute to positive evidence of similarity by applying spreading activation. Meanwhile, the correspondence between semantic distance (shortest path length) and semantic relatedness (conceptual distance) will also be strong.

The relations between concepts reflect the semantic correlation between two concepts. We assume that semantic relations are transitive so the more related two concepts are, the shorter paths they will have. The relatedness between two concepts varies inversely with the length of the shortest path between the two concepts. Conceptual relatedness is a monotonically decreasing function of path distance. Our approach takes into account the length of paths between two concepts. In ConceptNet, because the edges between concepts are directional, we combine the length of the path between concept c_1 and c_2 as well as path between c_2 and c_1. The similarity between two concepts are defined as:

$$S_{CN}(c_1, c_2) = max(ActivationScore(c_1, c_2), ActivationScore(c_2, c_1)) \tag{3.1}$$

where $ActivationScore(c_1, c_2)$ represents the activation score of c_2 starting from c_1, and vice versa. Here, we use activation score to represent the correlation of two concepts. *ActivationScore* is performed by spreading activation in ConceptNet to find the most similar concepts with regard to a starting concept. The starting concept is initialized with activation score 1.0 and then the nodes connected with the starting concept with one link path, two links path, etc. are activated. The activation score of connected node b with original node a is defined as:

$$ActivationScore(a,b) = \sum_{c \in Neighbor(b)} ActivationScore(a,c) \times d \times w(c,b) \tag{3.2}$$

where d is a distance discount ($d < 1$) to give the concepts far from the original concept a lower weight and $w(c,b)$ is the relation weight of the link from c to b. In this thesis, we apply the same relation weight for *ActivationScore*. For any given concept b, the activation score related to a is the sum of scores of all nodes connected to it.

3.4 Concept Selection Based on Semantic Density

Our measure of semantic density relies directly on the semantic distance between concepts. If the distance measured between concepts is small, then the concepts have high density. The semantic distance is used as a measure by which the concepts are clustered to represent event semantics, as shown in Figure 3.3. In Figure 3.3, each triangle stands for one particular event. The concepts representing the events are illustrated by dots inside each event. In our semantic topic-related concept selection, we deal with the research question by means of identifying the similarity between concepts as a linguistic problem. The processing consists of text pre-processing, word similarity and phrase similarity calculations.

3.4.1 Text Pre-Processing

Concepts are represented in the form of textual descriptions and these descriptions are usually not normalized. In order to obtain more appropriate concept similarity, before concept similarity can be calculated all concept descriptions need to be normalized.

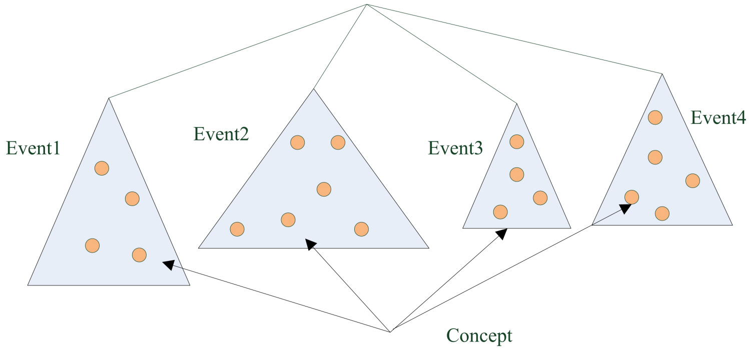


Figure 3.3: Event semantic density.

- **Tokenization**: Tokenization is applied to break the queries or descriptions into each separated word.

- **POS Tag**: Not all words in descriptions are useful in comparing their semantics. A parts-of-speech (POS) tag is a process to mark up the words in a text as corresponding to a particular part of speech, based on both its definition, as well as its context. The context includes the relationship with adjacent words at either a phrase or sentence level.

- **Stopword Removal**: stopwords are removed as part of the normalization of texts. Commonly occurring words are inspected and removed using the SMART stoplist [145].

- **Lemmatization**: In order to return proper result from the lexicon dictionary, the words need to be in their original form. After stopwords and punctuation are removed, lemmatization is the process of converting different inflected forms of a word to their original form so they can be analyzed as a single item. All lemmatized words are also converted to lowercase.

3.4.2 Conjunctive Concept Similarity

As we described earlier, an arbitrary document or query is represented as a vector of term weights for similarity comparison in an information retrieval system. The term vector can be regarded as a new distinct compound concept. The concept reflected by a document is best described by ANDing the concepts represented by its index terms [135], which facilitates the documents being treated as conjunctive concepts.

When concepts have several disjunctive meanings in WordNet synsets, we apply 'disjunctive minimum' [135] to obtain the similarity between the two concepts. That is, when a concept has alternative synsets because it is polysemous, we calculate the minimum conceptual distance between the synsets and the other concept as the final distance between the two concepts. Assume that we have two concepts c_1 and c_2 and c_1 has three disjunctive synsets syn_1, syn_2, syn_3. In terms of 'disjunctive minimum', the conceptual distance between c_1 and c_2 will be given by: $d(c_1, c_2) = min\,[d(syn_1, c_2), d(syn_2, c_2), d(syn_3, c_2)]$.

In calculating conjunctive concept similarity, we take into account all elementary concepts in the conjunctive concept. [135] specified that:

> "When conjunctive concepts are compared, we must take into account the conceptual distances among elementary concepts."

We regard the comparison of the similarity of two conjunctive concepts as finding the best assignment for a bipartite graph. In both sides of the bipartite graph, the nodes represent elementary concepts. As with solving the best matching problem, we apply the Hungarian algorithm to decide the maximum similarity matching between the two conjunctive concepts. Note that the Hungarian algorithm is prohibitively computationally expensive especially for long sentences or documents. An alternative approach is to perform conjunctive concept similarity which is more computationally efficient and can be defined as [153]:

$$sim(c_1, c_2) = \frac{1}{M \cdot N} \sum_{i=1}^{M} \sum_{j=1}^{N} sim(e_i, f_j) \tag{3.3}$$

where c_1 and c_2 are the compound concepts being compared and e_i and f_j are elementary concepts for c_1 and c_2 respectively. In this formula, the sum of pairwise elementary concept similarities is normalized by the product of the length of conjunctive concepts to reduce the bias of the number of elementary concepts [135]. Thus the more elementary concepts a compound concept has, the less (relatively) a path through an elementary concept will account for similarity. Some other approaches to conjunctive concept similarity calculation can also be found in [153].

3.4.3 Density-based Concept Selection

In the concept set, each concept stands for a semantic entity in the semantic space. The pairwise relationship can be determined by their semantic similarity, which is represented as an $n \times n$ matrix M. The similarity matrix M is a symmetric matrix, each row or column of which stands for the similarity values of corresponding concepts with regard to all concepts. The most similar concept group can represent a subspace in the semantic space within which the concepts have high correlations.

Principle Component Analysis (PCA) is a useful tool in pattern recognition in high-dimensional spaces to reduce the number of dimensions without losing much of the information/variance represented by the data. With PCA, a *feature vector* can be selected with higher eigenvalues of the covariance matrix, retaining most of the information. The reduction of dimensions can help to compress data as well as reducing noise induced by too many dimensions. Although PCA can ensure the orthogonality of the bases, the representation of original data in terms of *feature vectors* is difficult to interpret and embed with it semantics, which is also agreed by Wei and Ngo in [169]. However, subsets of concepts which are clusters in semantic space represent specific domain semantics representatively. They should be as disjoint as possible to be selected as the bases in semantic space. Therefore, the number of clusters, that is also the number of bases selected by clustering, should be consistent with the number of *feature vectors* selected by PCA.

We apply PCA in helping to find the most appropriate number of clusters in density-based concept selection. The total number of clusters is decided by considering the inconsistency coefficient and PCA. The inconsistency coefficient value was used to decide the appropriate number of clusters in the dendrogram. The inconsistency coefficient value is defined to compare the height of a link in a cluster hierarchy with the average height of links below it. This value can be used to identify the groups of concepts which are densely packed in certain areas of the cluster dendrogram. The lower, the more similar the concepts are under the link.

To demonstrate how the approach works, we take ConceptNet contextual similarity as an example, which is described in Section 3.3.3. Figures 3.4, 3.5 and 3.6 are all demonstrated using the typical concept set (85 concepts) we will investigate in Section 4.3 as shown by Table 4.4, in italics. In Figure 3.4 (left), the number of clusters formed when inconsistent values are less than a specified inconsistency coefficient is shown. According to PCA, the cumulative energy content for the top k Eigenvectors is shown in Figure 3.4 (right). As described above, the number of orthogonal vectors represent disjoint semantics in semantic space. We hope to group as many similar concepts as possible which leads to less orthogonal bases in the space. Finally, the trade-off between PCA inconsistency coefficient is used to find a proper number of clusters for agglomerative algorithm. As shown in Figure 3.5, the intersection of PCA (blue) and inconsistency coefficient (green) curves is selected to decide the number of clusters. The cluster number at the trade-off point can still keep the cumulative energy higher than 90% while the inconsistent coefficient is at a relatively low level.

The dendrogram generated by hierarchical clustering is illustrated in Figure 3.6.

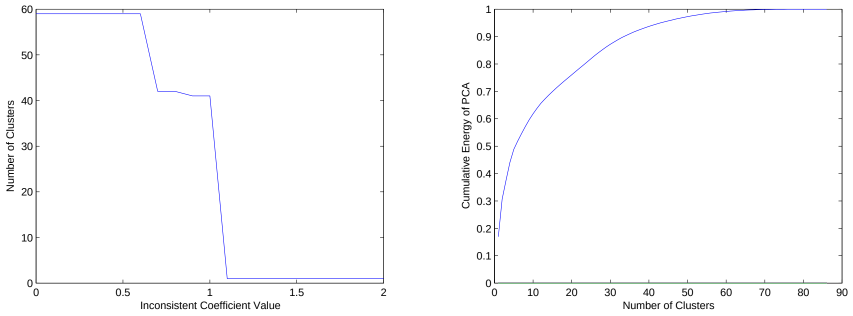


Figure 3.4: Inconsistency coefficient and PCA.

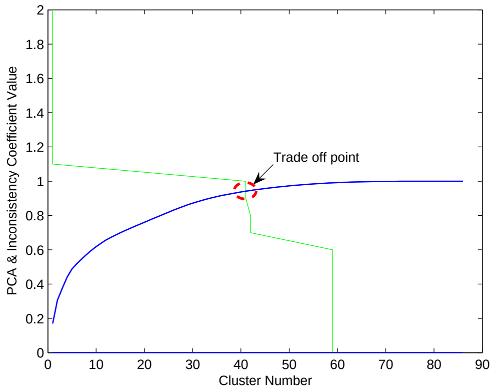


Figure 3.5: Number of clusters.

In the dendrogram, semantically related concepts are linked together within a cluster. For example, ‘food’, ‘table’, ‘people’, ‘drink’ and ‘plate’ are grouped together, from which it is not hard for us to understand that these concepts are more related to the activity ‘eating’ (shown as a dashed circle in Figure 3.6). More examples can be shown in Figure 3.6, such as ‘milk’, ‘water’, ‘cup’ are clustered for ‘drinking’ while ‘sky’, ‘path’, ‘tree’, ‘road sign’ and ‘road’ are clustered for ‘walking’. The semantic clustering facilitates the selection of topic-related concepts. As a concept, a given topic is also an instance which can be clustered in the concept space. Therefore, the concepts within the same cluster of a given topic can be the concept candidates.

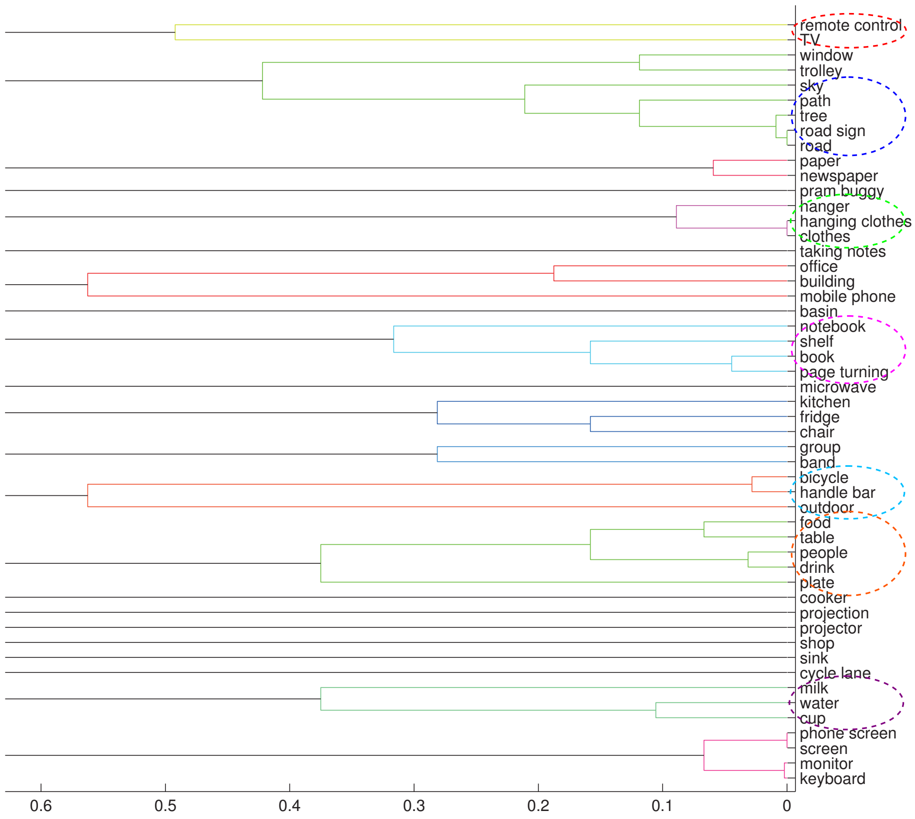


Figure 3.6: Relationship dendrogram for concepts.

3.5 Leveraging Similarity for Concept Ranking

In the previous section we described a method for selecting candidate concepts in similarity matching based on clustering concepts in a semantic space. Although the selected concepts have high correlation with the given activity topic, there may still be other concepts missing which might be related to the topic. This is because the clustering algorithm only considers the local distance in the semantic space. Since the selected concepts have high correlation semantically with the given topic, they can be used as seeds in finding other related concepts. To leverage the concept similarity in a global view, we employed the random walk model which has been shown to be efficient in many applications.

3.5.1 Concept Similarity Model

Random walk is a widely used algorithm which uses links in a network to calculate global importance scores of objects which are connected in the network. Random walk allows us to compute the probability of a random walker being located in each vertex through time series. This is performed as a discrete Markov chain characterized by a transition probability matrix. Its application as PageRank [126] has shown great success in web searching. The intuition of PageRank views web pages as a connected graph by forward and backward hyperlinks. In PageRank, a web page is important if there are also important pages which link to it.

We model concept similarity as a graph $G = (C, E)$, where V is the concept set and E is a set of edges that link concepts. Each edge is assigned with a given similarity value describing the probability that a random walker jumps among the concepts. As is shown in Figure 3.7, the concept sets and given topics can both be viewed as vertices in the graph, connected by similarity links. In last section, the concepts that were similar to the given topic are selected as candidates, shown as the shaded concepts in Figure 3.7. However, the concepts which are similar to candidate concepts but have no direct similarity link with the given topic, are ignored. The random walk model is employed to rank the concepts with candidate concepts as seeds from a global similarity view.

3.5.2 Similarity Rank

In this model, we can consider the process as a Markov chain where the states are concepts and transitions are similarity links between them. A random walker will start with a prior probability and surf on the graph by following the similarity links. The similarity random walk is based on mutual reinforcement of concepts, that is,

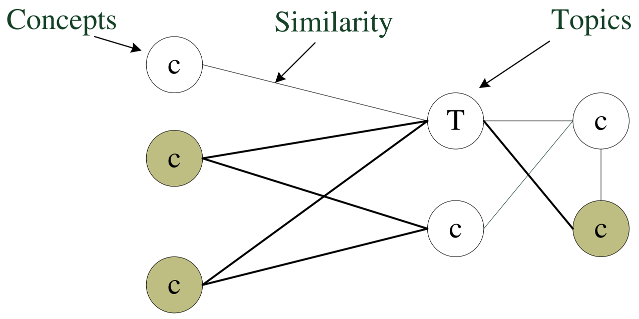


Figure 3.7: Concept similarity link.

the score for concept relative to a given topic influences, and is being influenced by, the score of other concepts. We formulate the calculation of the score for c_i as:

$$x(c_i) = \sum_{j=1}^{n} Sim_{ij} x(c_j) \tag{3.4}$$

where Sim_{ij} is a normalized similarity value between c_i and c_j. Following the PageRank algorithm, we update the score of concepts by:

$$\begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} = \alpha \begin{pmatrix} Sim_{11} & \dots & Sim_{1n} \\ \vdots & \ddots & \vdots \\ Sim_{n1} & \dots & Sim_{nn} \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} + (1-\alpha) \begin{pmatrix} d_1 \\ \vdots \\ d_n \end{pmatrix} \tag{3.5}$$

where $(d_1 \dots d_n)^T$ is prior score vector, and α is decay factor. The equation can be formalized in a compact matrix form as:

$$\mathbf{x} = \alpha \mathbf{T} \mathbf{x} + (1-\alpha)\mathbf{d} \tag{3.6}$$

In this formula, $\mathbf{x}$ stands for the score vector and T is the similarity matrix with the sum of each column normalized to 1. For each concept c_i, there is $x_i = \sum_{j=1}^{n} \alpha \cdot Sim_{ij} x_j + (1-\alpha) \cdot d_i$ for the score. To solve Equation 3.6, we can convert it to:

$$\mathbf{x} = \alpha(\mathbf{T} + (1-\alpha)/\alpha \cdot \mathbf{d} \cdot \mathbf{1})\mathbf{x} \tag{3.7}$$

If we assume $\mathbf{A} = \alpha(\mathbf{T} + (1-\alpha)/\alpha \cdot \mathbf{d} \cdot \mathbf{1})$, then $\mathbf{x}$ will be the Eigenvector of $\mathbf{A}$. Although this leads to a direct solution for the formula, the iterative calculation converges fast enough and is usually employed. In our experiment to be described in Chapter 4, the iteration starts with $\mathbf{x}$ initialized as $\mathbf{0}$.

3.6 Summary

In this chapter, the selection of everyday activities and concepts for analysis of lifelogging data are investigated. To facilitate the indexing and retrieval of lifelog media, a semantic density-based concept selection algorithm is proposed which can utilize the semantic similarity obtained from ontologies. The prevalent ontological similarity based on WordNet and ConceptNet are investigated and used in this thesis to obtain pairwise concept similarity. The algorithm applies agglomerative clustering to select densely relevant concepts as candidates based on a similarity score. The final concept list is then ranked in a PageRank-like algorithm, which we call *similarity rank*. Both the clustering algorithm in concept space and similarity rank try to handle the concept similarity globally by applying pairwise concept relationship reasoned from ontologies. While the clustering in concept space can return the most relevant concepts, similarity rank returns the final list of concepts ordered according to their relevance to the given activity. For example, by employing *ConceptNet* similarity, concept ‘cycle lane’ is first selected as a potential concept for activity ‘driving’. By applying similarity ranking, a list of concepts are ranked and returned as ‘cycle lane, window, people, car, inside car, glass, traffic light, road, tree, road sign, sky...’. More relevant concepts like ‘car’, ‘inside car’, ‘traffic light’, ‘road’, ‘tree’, etc. are ranked on the top of the final list.

We now progress to assessing the density-based concept selection algorithm and the impact of ontological similarity measures in a set of experiments reported in the next chapter.

Chapter 4

Evaluating Concept-based Similarity and Selection

In Chapter 3 we introduced a mechanism for computing similarity between objects such as a concept and a lifelog event type, based on the structure of semantic concepts within ontologies and the relative frequencies of occurrence of those same concepts elsewhere in a collection. Also considered in this similarity measure was the "distance" between concept pairs in terms of the navigation distance between them in a pre-constructed ontology. In this chapter we report a set of experimental results which assess the effectiveness of the proposed methods. Note that we still inherit the terminology we used in Chapter 3 and refer to 'topic' as a type of everyday 'event' or 'activity'.

4.1 User Experiment

Our experiments first started with a user investigation to find out the set of possible concepts involved in interpreting lifelog events. The respondents in our experiments are chosen from among the researchers or students in our own research group, most of whom are working in computer science and some of them are also logging their everyday life with SenseCam so the group are sympathetic to and familiar with the idea of indexing visual content by semantic concepts.

In total, 13 respondents took part in our user experiment, for whom the demographic information and experience with SenseCam are shown in Table 4.1.

Table 4.1: Demographic information of participants

User ID	Gender	Age Group	Ever Used SC	Working on SC
1	Male	26-30	Yes	Yes
2	Female	21-25	No	No
3	Male	36-40	Yes	Yes
4	Male	26-30	Yes	Yes
5	Male	26-30	No	No
6	Male	26-30	No	No
7	Female	26-30	Yes	Yes
8	Male	26-30	Yes	No
9	Male	31-35	Yes	Yes
10	Male	21-25	Yes	No
11	Female	21-25	Yes	No
12	Female	26-30	No	No
13	Male	31-35	No	No

Among the 13 participants, there are 9 males and 4 females, whose ages are all in the range of 20 – 40 years old. About half of the participants (7 in 13) are in the age group of 26 – 30 while 3 are in 21 – 25 and another 3 are over 30. There are 8 participants who are familiar with SenseCam and have worn it for various periods. 5 participants are currently doing research using SenseCam and are engaged in different tasks like visualization, concept detection, medical therapy, etc.

In our user experiment, participants were shown SenseCam images for samples of activities and were then surveyed by questionnaires based on their common understanding of SenseCam activity images as well as the relevant concepts occurring in those SenseCam images. The experiment was organized into three phases: a study phase, pooling phase and rating phase. In the study phase, target activities were first described to the respondents to get them familiar with the activity concepts. Exemplar image streams for each activity listed in Table 3.2 were shown to the group and we asked them to inspect the SenseCam images. In the pooling stage, participants were asked to go through images collected individually to list the possible concepts they thought might be helpful in order to retrieve the activities. The aim of the second phase is to determine a large concept set that might be helpful in analyzing SenseCam images in order to detect activities. In the final rating phase, the number of subjects who thought a concept was relevant to the given activity is calculated, for all target activities. Then the higher number of "votes" the concept gets and the greatest agreement among all subjects, the more relevance we give to the concept for that activity, in the experiment.

Table 4.2: Experimental data set for pooling

Topics	Eating	Drinking	Cooking	Clean/Tidy/Wash	Wash clothes
Events	5	5	5	5	2
Images	260	66	398	125	127
Topics	Watch TV	Child care	Food shopping	General Shopping	Bar/Pub
Events	5	5	5	5	5
Images	70	146	161	269	758
Topics	Reading	Cycling	Pet care	Going to cinema	Driving
Events	5	2	1	1	5
Images	148	92	2	728	227
Topics	Walking	Meeting	Presentation (give)	Presentation (listen)	Use computer
Events	5	2	2	1	5
Images	93	81	164	256	226
Topics	Use phone	Taking bus	Talking		
Events	5	5	17		
Images	241	226	704		

To make the user experiment more efficient and to optimally mine the users' social experience and knowledge, the subjects were asked to list as many concepts associated with each event topic of interest, in the pooling stage. The application for inspecting the SenseCam images was built with a controlled browsing speed to help the subject look through SenseCam images at a comfortable rate. Details of the data used in the pooling stage are shown in Table 4.2. Taking 'eating' event for example, there are totally 5 event samples for 'eating' as shown in Table 4.2 and these 5 sample streams consist of 260 SenseCam images in all. Note that the activities listed in Table 4.2 are all from the everyday activities we investigated in Section 3.2.1, as shown in Table 3.1. To provide cues for participants to find relevant concepts, the images depicting different activities were shown. In our later experiment on evaluating concept selection, Section 4.4, we will use the concept set obtained from the user experiment. The concepts investigated include 171 concepts in total whose details will be given in Section 4.3. The large set of concepts and concept diversity also reflect the functionality of SenseCam images in associating with concepts. Some typical concepts related and their corresponding activities are shown in Table 4.3:

Table 4.3: Examples of everyday concepts

Activities	**Concepts**
Eating	food, plate, cup, table, cutlery
Drinking	cup, glass, table
Cooking	hands, sink, fridge, microwave
Use computer	keyboard, table, hands
Watch TV	TV, remote control
Care for Children	pram/buggy, child, toy
...	...

4.2 Experimental Evaluation Methodology

Two baselines were employed to evaluate our concept selection algorithm, namely the user experiment as the "oracle" result and the mutual information based concept selection (to be described in this section) output. In the user experiment, the ranked concepts are analyzed to select the best set of agreed ones which are decided unanimously for the evaluation. The annotated dataset will be analyzed to select relevant concepts by calculating the mutual information value. Generally speaking, the semantic density-based concept selection and mutual information (MI) based approaches are both automatic approaches compared to the manual user experiment.

To give a brief description of MI and how it is used in concept selection, we follow the formalizations by [101] in video retrieval as: $I(R;C) = \sum_{r,c} P(r,c) log \frac{P(r,c)}{P(r)P(c)}$, where the R and C are both binary random variables. R stands for the relevance of a video shot for which the instance $r \in \{relevance, irrelevance\}$ while C represents the presence or absence of a concept in a video shot for which the instance $c \in \{presence, absence\}$. MI reflects the contributing of knowledge about C in reducing the entropy of R using maximum likelihood estimates, so concepts can be ranked according to MI. After removing the concepts with the suggested threshold of 1%, the negatively helpful concepts are also filtered out by the criterion that $I_p(absence, relevance)$ of a concept is greater than $I_p(presence, relevance)$, where $I_p(r,c)$ is pairwise mutual information, defined by $I_p(r,c) = log \frac{P(r,c)}{P(r)P(c)}$.

Note that no algorithm is perfect in concept selection. Each algorithm has its pros and cons which depends on the application. Even the “oracle” user experiment also has the problem of finding broader concepts compared to the MI-based approach. The MI based algorithm, however, tends to select some non-relevant concepts but which co-occur with the event topic often. Event segmentation which can also have errors, and erroneous annotation can also introduce artifacts leading to poor performance for the MI-based approach. The MI-based approach also suffers from the lack of representative events in the annotation sets. The benchmarks are introduced here to evaluate the algorithms from different performance points of views. These viewpoints are group consistency, set agreement and rank correlation [78].

Group consistency:

In order to assess the clustering result of our algorithm, we define *group consis-*

tency to measure the degree of semantically related concepts to be clustered. When two related concepts are grouped in the same cluster by our algorithm, this should give a positive contribution to the overall consistency value, otherwise, a negative contribution should be given to overall consistency. To determine whether two concepts should be grouped together is a subjective decision hence the results of human experiments are used as an oracle evaluation. We formalize the notion of human judgement on concept group consistency as a binary function O:

$$O(c_i, c_j) = \begin{cases} 1 & \text{if } c_i \text{ and } c_j \text{ are under the same topic} \\ 0 & \text{if } c_i \text{ and } c_j \text{ are not under the same topic} \end{cases} \tag{4.1}$$

Similarly, we define another binary function G to reflect the grouping result of two concepts by clustering as:

$$G(c_i, c_j) = \begin{cases} 1 & \text{if } c_i \text{ and } c_j \text{ are in the same cluster} \\ 0 & \text{if } c_i \text{ and } c_j \text{ are not in the same cluster} \end{cases} \tag{4.2}$$

Note that these two binary functions are both symmetric which means $O(c_i, c_j) = O(c_j, c_i)$ and $G(c_i, c_j) = G(c_j, c_i)$. Generating a set $\mathcal{C}$ of ordered pairs $\mathcal{C} = \{(c_i, c_j), 1 \leq i, j \leq |C|, i \neq j\}$ from concept set C, the overall group consistency for C is defined based on these two functions and is formalized as:

$$GC = \frac{|\mathcal{C}| - \sum_{(c_i,c_j)\in\mathcal{C}} IC(O, G, c_i, c_j)}{|\mathcal{C}|} \tag{4.3}$$

where

$$IC(O,G,c_i,c_j) = \begin{cases} 1 & \text{if } O(c_i,c_j) \neq G(c_i,c_j) \\ 0 & \text{if } O(c_i,c_j) = G(c_i,c_j) \end{cases} \tag{4.4}$$

Group consistency reflects the performance of similarity-based clustering in a form of pairwise grouping result. The ratio is computed as the fraction of the pairs for which the semantic clustering algorithm gives the same output as the user experiment. If there are no cases in which semantic clustering mis-groups a concept pair, GC is equal to 1. Conversely, GC is equal to 0 when no concept pairs are correctly grouped.

Set agreement: Set agreement is used to compare two concept sets without considering the ranking measurement. It defines the positive proportion of specific agreement between two sets [78]. The score of set agreement is equal to 1 when the two sets $C_1 = C_2$, and 0 when $C_1 \bigcap C_2 = \phi$.

Rank correlation: Rank correlation is used here to study the relationships between different rankings on the same concept set. We employ the Spearman's ranking correlation coefficient to measure the final score. According to the definition of Spearman's ranking correlation coefficient, the score is equal to 1 when agreement between the two rankings are the same, and -1 when one ranking is the reverse of the other.

4.3 Evaluation Setup

We recruited 13 persons to be involved in the user experiment for concept recommendation. Diverse concepts are suggested by our subjects as shown in Figure 4.1. As we can see from the figure, the number of concepts increases significantly when less agreement is achieved, from 13 votes to 2 votes. Concepts with only 1 vote are ignored in our experiment because one subject's suggestion means very little in terms of a common understanding of concept selection.

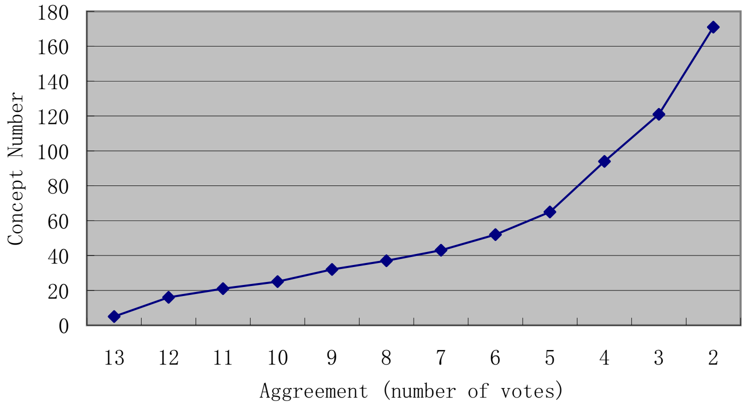


Figure 4.1: Concept number vs. agreement.

We first concentrate on a smaller concept set in which most concepts are selected with *agreeement* $\geq 50\%$. When too few concepts are selected for a topic, more concepts with a smaller agreement will also selected in order to make each topic have at least 5 concepts selected. In this concept set, there are a total of 85 concepts selected for our evaluation experiment. The whole group of concepts are shown in Table 4.4 as a universal set organized into general categories of objects, scene/setting/site, people and events, from which the 85 concepts with high relevance rating are highlighted in italics.

To test the robustness of different similarity measures used in our density-based concept selection algorithm, we also carried out experiments on a larger concept set. This concept set involves concepts selected with less agreement among users ($vote \geq 2$), forming a broader set of 171 concepts.

The distribution of all 171 concepts across activities is depicted in Figure 4.2. As shown in Figure 4.2, most activities have a number of concepts between 10 and 20 and the overall average concept number for all 23 activities are 15. Among all activities, 'Cooking' has more relevant concepts selected as more visual concepts are involved and are helpful to identify the activity, such as various kitchen items and food which are very specific. Activities like 'Using phone', 'Reading', 'Pet care'

Table 4.4: Experimental concept sets

Objects	*plate, cup, cutlery, bowl, glass, bottle, milk, drink, fridge, microwave, cooker, water, cloth, clothes, glove, soap, hanger, screen, keyboard, monitor, TV, remote control, basket, trolley, plastic bag, mobile phone, phone screen, book, newspaper, notebook, paper, handle bar, steering wheel, car, bus, bicycle, pet, road sign, traffic light, cat, yellow pole, chair, laptop, projector, pram/buggy* advertisement, back mirror, back of seat, bag, bar code, bin, bucket, cabinet, candle, chopping board, chopstick, coaster, computer, cupboard, dashboard, kettle, kitchen items, knife, label, lamp, light, menu, milk bottle, mirror, mouse, napkin, pan, pen, pot, press, product, seat, sign, sofa, speaker, spoon, street light, wire, sun, table cloth, tap, tea pot, tile, tissue, towel, view mirror
Scene / Settings / Site	*indoor, outdoor, office, kitchen, table, sink, basin, toys, shelf, cashier, door, building, fruit, vegetable, deli, food, road, path, cycle lane, sky, tree, dark, window, inside bus, shop, inside car, projection,* bright, ceiling, colorful, colorful light, corridor, counter, fridge inside, fridge outside, furniture, grass, home, house, kitchen counter, living room, park, queue, restaurant, stage, stair, street, wall
People	*face, people, group, child, hand, finger,* back of people, back of person's head, crowd, cyclist, head, people sitting, presenter, staff
Event	*hand washing, hanging clothes, hand gesture, finger touch, page turning, presentation, taking notes* holding cup, laughing, peeling, pouring, shaking hand, sitting, standing, typing, walking, walking child, washing

and 'Going to cinema' tend to have relatively similar images within one single event sample, therefore have less concepts recommended.

To measure semantic similarity, we employed both taxonomic similarity and contextual similarity using the ontologies of WordNet and ConceptNet, respectively. For taxonomic similarity, we also compared 5 mainstream similarity measures which are those of *Wu* and *Palmer* (*W&P*), *Leacock* and *Chodorow* (*L&C*), *Resnik* (*Res*), *Jiang* and *Conrath* (*J&C*), *Lin* all of which were introduced and described earlier in Section 3.3.2. Contextual similarity is obtained by spreading activation through ConceptNet links. After normalising by textual processing, the word-word semantic similarity is first calculated and then combined to get phrase-level similarity for conjunctive concepts composed of multi-words.

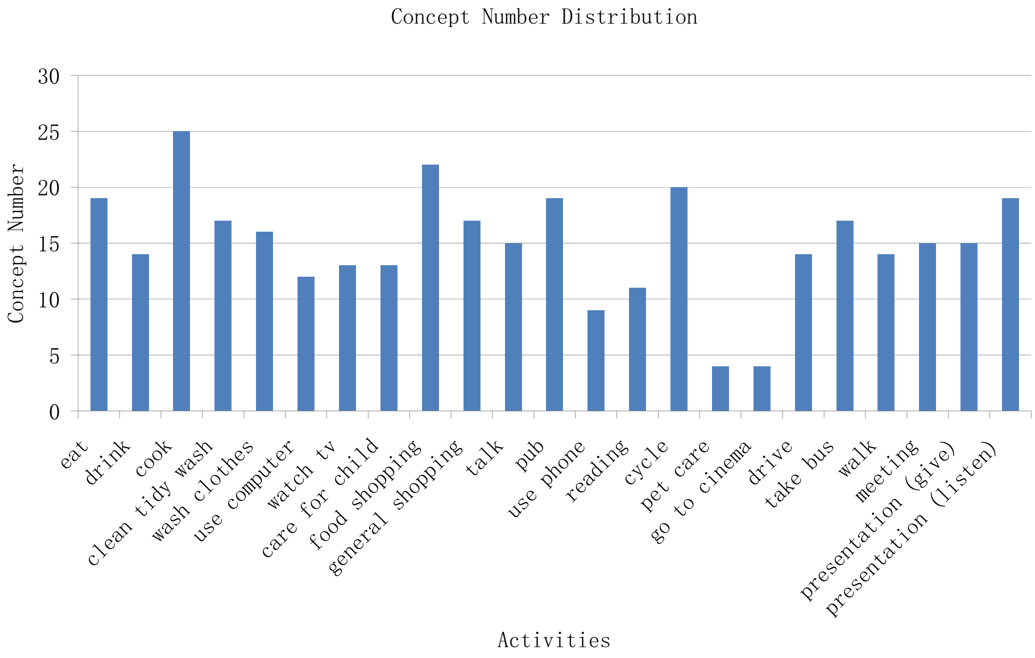


Figure 4.2: Concept distribution.

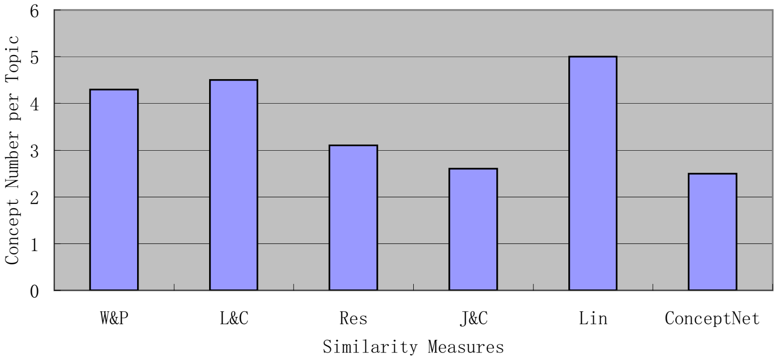


Figure 4.3: Average number of concepts selected per event topic.

The concept-concept similarity and topic-concept similarity are both used in our density-based concept selection algorithm to cluster the most similar concepts in the same clusters with corresponding event topics. The output concepts from hierarchical clustering are first analyzed to show the diversity of result concepts by different semantic similarity measures. The average number of concepts selected per event topic is depicted in Figure 4.3. Though there is not much difference in the average concept number per topic, *Lin* selected more concepts than the others. On average, 5.0 concepts are selected by *Lin*, compared to *Jiang* and *Conrath* and *ConceptNet* which both select 2.6 and 2.5 concepts per topic respectively. The same trend is shown in Figure 4.4 from which the proportion of selected concepts (for all topics) in the universal concept set (85 concepts) is shown.

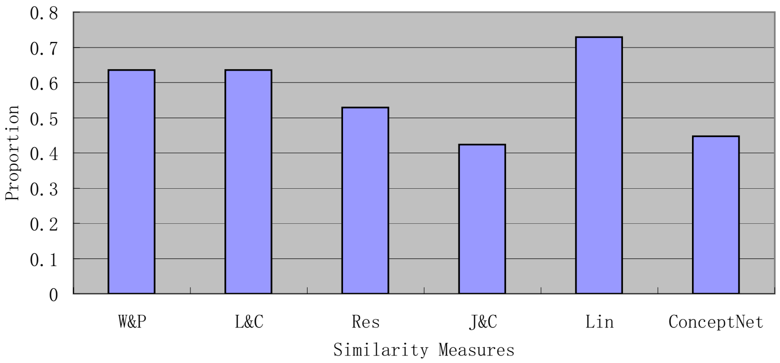


Figure 4.4: Proportion of selected concepts in the concept set.

4.4 Result Evaluation

The experimental results are assessed to compare the performance of applications of the two prevalent ontologies, WordNet and ConceptNet, for semantic density-based concept selection in a lifelogging domain. Our density-based concept selection and re-ranking algorithm involves several steps including similarity calculation, agglomerative clustering, similarity ranking and so on. Therefore, we evaluate the experimental results in manifolded ways.

4.4.1 Evaluating the Clustering Algorithm

Our algorithm first applies clustering to group semantically related concepts based on a similarity measurement. Group consistency is first calculated for each ontology to assess the clustering performance of our agglomerative algorithm in capturing the semantic relationships in everyday life events. The comparison of all above referred ontologies are shown in Figure 4.5.

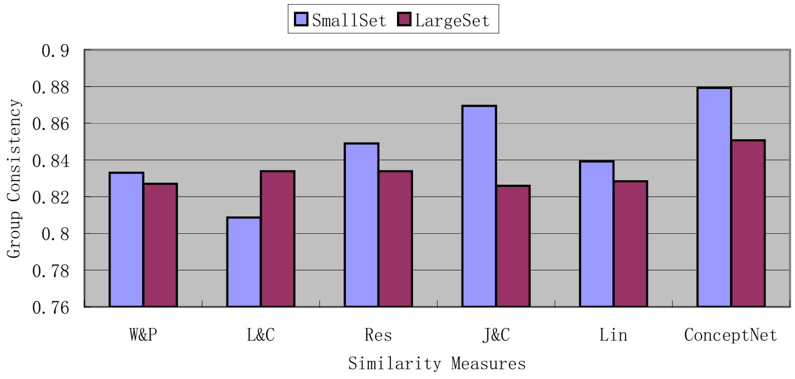


Figure 4.5: Group Consistency comparison.

The assessment is first carried out on a small concept set (85 concepts) as shown by blue bars in Figure 4.5. As we can see, ConceptNet-based similarity shows more consistency compared to the other similarity measures. Using the same concept set and agglomerative clustering algorithm, this can denote that the similarity values returned by our spreading activation from ConceptNet are more suitable to reflect the semantics of everyday activities. We increased the testing concept set by also applying the larger concept set (171 concepts) as shown by red bars and we found that *ConceptNet* still outperforms the other similarity sources.

In Figure 4.6, the precision of selected concepts is compared for each topic on *ConceptNet* and *Lin*. Although *ConceptNet* selects less concepts for each topic as shown in Figure 4.3, the precision outperforms *Lin* on more topics. There are 9 topics on which the precision for *ConceptNet* is above 50% while *Lin* only has 3 topics. Since *ConceptNet* is contextual ontology for common sense, it captures more contextual relations rather than taxonomic relations. Therefore, more context-related concepts are selected which increases the overall accuracy. For instance, 'keyboard', 'screen' and 'monitor' are all selected by *ConceptNet* under the topic of 'Using computer'.

These concepts are not taxonomically tight enough in WordNet, which can be interpreted as a long path between the nodes in the hierarchical lexical ontology. However, they are tightly connected under the same context of 'Using computer' showing that *ConceptNet* can more accurately reflect everyday concept relationship.

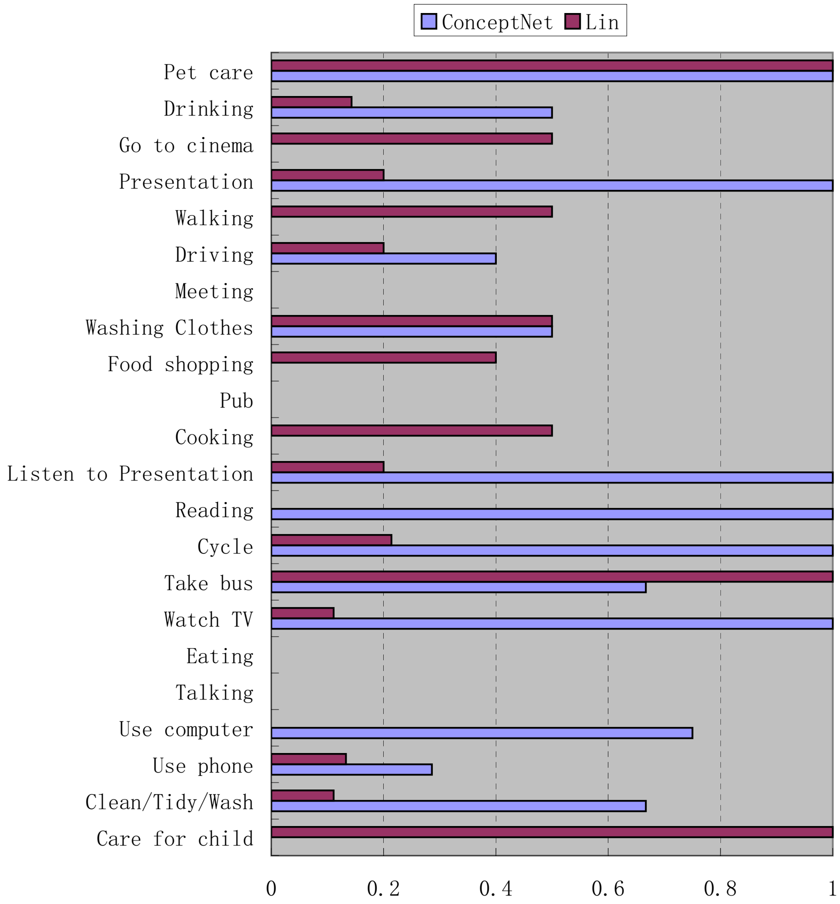


Figure 4.6: Comparison of precision.

In the larger concept set, the semantic similarity calculation is also performed first for these 171 concepts and topics and then goes through a hierarchical clustering algorithm which has been shown to be efficient in finding groups of relevant concepts in the concept space [169]. The output concepts are compared on a topic basis with the groundtruth pooled in the user experiment. Comparison is done on *SetAgreement* and *RankCorrelation* to evaluate the performance of different similarity measures. Because the topics are not uniform in assessing the performance, we do not use the average result over all topics which has no meaning on $SetAgreement$ and $RankCorrelation$. The investigation results of our density-based everyday concept selection are demonstrated in Figure 4.7 and Figure 4.8.

The performance of different similarity measures investigated in our experiment is compared in Figure 4.7 on the metric of $SetAgreement$. We find that ConceptNet-based concept selection has the highest median value and better quartile score than WordNet-based measures. Among WordNet-based similarities, *Leacock* performs best on $SetAgreement$ but does not show advantages on $RankCorrelation$ as shown in Figure 4.8.

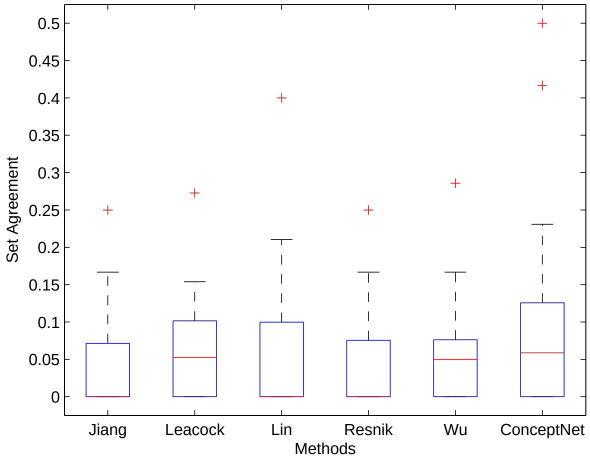


Figure 4.7: Comparison of Set Agreement.

ConceptNet-based concept selection results in the highest median and quartile scores on $RankCorrelation$. *Jiang* achieves the best performance among WordNet-based similarities, but is still out-performed by $ConceptNet$. Finally, we can reach a conclusion that ConceptNet-based similarity performs the best not only on the concepts selected (as implied by $SetAgreement$), but also on the ranking of these concepts (as implied by $RankCorrelation$). The contextual ontology is thus more suitable and efficient in everyday concept selection for the lifelogging domain.

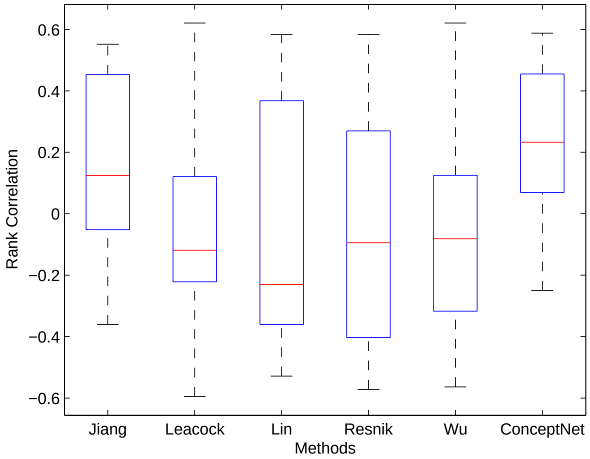


Figure 4.8: Comparison of Rank Correlation.

4.4.2 Similarity Ranking Assessment

Similar to group consistency, we define pairwise orderedness [66] to evaluate ranking performance of our algorithm, as the following formula:

$$PO = \frac{|\mathcal{C}| - \sum_{(c_i,c_j)\in\mathcal{C}} IC(O,R,c_i,c_j)}{|\mathcal{C}|} \tag{4.5}$$

where

$$IC(O,R,c_i,c_j) = \begin{cases} 1 & \text{if } R(c_i) \geq R(c_j) \text{ and } O(c_i) < O(c_j) \\ 1 & \text{if } R(c_i) \leq R(c_j) \text{ and } O(c_i) > O(c_j) \\ 0 & \text{otherwise} \end{cases} \tag{4.6}$$

$O(c)$ is equal to 1 if concept c is selected as a ground truth concept in user experiment. Otherwise, $O(c)$ is equal to 0. $R(c)$ is the final score for concept c returned by the similarity ranking. Concept pair set $\mathcal{C}$ has the same definition as given in the formalization of group consistency in Section 4.2.

The performance comparison of ontology similarities using pairwise orderedness is shown in Figure 4.9 on the small concept set (85 concepts). *ConceptNet* similarity outperforms the other measures in most cases for which the curve of *ConceptNet* (CN) is above all the other curves (activities before 'cook'). There are only four cases in which *ConceptNet* performs worse than WordNet-based similarity measures. They are 'cook', 'listen to presentation', 'general shopping' and 'presentation'. We also analyzed the poor performance of *ConceptNet* on these activity types to give explanations. For 'listen to presentation' and 'presentation', *ConceptNet* didn't perform well due to the lack of context information for the concept 'presentation'. By looking up the ontology structure of ConceptNet, we can find only two concepts that are contextually connected to 'presentation' with high correlation. They are 'fail to get information across' and 'at conference' and connected with 'presentation' by relationships 'CapableOf' and 'LocationOf' respectively. Therefore, it's hard to quantify related concepts in our concept set with a high similarity weight. In our experiment, 'general shopping' is introduced as a very general concept for which even humans can find it hard to decide the most related concepts. When concepts are selected in our user experiment, they usually are loosely connected with this topic. The poor performance on the topic 'general shopping' can be explained as the lack of specification of semantic context.

The evaluation on pairwise orderedness is also carried out on the larger concept set (171 concepts). As we can see from Figure 4.10, the comparison still shows that ConceptNet-based semantic similarity performs better than other similarity measures in most cases. In only three cases, *ConceptNet* does not perform as well as WordNet-based similarities, and those three cases are 'cook', 'presentation' and 'general shopping'. The reason for poor performance can be explained in the same way as when we were using the small concept set. Note that in the 'cook' topic, more procedures such as 'washing', 'peeling potatoes', 'stir frying', to name a few, are involved. The

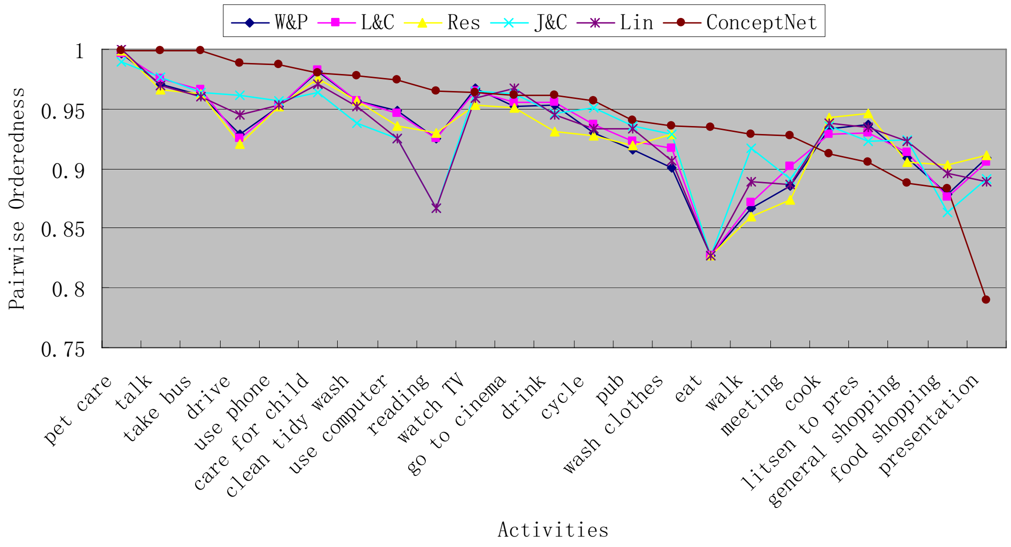


Figure 4.9: Comparison of Pairwise Orderedness (small set).

contextual diversity also makes it difficult for *ConceptNet* to return the contextual similarity correctly.

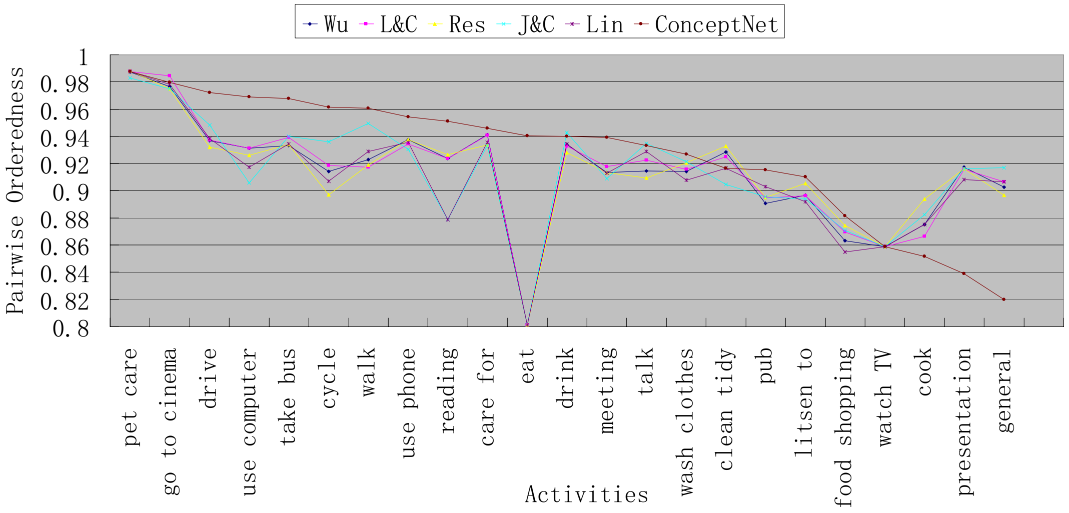


Figure 4.10: Comparison of Pairwise Orderedness (larger concept set).

Ranked concepts based on semantic similarity are also compared using metrics of *SetAgreement* and *RankCorrelation*. To simplify the comparison, we perform the evaluation on the smaller concept set with the selection on the Top-5 and Top-10 concepts returned by the similarity rank algorithm. The performance of different semantic similarity measurements are shown respectively in Figure 4.11 and Figure 4.12.

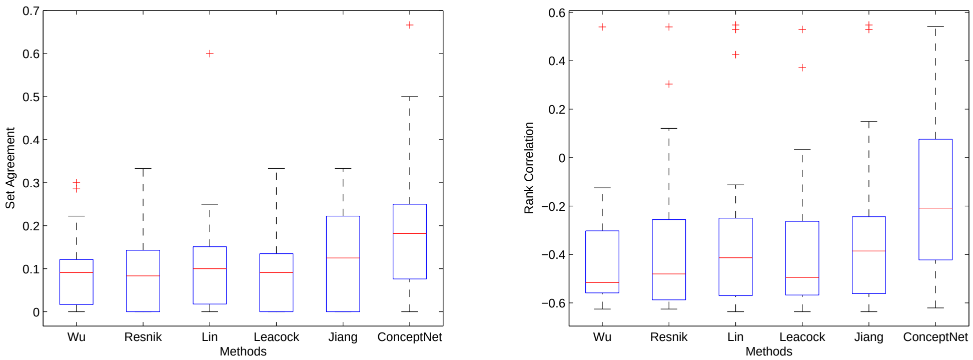


Figure 4.11: Comparison for Top-5 ranked concepts (smaller concept set).

As we can see from Figure 4.11, the advantage of using *ConceptNet* is more obvious as we select more concepts after similarity rank compared to very few concept seeds selected by clustering. In Figure 4.11, the ConceptNet-based algorithm outperforms the others not only in *SetAgreement* but also in *RankCorrelation*. The advantages of *ConceptNet* when Top-10 concepts are selected as depicted in Figure 4.12 show the robustness of our similarity rank algorithm. The rank algorithm propagates the similarity network and give higher weights to more relevant concepts based on the selected seeds selected by the clustering algorithm. When better seeds are selected, as done when using *ConceptNet*, the ranking algorithm tries to find more relevant concepts with regard to the already selected seeds.

4.5 Evaluation on TRECVid

The density-based concept selection algorithm was also evaluated on a data set provided as part of the TRECVid benchmark. We evaluated the performance of our algorithm with two benchmarks, namely the *Human Benchmark* and the *Collection Benchmark* [78]. The human benchmark is a human-generated concept selection pro-

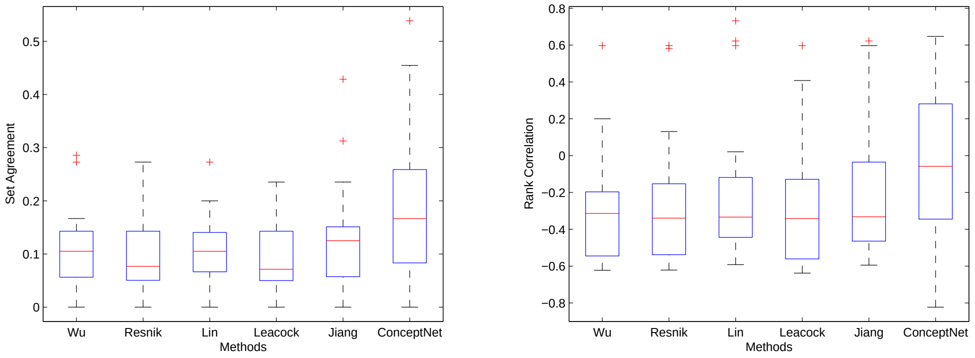


Figure 4.12: Comparison for Top-10 ranked concepts (smaller concept set).

cess that participants would consider useful for a query topic. Collection benchmark is generated from an already annotated corpus with concepts. This benchmark is usually created according to the relevance of concepts to a query, calculated by e.g. mutual information as we described in Section 4.2. In this case, a concept can be mapped to a query if it reduces uncertainty of a potential candidate being relevant to that query [78].

Using MediaMill video search engine, Snoek *et al.* manually extended both the number of concepts and the number of annotations by browsing the training video shots for TV news broadcast programme. The manual annotation process finally yielded a pool of ground truth for a lexicon of 101 semantic concepts according to [159]. In [159], the MediaMill team also published a collection of machine-learned detectors for these 101 concepts. To assess the evaluation on MediaMill's 101 concepts, we first ran our algorithm on this lexicon. Then, the results were evaluated on the topics shown in Appendix A, for which both human benchmark and collection benchmark results are released in [78]. Figure 4.13 demonstrates the comparison of ontologies in seed selection by agglomerative clustering on the two benchmarks.

As we can see from Figure 4.13, the performances of different ontologies change significantly when the application domain is changed from everyday lifelogging activities to TRECVid TV news broadcasting. *ConceptNet* does not perform that well

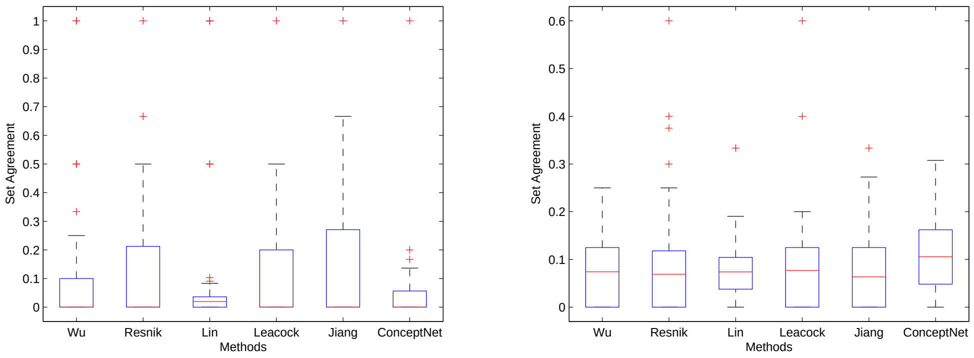


Figure 4.13: Comparison of Set Agreement (left: human benchmark, right: collection benchmark) in TRECVid.

compared to how it does in lifelogging activities. For the human benchmark, *Lin* performs worst among all WordNet-based ontologies in selecting useful concepts during the clustering procedure. The reason for the poor performance can be explained as due to the poor concept semantic consistency, as reflected by group consistency, which is shown in Figure 4.14. Group consistency directly reflects the clustering performance carried out based on concept similarity. When concept similarity returned by reasoning through ontologies like WordNet and ConceptNet can correctly reflect the pairwise concept relationship, the clustering will achieve better group consistency. From group consistency, we can see that *Lin* fails to cluster the semantically similar concepts together as the similarity scores are not properly returned for these concepts. Though $ConceptNet$ does not show much advantage when assessed in the human benchmark, it slightly out-performs the others on the collection benchmark. The collection benchmark tends to select more concepts for each topic [78], introducing more non-relevant concepts. This probably compensates for the defect that more concepts are selected as useful by using $ConceptNet$ for TRECVid, even though some are not that relevant.

In order to test the capability of semantic density-based concept selection and ranking, assessments on the final ranked concept lists are also carried out. In this

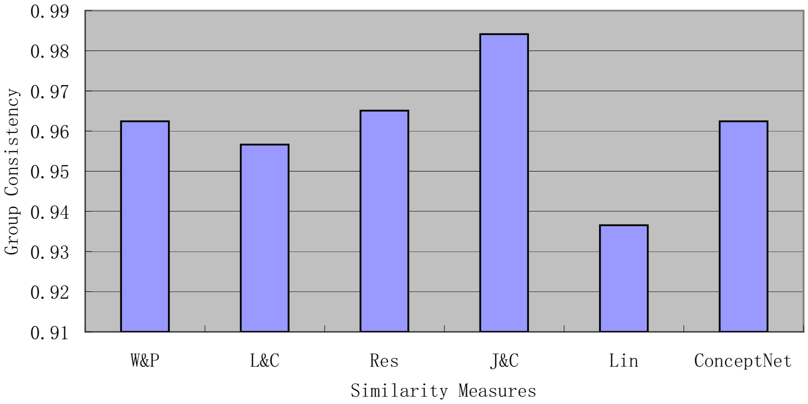


Figure 4.14: Comparison of Group Consistency (MediaMill 101 concepts).

assessment, the prevailing ontology-based concept selection proposed in [156] is employed as the baseline. The final results are evaluated on two benchmarks which are the human benchmark and the collection benchmark, whose results are released in [78]. The selected concepts in MediaMill's 101 concepts set are extracted and compared with automatically-selected concepts.

Figure 4.15 shows the comparison of density-based selection (top) and the baseline (bottom), on the human manual selection benchmark. We compared the two approaches each time using the first K concepts in the final selected concept lists for all topics, named as Top-K in Figure 4.15, to see the change of performance when the number of selected concepts increases. We remove the effect of similarity measures on the performance by choosing the same semantic measure as the baseline, which is *Resnik* similarity measure. Though there seems to be no significant improvement by our algorithm at small Top-K values, our approach achieved better results when $K > 7$. Because we are using the same similarity measure here, which is *Resnik*, the most relevant concepts returned by the measure shouldn't be too different. Therefore, the performance of the very top concepts in the final list is similar. However, the similarity ranking algorithm applies global similarity relations to ranking concepts. The useful concepts which might not get high similarity values for the topic are boosted by the concepts with high similarity.

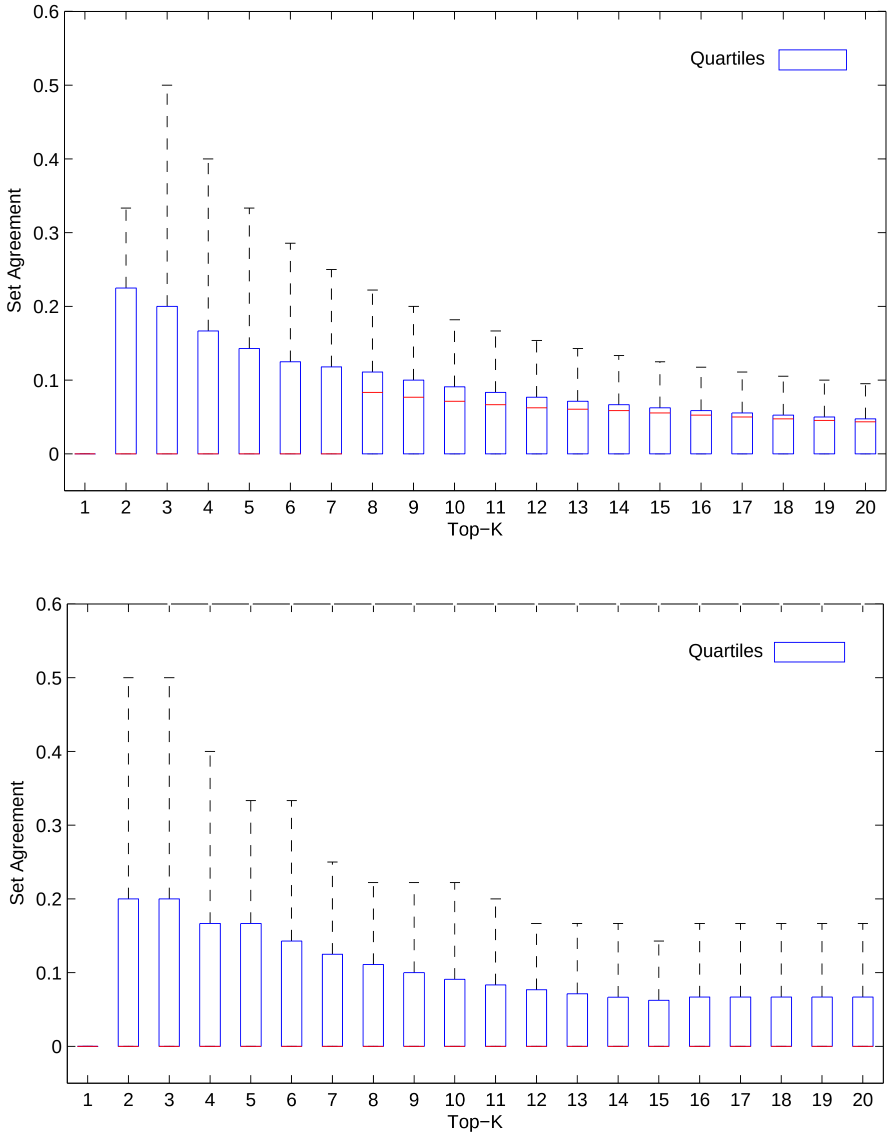


Figure 4.15: Comparison of Top-K concepts (human benchmark).

The same advantage is shown in Figure 4.16 on the collection benchmark. Our algorithm achieves higher median values at earlier K values ($K = 4$) than the baseline ($K = 6$). The median and quartile values remain better than the baseline especially when K has a high value.

Across the two domains we investigated when applying our automatic concept

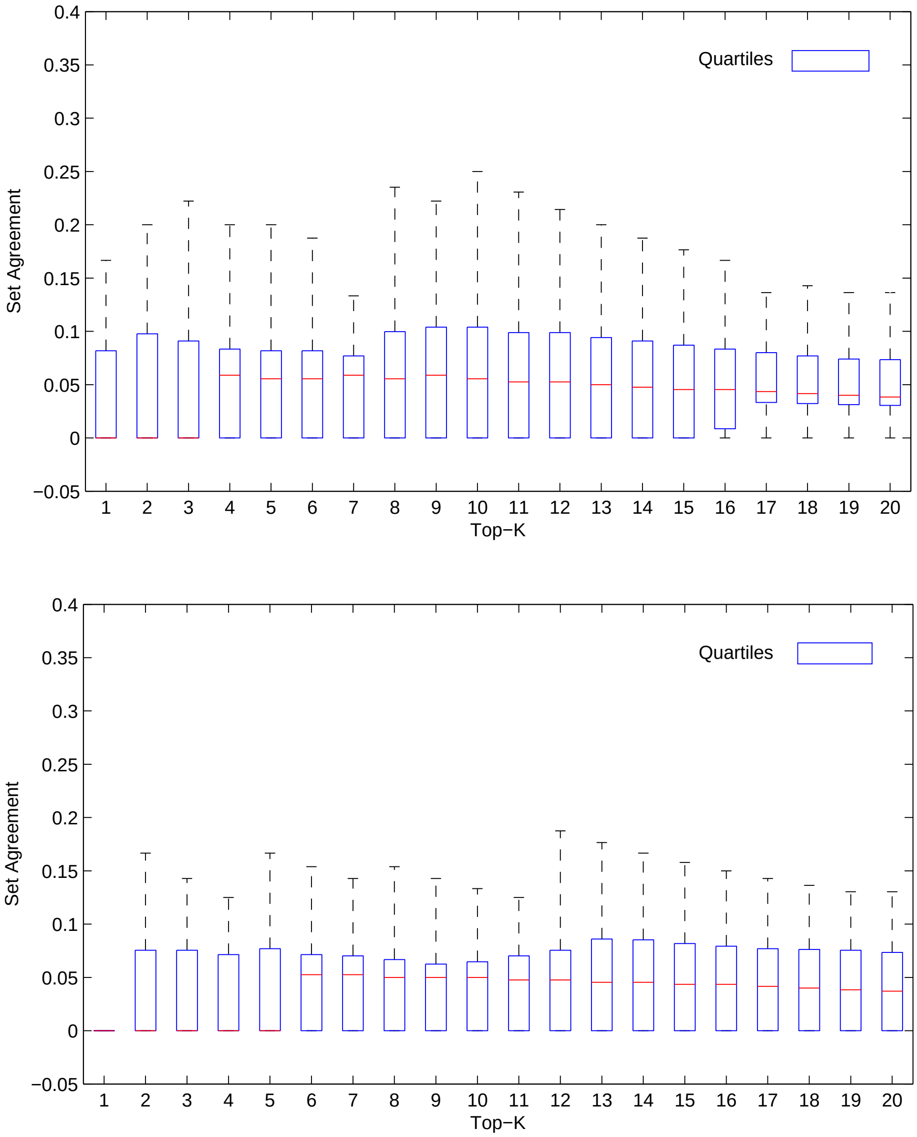


Figure 4.16: Comparison of Top-K concepts (collection benchmark).

selection algorithm, the summary points we can conclude from our experiments are as follows:

- The density-based concept selection utilizes the global similarity of all topics and concepts in concept selection and ranking. The advantages are shown in candidates selection and ranking the concept from a global view. The performance is better than the baseline in selecting the most relevant concepts for the given topic. The algorithm can also be used as a computational platform for various domains like lifelogging and news video retrieval.

- The candidate concepts selected by clustering depend on grouping consistency. Usually, the similarity measures which correctly reflect the semantic relationship between concepts can obtain better group consistency, as demonstrated by the good performance of *ConceptNet* similarity in lifelogging.

- Our application of contextual similarity obtained from spreading activation of ConceptNet performs the best in lifelogging concept selection. *ConceptNet* similarity better reflects the semantic relationship of everyday activities and concepts because they are more contextually relevant in the lifelogging domain.

- Contextual similarity does not show the advantages it has when the application domain changes from lifelogging to TV news broadcasting. However, most lexicon similarity obtained from WordNet performs well in capturing semantics between news topics and concepts.

4.6 Summary

In this chapter, we assessed the performance of various similarity measures using lexicons in lifelogging domain and TRECVid TV news broadcasting domain. To answer research questions (RQ1) and (RQ2) we proposed at the beginning of this thesis, in Chapter 3 we exploited the methodology of automatic concept selection by applying a density-based selection approach in concept space. Since the premise of this algorithm is the automatic reasoning on concept ontology, in this chapter we assessed various ontological similarity reasoning measures on two comprehensive concept ontologies which are WordNet and ConceptNet. The performance of our concept selection algorithm is also evaluated to reflect the effectiveness of clustering and ranking algorithm using different similarities. The final ranked concept list is also evaluated on TRECVid benchmark topics and concepts using the prevailing ontological selection approach as the baseline. Both evaluations on lifelogging and TRECVid domains demonstrated the efficacy of our algorithms.

Chapter 5

Fusion of Event Semantics for Lifelogging

Mobile devices are becoming ubiquitous in everyday life and digital media is proliferating. We find that this is happening not only in online social sharing but also in lifelogging. Because of the variety of activities that people usually engage in, a wide range of semantic *concepts* referred to earlier will appear in visual lifelog media, which in turn increases the challenges in developing automatic concept classifiers for such a diverse range. As a demonstration, we took the 85 concepts investigated in Chapter 4 and analyzed the properties of these concepts as reflected in the visual lifelogging domain. Even though these 85 concepts are far from representing a universal concept set encountered from the lifelogging point of view, they can reflect the common characteristics of concept semantics. Figure 5.1 shows a histogram of the frequency of appearance of these 85 concepts based on a manual annotation of a collection of 12,000 SenseCam images collected from within our group. From this figure we can conclude that:

1. The distribution of concepts is imbalanced. Some concepts' frequencies are extremely high while some are much lower. This imposes a burden on automatic

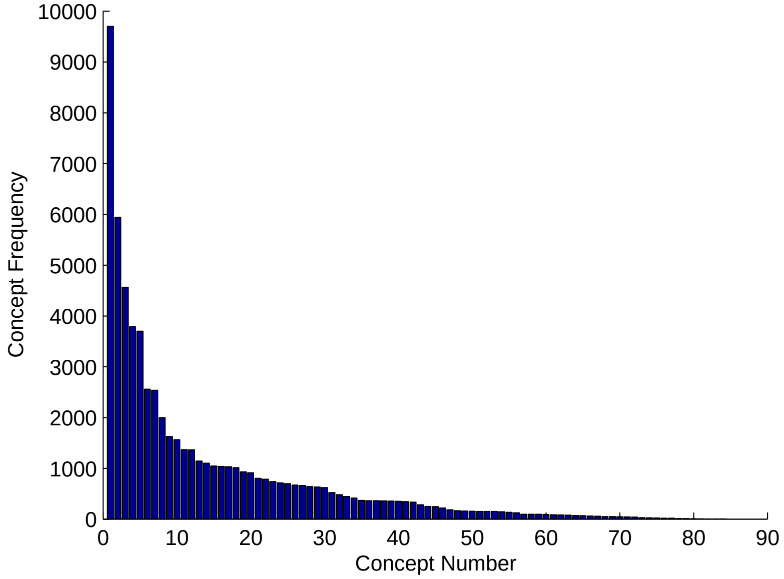


Figure 5.1: Concept distribution among SenseCam images.

concept classification for which balanced training data is necessary for better performance. Especially, a reduced frequency of concept occurrence makes it difficult to develop good detectors due to the absence of enough positive training instances for the learning machines.

2. The span of concepts is wide which means that many concepts can be involved in visual lifelogging. This property raises the challenge of high computational complexity.

3. The contributions of concepts are different in interpreting the semantics of events and appropriate concepts are needed. Concepts with very low frequency can incur noise and error easily if used in the concept space. An appropriate weight for concepts in the new space are needed for later stable representation or classification of events.

4. Each input image might have more than one concept, which raises the problem of mapping the low-level features from a single image to multiple concept classes.

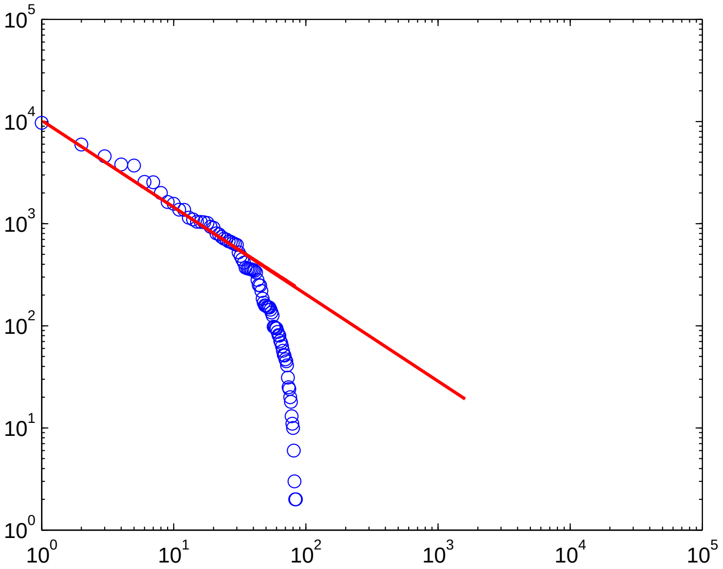


Figure 5.2: Concept distribution – logarithmic scale plot.

It is interesting to see that the distribution of concept frequencies has a good fit with Zipf's law. If we plot the frequency and concept rank according to decreasing order of frequency in logarithmic scale, the curve shows the pattern to be near-linear especially for concepts with high frequency, as shown in Figure 5.2. The linear relationship shows a simple relationship with frequency and rank for concepts, though the distribution of low-frequent concepts seems not to fit Zipf's Law as well in Figure 5.2. That is because of the idiosyncratic concept selection we used in our work which is the same reason as described in [69]. In Figure 5.2, the curve is first fit as power function before being plotted in logarithmic scale. The linear relationship for concept distribution yields a line with slope -0.841 which is very close to the theoretical value of -1.

Theoretically, the maximum number of possible concepts in lifelogging, based on the distribution can be estimated at more than 10,000 (see the intersection of the red line and horizontal axis). Since it is impractical to build so many concept classifiers and the effort to annotate that many concepts would be huge, in Chapter 3 we dealt with research questions (RQ1) and (RQ2) of the selection of concepts which can significantly reduce the burden of concept annotation and classifier training. However, the differences in occurrence frequency poses another challenge which affects accuracy of the concept detection because of imbalanced positive-negative sample distributions and visual diversity. This is also shown by Lin and Hauptmann in [101] where, the authors indicated that frequent concepts and scene-based concepts should be given higher priority as well as concepts which could benefit most search queries. It is not hard for us to imagine that the performances of concept detectors can vary significantly due to the above discussed differences of concept characteristics. The prior knowledge of concepts which can be modeled in concept ontology can reflect the inherent property of concepts and relationship between them, hence can be used to leverage the detection performance of different concept detectors. In this chapter, we will first discuss the ontological multi-concept classification before we apply the concept detection result for further event-level processing.

After talking about ontological multi-concept classification, we will turn to the discussion of event-level concept aggregation and semantic activity detection. Both of them try to answer the research question of (RQ3) by handling concept diversity at event-level. Different from each other, our concept aggregation algorithm to be proposed in this chapter fuses image concepts from a static view without considering the dynamic patterns of concept appearance. On the contrary, our high-level semantic activity detection tries to model the time-varying concept patterns from a dynamic view. As an application of even-level concept aggregation, we apply our algorithm into the selection of semantic event representation. In the experiment, the proposed algorithms will be evaluated together with corresponding applications in the last section of this chapter.

5.1 Event-based Visual Processing

In our research, visually processed events are the basic unit to be interpreted semantically by modern multimedia retrieval and Semantic Web technologies. Before we move on to a discussion of visual event processing, it is necessary for us to first have a conceptual definition of an event. Actually, *"events"* have been assigned various definitions in different research domains. As described in [177], an event is defined as a pattern when it is matched with a certain class of pattern types. This kind of pattern matching is named as an event in pattern recognition, while in the signal processing field, when a status changes in the signal this trigger is also viewed as an event. These categories of events are very similar to the definitions of events in some information systems, for which certain changes of system states or the occurrences of pre-defined situations are all regarded as events. Though these definitions are useful in some event analysis systems, in lifelogging, a definition which reflects an event's role in human understanding of everyday experience is needed. In [177], an event is regarded as a symbolic abstraction for the semantic segmentation of happenings in a specific spatio-temporal volume of the real world. The spacial and temporal attributes of events help us to organize our memories of life experience episodically. This has been shown in the neuroscience area in work such as [180], in which transient changes in neural activities are detected at event boundaries when participants are shown video depictions of everyday activities passively or asked to do active segmentation on them. This notion of an event is also accepted as a fundamental concept in the multimedia mining field as shown by Xie *et al.* in [175]. We use a similar definition of an event as a "real-world occurrence at specific place and time". Under this definition, the meaningful structures with spatial and temporal properties in lifelogging, like "Going to work", "Watching TV at home", "Talking with friends", etc., are all events.

As a general characterization of event contexts, the maxim of five "W"s and one "H" for reporting real-world events in journalism is used to represent the aspects of events, namely who, what, where, when, why, and how [178, 175]. The pervasive adoption of computing devices especially mobile devices increases the volume of multimedia data captured for real-world events. The multimedia resources related to events vary with the triggering of events in the form of heterogenous data types such as image, video, text, sensor readings and so on. Among these media data, visual images and videos contain more information and we call this kind of event processing, "visual event processing". These captures of events can reflect the who, what, where and when aspects of multimedia data and enhance the probability of further interpretation of event semantics like why and how. However, according to [170], these media are descriptions of the event rather than the event itself. That is to say, the media contain partial descriptions of the real-world event and the semantics of an event needs to be inferred from the captured media [181].

Visual lifelogging is a typical application of visual event processing. In various adoption fields of lifelogging such as memory aids, ADL analysis, and so on, a full understanding of events is necessary for better event retrieval and representation. However, there is still comparatively little metadata labeled on the multimedia data representing event semantics. Finding desired events with such little metadata is faced with many difficulties from large amount of media especially long-term lifelogged data. In visual lifelogging, much work has been done in event segmentation [53], event representation [51], life pattern analysis [86], event enhancement [50] and so on. These works still focus on the low-level visual features or raw sensor data processing. The semantic gap between events capture digitally and the human understanding of events are now fully bridged. In [34], the idea of semantic concept detection is explored to detect high-level concepts (such as indoors, outdoors, people, buildings, etc.) using supervised machine learning techniques. Though often used within video retrieval, this semantic indexing method has shown its capability in relating low-level visual features to high-level semantic concepts for visual lifelogging. Similar work has also been done by [52] in which the detected everyday concepts are applied into life pattern analysis. Though current concept detection can index lifelog visual media with meaningful annotations of concepts, the annotations are still handled at the image level. Efficient event indexing and management tools can not be provided by current image-level semantic annotations. In the following sections of this chapter, the challenges of fusing image-level semantics for event detection and representation are discussed. Algorithms are proposed to deal with these challenges and the evaluation is given in the experiment section.

5.2 Multi-Concept Event Semantic Aggregation

In visual lifelogging, successively captured images may have quite different visual appearances and a variety of concepts detected, unlike traditional video for which two successive frames within the one shot will be visually very similar. This makes it impossible to use the concepts from one single lifelogged image to infer the semantics of a whole event. The concept diversity in lifelogging events not only challenges event representation but also poses difficulties in multi-concept detection. The problem of event-level concept aggregation is tackled in this section of the thesis.

5.2.1 Ontology-based Multi-Concept Classification

The accuracy of a concept detector/classifier is always an important factor in providing satisfactory solutions in multimedia information retrieval, to map low-level feature to high-level concepts. To interpret the semantics of lifelogging events, accurate concept detectors are needed to extract concepts from image readings which are the main information sources and can imply more semantics than other sensor readings in everyday event interpretation. The concept detectors based on the processing of SenseCam images are therefore crucial for efficient lifelogging event-based organization and retrieval.

To decide on the appearance of concepts in visual media, machine learning approaches like Support Vector Machines (SVM) [163] are widely used to find a satisfactory separation (usually a hyper plane) between positive and negative concept instances in a high-dimensional space projected from the low-level perceptual image feature space, through the transformation of kernels. The result of this classification is returned as a confidence value to reflect the distance between an instance and the trained hyper plane. When the confidence of the positive class is high enough, we can annotate the target instance with the existence of concepts. The goal of multimedia indexing is then achieved through this mapping from image features to textual annotations. In concept detection, the applications of machine learning usually make the common assumption that the classifiers for a set of concepts are independent of each other, and equally weighted in terms of importance. The intrinsic relationships between concepts are neglected under this assumption. Eventually, this assumption ends up with multiple isolated binary classifiers and leads to a result of ignorance of concept semantics. The notion is likely to suffer from the shortcomings of misclassification or inconsistency between the detected concepts.

A concept ontology provides a methodology to model concept semantics and improve the one-per-class classification accuracy if the semantics of concepts can be fused in the detection procedure. In our solution, the lexicon of lifelogging concepts are constructed as a concept ontology and concept relationships are applied in a top-down approach to adjust the classifier outputs, making the detected confidence score reflect the semantic relationships between concepts. In our experiment, the semantic enriched multi-concept classification algorithm shows added value in improving detection performance despite the diversity of concepts in the lexicon. The detected concepts can then be fused for activity classification or event representation, which we will describe later in Section 5.3.3 and Section 5.4.

The intuitive notion of ontology-based multi-concept classification is to utilize concept semantics formalized in an ontology in order to improve detection accuracy by modifying the confidence for each concept in lifelogging. In our approach, we combined a concept ontology and the multi-class level integration proposed in [97] to achieve an ontology-based classification solution for multiple concepts. The procedure involves the following steps: firstly, the class prediction confidence is calculated by an SVM binary classifier. Each concept is given a confidence to represent the likelihood for the image to contain the concept. This basic SVM classifier is one-per-class discriminative concept detector without considering concept semantics as we described above. Secondly, two important relationships between concepts are considered and then formalized in the ontology, which are *Subsumption* and *Disjointness*. Subsumption is a relationship restricting the membership of a concept. Using subsumption, a taxonomic structure can be created between a concept and other concepts. By relating two concepts with disjointness, no instance of either class can be an instance of both classes. In concept detection, disjointness can be interpreted as that the two disjoint concepts can not co-occur in the same image. Both of the two relationships have intuitive guidelines we follow in designing our algorithm. Thirdly, the concept semantics is applied into the adjusting of concept prediction confidence to improve classification accuracy. This is done by learning the adjusting factor first from the correlation between detection performance and confidences of relevant concepts.

A snippet of the concept ontology we used in our lifelogging interpretation is depicted in Figure 5.3 and in Figure 5.4, in which the high level concepts of ‘Indoor’ and ‘Outdoor’ are highlighted respectively.

In Figure 5.3 and Figure 5.4, each concept in our lexicon is represented as one node of the tree. The subsumption relationships are visualized by arrows pointing from superclass concepts to subclass concepts. Another semantic relationship modeled

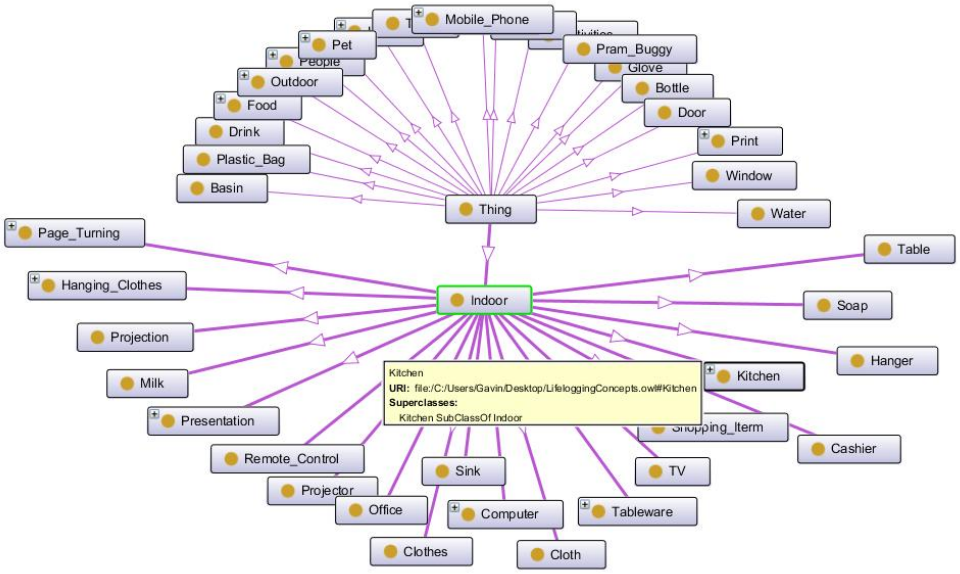


Figure 5.3: Ontology for multi-concept lexicon (indoor highlighted).

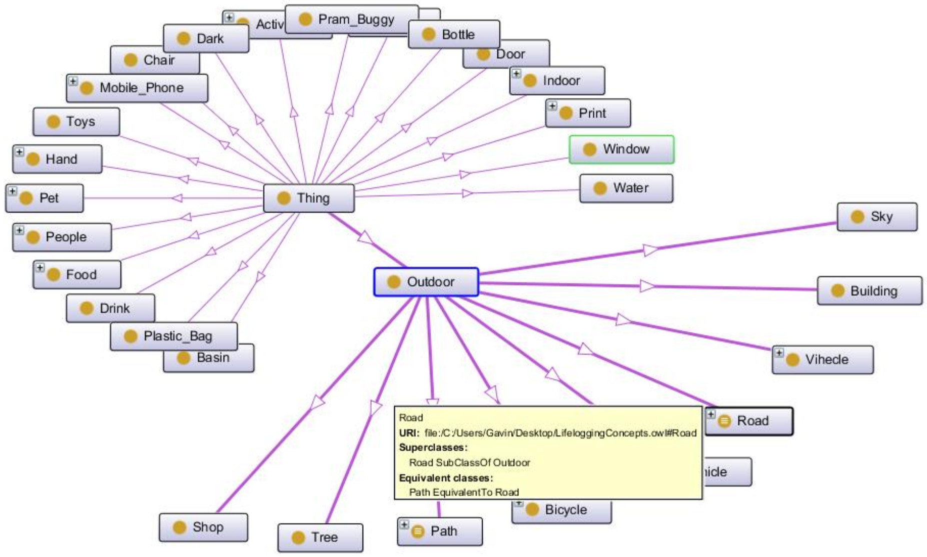


Figure 5.4: Ontology for multi-concept lexicon (outdoor highlighted).

in this ontology is disjointness. As standard Semantic Web languages which can explicitly specify the term relationships, ontology language OWL [114] and RDFS [28] vocabulary can be applied here for concept semantic modeling. More details of ontology construction and semantic description syntax like OWL and RDFS will be described in Chapter 6. Here we simply use their semantic modeling functions. A set of disjoint concepts are related using the owl:disjointWith constructor which asserts that one concept can not simultaneously appear in the same image together with a specified other concept. In Listing 5.1, the ‘Outdoor’ concepts is specified as a disjoint concept of ‘Indoor’. Meanwhile, all of the concepts have the same root and are derived from the concept ‘Thing’. The subsumption relationship is created by the property rdfs:subClassOf.

```
<owl:Class rdf:ID="Indoor">
  <rdfs:subClassOf rdf:resource="#Thing"/>
  <owl:disjointWith rdf:resource="#Outdoor"/>
</owl:Class>
```

Listing 5.1: Example of disjointness specification

As modeled in a standardized formulation, ontology-based concept detection utilizes the underlying concept relationships among classifiers to improve the final accuracy. The influence of the ontology on concept classification is performed by adjusting the confidence value calculated by the single concept classifier. Literally, the subsumption influence makes full use of the effects of parent concept nodes while disjoint influence considers the effects of disjoint concepts. In Figure 5.3 and in Figure 5.4 the parent nodes are the superclass of children nodes. The hierarchical structure not only reflects the semantics of concepts but also influences the concept detection performance of concepts at different hierarchical levels. As demonstrated by Byrne, Doherty, *et al.* in [34], the higher level concepts like ‘Indoor’, ‘Outdoor’, etc. have much better detection performances. One important reason for the performance difference is that concepts located at the lower levels of the ontology hierarchy tend to have less positive training data compared to those concepts at the upper levels [173]. As we can see from the ontology structure, only a few concepts have child nodes while most concepts are leaves in the tree. However, these leaf concepts are more specific concepts so that the detectors are less accurate than the more general concepts. With regard to the disjointness relationship, the disjoint concepts can not co-exist in the same image.

In determining the adjusted confidence by exploiting the subsumption and disjointness relationships, we define a target concept of an image x as c. The ascendant concepts and descendant concepts for concept c are denoted as $ASC(c)$ and $DES(c)$. Similarly, the disjoint concepts explicitly modeled in the ontology are $DIS(c)$. The confidence of image x belonging to concept c returned by the SVM classifier is represented as $Conf(c|x)$. Assume we have M classifiers which are one-per-class concept detectors for M concepts. Without employing the semantic relation of these classifiers, we directly binarize $Conf(c|x)$ to obtain the appearance of each concept in image x. From a set of disjoint concepts, the concept with maximum confidence is usually chosen as the final concept detected as $\omega = argmax_{1 \leq c \leq M} Conf(c|x)$. The adjusting of $Conf(c|x)$ with respect to concept semantics is now described in detail.

As we described, most of the concepts in the lexicon are leaf concepts. Though detectors for these specific concepts usually have lower accuracy than the more general ones, they also have a greater number of disjoint concepts at the same level or derived indirectly from other levels, which can be used to improve their accuracy. Aiming to apply the constructed ontology to multi-concept classification, we introduce the multi-class margin factor [97] [63]:

Definition 1. Multi-class Margin

$$t_m = Conf(c|x) - max_{c_i \in D} Conf(c_i|x) \tag{5.1}$$

where D is the universal set of disjoint concepts of c. Note that $D \supseteq DIS(c)$ because there are also concepts modeled implicitly as disjoint with c in the ontology. Indeed, D includes $DIS(c)$ as well as $DES(DIS(c))$, which are all descendants of disjoint concepts of c, and disjoint concepts of ascendent concepts above c, denoted as $DIS(ASC(c))$. These statements of disjointness can be asserted or inferred. The former is created directly by the ontology to assert the statement (using `owl:disjointWith` property). However, for the latter, a semantic reasoner is required to infer additional disjointness statements logically. Current Semantic Web technology already provides reasoners at various levels such as RDFS inference and OWL inference, to add inference to different application needs. A detailed description of Semantic Web inference will be given in Chapter 6. In our algorithm, the reasoner is embedded straightforwardly to leverage explicit statements to create logically valid but implicit statements.

To demonstrate the effect of the multi-class margin on detection accuracy, we plot the misclassified and correctly-classified image samples in Figure 5.5, using the 'Indoor' concept as an example. The detection confidences of concepts such as 'Indoor', 'Outdoor', 'Road', 'Sky', etc. are returned by standard SVM classifiers as we describe in Section 5.5.3. This figure visualized a data set of about 10,000 SenseCam images for which the ground truth of 'Indoor' concept is annotated manually. The disjoint concepts used to calculate multi-class margin of 'Indoor' are 'Road', 'Sky', 'Tree', 'Building', 'Grass' and 'Outdoor'. In Figure 5.5, the x-axis stands for the confidence of the 'Indoor' concept returned directly by the classifier, the y-axis is the multi-class margin calculated by Equation 5.1. A blue star stands for misclassified images while a red circle stands for correctly classified images. From Figure 5.5, we can easily find that there will be fewer misclassified instances when the confidence and multi-class margin are high. For most misclassified samples, the multi-class margin is lower than confidence and most misclassifications are located in the region with the multi-class margin lower than 0.6. Using a multi-class margin has been proved to be effective for a better separation between two kinds of instances in [97] and [63], achieving reduced classification errors. We also use a multi-class margin as a criterion to improve the classification accuracy in our work.

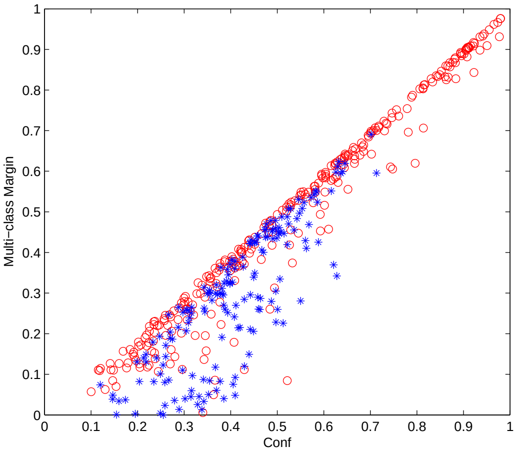


Figure 5.5: Concept classification results (Conf vs. Multi-class Margin).

To show how the multi-class margin improves concept detection accuracy, we plot the distribution of concept accuracy and multi-class margin in Figure 5.6, shown by blue "+" marks. In the figure, the accuracy-confidence distribution is also plotted for comparison purpose by blue circle style marks denoted by "$\circ$". Compared to the intuitive confidence-based classifier, the correlation between classification accuracy and multi-class margin shows greater advantage in Figure 5.6 for both 'Indoor' and 'Outdoor' detection. It's easy to notice that the multi-class margin has higher accuracy than the original confidence and converges earlier in both graphs.

This correlation between detection accuracy and multi-class margin can then be used to adjust the concept detection confidence. We modify the original confidence value $Conf(\omega|x)$ by the formula:

$$Conf = \sqrt{Conf(\omega|x) \times g(t_m)} \tag{5.2}$$

where function g is the adjusting factor and is calculated by fitting the sigmoid

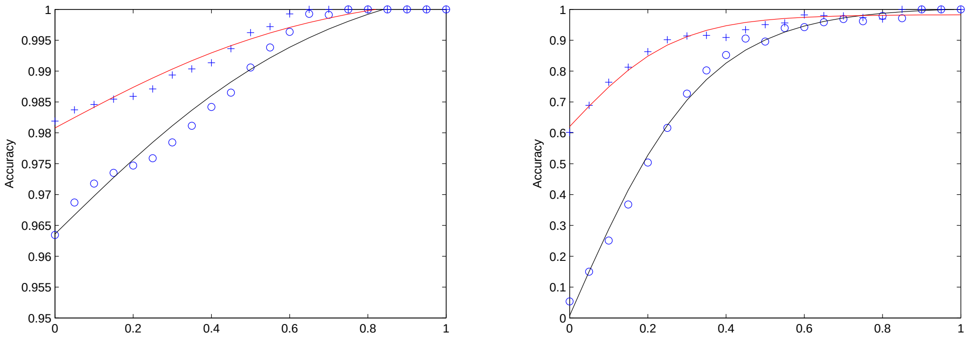


Figure 5.6: Correlation of accuracy and confidence/multi-class margin (left: indoor; right: outdoor).

function of the relationship between classification accuracy and multi-class margin reflected in Figure 5.6. In Figure 5.6, sigmoid functions are fit according to the accuracy-confidence/multi-class margin distributions. The curves of multi-class margin (in red) are located above those of original confidence (in black), achieving better performance for concept detection. The sigmoid function $g(x)$ we used for fitting the correlation has the form as follows:

$$g(x) = A + \frac{B}{1 + exp(-C \times x)} \tag{5.3}$$

The details of our implementation of ontology-based multi-class classification is shown in Algorithm 1. In Section 5.5.2, the evaluation of this algorithm will be elaborated.

```
Input:
$O$: Concept ontology model built for lexicon
$x_{training}$: Instances for parameter learning
$x_{testing}$: Instances for confidence adjusting
Output:
$Conf$: Adjusted confidences for $x_{testing}$ by $O$
Data:
$L$: Universal concept lexicon
$c$: Instance of concept
$DIS(c)$: all disjoint concepts of $c$
$Conf(c|x)$: Original confidence returned by SVM classifier
$t_m$: Multi-class margin
$A, B, C$: Parameters for sigmoid function
$g(t_m)$: sigmoid function value of multi-class margin
begin
    $O \leftarrow$ ReadOntology();                    // Read ontology into model
    $O \leftarrow$ InferOntology($O$);       // Perform semantic inference on $O$
    for $x \in x_{training}$ do
        for $c \in L$ do
            $Conf(c|x) \leftarrow$ SVMDetector($x, c$);          // Confidence by SVM
        end
        for $c \in L$ do
            $DIS(c) \leftarrow$ QueryDisjoint($c$,$O$);          // All disjoint of $c$
            $t_m \leftarrow$ MultiClassMargin($x, DIS(c)$);
        end
    end
    Learn parameters $A$, $B$ and $C$ from calculated $t_m$ ;
    for $x \in x_{testing}$ do
        for $c \in L$ do
            $Conf(c|x) \leftarrow$ SVMDetector($x, c$);          // Confidence by SVM
        end
        for $c \in L$ do
            $DIS(c) \leftarrow$ QueryDisjoint($c$,$O$) ;         // All disjoint of $c$
            $t_m \leftarrow$ MultiClassMargin($x, DIS(c)$);
            Calculate $g(t_m)$ with learned parameters $A$, $B$ and $C$;
            $Conf \leftarrow \sqrt{Conf(c|x) \times g(t_m)}$ ;
        end
    end
end
```

Algorithm 1: Ontology-based multi-class classification algorithm

5.2.2 Interestingness-based Concept Aggregation

Concept classification is implemented at image level to extract the potential semantics reflected by a single image. In event-based lifelogging, it is the semantics of *events* rather than single images from events, that is the focus to help a user to understand what he did, when and where a specific event happened and whom he was with during that period. However, when successively captured images have quite different visual appearance and a variety of different concepts detected, it's unrealistic to represent the semantics of a whole event by the concepts detected from one single image. Meanwhile, different concepts play different roles in interpreting event topics. For example, in analyzing concepts for a 'Meeting' event, we can detect such concepts as 'Indoor', 'Office' and 'Face'. As 'Indoor' is not a unique concept for 'Meeting' compared to other events such as 'Working', 'Shopping' that also have the concept 'Indoor' occurring, it should be ranked lower while concepts like 'Office' and 'Face' are better representations for 'Meeting'.

5.2.2.1 Event Concept Interestingness

To tackle the above difficulties we are faced with in lifelogging, an interestingness-based concept aggregation algorithm [1] is proposed in this section to fuse image concepts for appropriate event-level semantic representations. The interestingness-based concept aggregation is motivated by the notion that the best descriptive concepts for an event should be the most unique across the collection yet representative, in order to differentiate a given event from others; meanwhile the concept should also have relatively high frequency within the event. This is the same rationale as $tf \times IDF$ weighting in standard information retrieval.

In vector-based retrieval systems, the documents and queries are represented by vector descriptions in which each dimension corresponds to an elementary concept in the lexicon. In this multi-dimensional space, conceptual similarity can be easily obtained by measuring the geometric distance between the vectors. Traditional information retrieval systems apply $tf \times IDF$-like weight to quantify the coordinate of a vector along a dimension as the relative importance of the corresponding elemen-

[1] While choosing a proper word to describe the contribution of a concept to the representation of true event semantics, we had a discussion on the use of 'influence', 'uniqueness' or 'specificity' for this task. To keep consistency with the terminology referred to in [54, 168], we will inherit the usage of 'interestingness' for the rest of this thesis.

tary concept for the document (or query). Geometric distance such as Euclidian or Cosine can easily be applied to return the similar vector for a query. With the same notion, we extend the research problem as the following task: given a particular event and all consecutive images representing it and each image has concept appearances detected, the mission is to identify the best concepts representing the event and rank them according to their contribution to event semantics. Though some events like a holiday covers many days and a user's recall of his holiday might have longer time span, to reminisce more details of a holiday a user usually interprets events on a daily basis. To simplify the problem domain, we limit event coverage within the range of one day. That means we need to find the most representative concepts for an event with respect to the other events in the same day. The algorithm is generic and can easily be extended to a week or month basis which has broader time intervals.

With the same terminology as in the previous sections, we have the universe of concepts C. Let $\{E_1, E_2...E_N\}$ be the event sets in a specific day. Event E_i is represented by successive images $I^{(i)} = \{Im_1^{(i)}, Im_2^{(i)}...Im_m^{(i)}\}$. Each image Im_j^i might have several concepts detected, we assume the concepts appearing in image $Im_j^{(i)}$ are $C_j^{(i)} = \{c_{j1}^{(i)}, c_{j2}^{(i)}...c_{jn}^{(i)}\}$. Then the frequency of concept c occurring in event E_i is calculated in the form of $f(c, E_i) = \sum_{1 \leq j \leq m} 1\{c \in C_j^{(i)}\}$, where $1\{\cdot\}$ is the indicator function.

The weight for each concept $c \in C$ for E_i given the above assumption is:

$$w(c, E_i) = \frac{f(c, E_i)}{\sum_{1 \leq j \leq N} f(c, E_j) + \xi} \tag{5.4}$$

The definition above can satisfy the assumptions [54] as follows:

1) Frequently occurring concepts show the semantic consistency within the event and should be selected as concept candidates for the event.

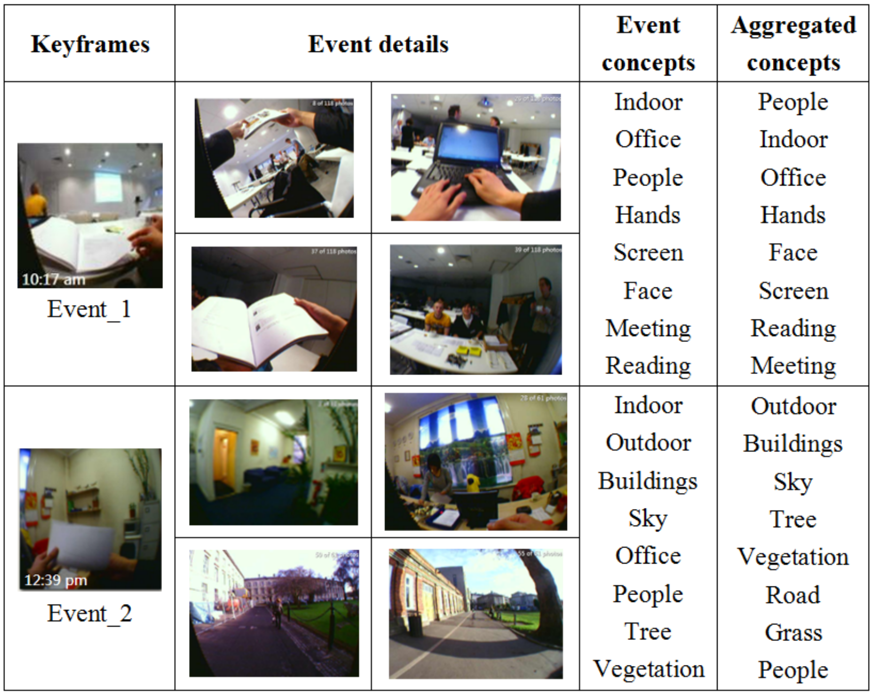
2) Concepts appearing more during E_i than the other events are more unique and should have higher weights.

Concepts detected at the image level are prone to noise and suffer from misclassification due to the limited performance of classification. ξ in the denominator of Equation 5.4 is used to filter misclassified concepts with very low frequency. However, the aggregation at the event level can filter the misclassified concepts and only consistent concepts having higher weight will be selected. We can imagine that event level aggregation of concepts is more robust than image level and this idea will be tested and verified in our experiments in Section 5.5.3.

5.2.2.2 Semantic Aggregation of Concepts

In the event segmentation stage, each event is separated from others using sensor readings from the SenseCam's onboard sensors [50] and a keyframe is selected as the best representative image for each event [51]. Though concept detection is easily affected by noise at the image level, our concept aggregation fuses the dominant concepts from the event level which shows greater robustness to concept detection noise. The fusion procedure returns the Top-k concepts for event E_i ranked according to concept interestingness as $\{c_1^{(i)}, c_2^{(i)} ... c_k^{(i)}\}$, where interestingness weight $w(c_j^{(i)}, E_i) \geq w(c_{j+1}^{(i)}, E_i)$. The choice of Top-$k$ value can be modified, which will be explored in the experiments in Section 5.5.3.

The main contribution of concept aggregation is representing events with a vector of concepts which not only reflects event semantics, but also facilitates event visual representation, i.e. keyframe selection. Some examples are shown in Figure 5.7 in which the resulting concepts from the aggregation algorithm are listed. Due to the disadvantages of the single concept classifier, only those concepts with high confidence can be regarded as true from each image. Thus some concepts which might be more relevant at the event level are easily missed. In Figure 5.7 we can see that the keyframe selected by our SenseCam browser [51] may be visually representative of the event but we are not sure if it is semantically representative. In Event_1, only two concepts



Keyframes	Event details		Event concepts	Aggregated concepts
10:17 am Event_1			Indoor Office People Hands Screen Face Meeting Reading	People Indoor Office Hands Face Screen Reading Meeting
12:39 pm Event_2			Indoor Outdoor Buildings Sky Office People Tree Vegetation	Outdoor Buildings Sky Tree Vegetation Road Grass People

Figure 5.7: Event-level concept aggregation.

can be detected from the keyframe, namely 'Indoor' and 'Office', forming a concept vector $C_{kf1} = \{Indoor, Office\}$. From these two concepts there is ambiguity as to the nature of Event_1. The aggregated method ranks the more unique concepts higher when their occurrence frequencies are high enough through interestingness weight vector $\mathbf{v_{e1}} = (0.037, 0.034, 0.022, 0.020, 0.011, ...)$, of which each value represents the weight for 'People', 'Indoor', 'Office', 'Hands', 'Face' and so on. These concepts have higher correlation with event semantics such as 'Talking' ('People', 'Face') and 'Using Computer' ('Hands','Screen'). These two types of activities reflect the core semantics of Event_1.

5.2.3 Vector Similarity in Semantic Space

To quantify the relationship between entities in the semantic space, we will discuss the similarity of concept lists. The $tf \times IDF$ weight is used as the most efficient weighting definition in the Vector Space Model [14] where both documents and queries are associated with t-dimensional vectors $\mathbf{v_j} = (w_{1j}, w_{2j}, \ldots, w_{tj})$, where each dimension is a weight and t is the size of the lexicon. Traditional vector similarity measures can be employed to quantify the relevance between two vectors, such as inner product $(\mathbf{v_i} \bullet \mathbf{v_j})$ or Cosine of the angle among those two vectors as:

$$sim(\mathbf{v_i}, \mathbf{v_j}) = \frac{\mathbf{v_i} \bullet \mathbf{v_j}}{||\mathbf{v_i}|| \times ||\mathbf{v_j}||} = \frac{\sum_{k=1}^{t} w_{ki} \cdot w_{kj}}{\sqrt{\sum_{k=1}^{t} w_{ki}^2}\sqrt{\sum_{k=1}^{t} w_{kj}^2}} \tag{5.5}$$

However, the semantic contribution of each dimension to the vector is ignored by these measures. Especially, it worsens the case if terms, which are concepts in image or video retrieval, cannot be detected perfectly. The noise introduced by imperfect concept detection will degrade the performance. For example, assume we have three semantic vectors: $\mathbf{v_1} = (0.1, 0.2, 0.1)$, $\mathbf{v_2} = (0, 0.2, 0)$, $\mathbf{v_3} = (0.2, 0.1, 0.2)$, whose components represent the weight for different concepts representatively. Though Cosine similarity $sim(\mathbf{v_1}, \mathbf{v_2})$ is equal to $sim(\mathbf{v_1}, \mathbf{v_3})$, we prefer $\mathbf{v_2}$ to approach $\mathbf{v_1}$ because they semantically emphasize the same concept. Besides, the low weights in $\mathbf{v_1}$ such as 0.1 are more likely to be affected by noise introduced by concept detection, making the similarity unstable.

With this motivation, we define the similarity which considers both set agreement and rank consistency of two concept vectors and apply the measurement in judicious selection of an event keyframe later in Section 5.4. The similarity is shown as the following equation:

Definition 2. Concept Vector Similarity

$$sim(C_i, C_j) = \frac{1}{|C_i \bigcup C_j|} \sum_{k=1}^{|C_i|} \sum_{l=1}^{|C_j|} \frac{1\{C_{ik} = C_{jl}\}}{abs(k-l)+1} \tag{5.6}$$

where C_i, C_j stands for two concept vectors aggregated by approaches described in Section 5.2.2.2, $|C_i \bigcup C_j|$ is the cardinality of the set consisting of the union of two concept sets. $abs(k-l)$ gives the absolute value of ranking difference for the same concept in two vectors. The added “1” in the denominator is used to avoid division by zero.

The concept vectors are regarded as high-level features for interpreting event semantics. To demonstrate the similarity for high level features, let’s revisit the examples in Figure 5.7. We choose the top 5 concept vectors for Event_1 for simplicity, which are $C_{e1} = \{People, Indoor, Office, Hands, Face\}$. According to the definition above, the similarity of C_{e1} and C_{kf1} for the keyframe is 0.2 for Event_1. With the same manner, the semantic similarity between keyframe ($C_{kf2} = \{Indoor\}$) and event for Event_2 is 0.028. Event_2 has much lower vector similarity due to the existence of sub-events with disjoint semantics of ‘Outdoor’ and ‘Indoor’.

5.3 High-Level Semantic Activity Detection

According to research result in the neuroscience area, experiments showed that humans remember their past experience structured in the form of events [180]. This poses another need for lifelogging tools to provide high-level topic detection facilities to categorize events for organization or re-experience use. Besides, the continuing progress of automatic concept detection for multimedia data like images, videos has shown satisfactory results, especially for some concepts or in specific domains. This has raised the probability to apply sophisticated approaches to fuse the detected results in achieving goals for which traditional methods lose capability. In [72], the semantic model vector (the output of concept detectors) has already been shown to be the best-performing single feature for IBM’s multimedia event detection task in TRECVid. It is important to realize that lifelog events such as sitting on a bus, walking to a restaurant, eating a meal, watching TV, etc. consists of many, usually hundreds, of individual SenseCam images. In many cases, where the wearer is moving around, a large range of dissimilar images are generated. The variety of SenseCam images in lifelogging introduces difficulties for event detection when compared to traditional TV news broadcasting video, for example. The image capture rate also makes dynamic descriptors, spatial-temporal features like HOG (Histograms of Oriented Gradients) and HOF (Histograms of Optical Flow) descriptors, inapplicable, which are well adopted in video classification [84, 80, 72].

5.3.1 Problem Description

In our research, the problem for lifelogging event topic detection is also simplified as a classification problem, that is, to find the most likely event topic from a lexicon set with regard to the event input.

Suppose we are given an annotated training set $\{(x^{(1)}, y^{(1)}), ..., (x^{(N)}, y^{(N)})\}$ consisting of N independent examples. Each of the examples $x^{(i)}$ represents the i-th event in the corpus. The corresponding annotation $y^{(i)} \in [1, |T|]$ is one of the topic lexicon T. The task for event topic detection can be described as: given the training set, to learn a function $h : \mathcal{X} \mapsto \mathcal{Y}$ so that $h(x)$ is a predictor with an unlabeled event input x for the corresponding value of y.

Going through the concept detection procedure, each image is assigned labels indicating if specific concepts exist in the image or not. Still, if we have the universe of concept detector set C, event $x^{(i)}$ is represented by successive images $I^{(i)} = \{Im_1^{(i)}, Im_2^{(i)}...Im_m^{(i)}\}$. The concept detection result for image $Im_j^{(i)}$ can be represented as an n-dimensional concept vector, as $C_j^{(i)} = (c_{j1}^{(i)}, c_{j2}^{(i)}...c_{jn}^{(i)})^T$, where n is equal to the cardinality of C and $c_{jk}^{(i)} = 1$ if concept k is detected in the image, otherwise $c_{jk}^{(i)} = 0$.

While the SenseCam wearer is performing an activity which requires him/her to be moving around, his view may be changing over time, though not, for example, if he/she is watching TV or working in an office looking at a computer. We need to map time-varying concept patterns into different activities. We find that to classify an event consisting of a series of images in temporal order is very similar to recognizing a phoneme in an acoustic stream, to some extent. The event is analogous to a phoneme in the stream and every image within this event is analogous to an acoustic frame. Then the task of temporal activity classification is suitable to be addressed by a classical Hidden Markov Model (HMM) [134], which has been proved to be efficient in the speech recognition application. Due to the characteristics of different events, event lengths vary significantly making the classification difficult. For example, 'using computer' might contain hundreds of SenseCam images while 'using phone' might only have several images representing it when having a short conversation. HMM can adapt to various lengths of event streams and avoid the effort of dynamic time warping to account for variations in length. We now elaborate the construction of HMMs for the solution of the above formalized event topic classification problem.

5.3.2 Vocabulary Construction for SenseCam Images

Concept detection provides us with an efficient way to decide on the appearance of concepts in images, which can be used as high-level semantic features for later concept-based retrieval or even further statistical classification. As we can see from previous sections, concepts play different roles in representing event semantics, and some of them interact with each other through their ontological relationship. This means the dimensions in a concept vector $C_j^{(i)}$ are not independent and some of the concepts are still similar to each other in meaning. Ignoring concept relationships will likely degrade the performance of later activity classification.

We deal with the underlying semantic structure using Latent Semantic Analysis (LSA) [48] in our research. As in the traditional Vector Space Model, LSA also represents terms and documents by vectors and analyzes the document relationship in terms of the angle between two vectors. The advantage of LSA is that the terms and documents are projected to a potential concept space and retrieval performance is improved by getting rid of "noise" in the original space [48]. In LSA, the similarity of meaning of terms is determined by a set of mutual constraints provided by term contexts in which a given term does and does not appear [93]. The application of LSA in our research can be described as the following:

Assume that we have n concept detector and a corpus consisting m SenseCam images. We can construct an $n \times m$ concept-image matrix:

$$\mathbf{X} = \begin{pmatrix} x_{11} & x_{12} & \dots & x_{1n} \\ x_{21} & x_{22} & \dots & x_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{m2} & \dots & x_{nm} \end{pmatrix} \tag{5.7}$$

where each element $x_{ij} = 1$ if concept c_i appears in image I_j, otherwise $x_{ij} = 0$. In matrix $\mathbf{X}$, each row represents for a unique concept and each column stands for an image.

The LSA is carried out by applying Singular Value Decomposition (SVD) to the matrix. The concept-image matrix is decomposed into the product of three matrices as shown:

$$\mathbf{X} = \mathbf{U\Sigma V}^T \tag{5.8}$$

where $\mathbf{U}$ and $\mathbf{V}$ are left and right singular vectors respectively, while $\mathbf{\Sigma}$ is the diagonal singular matrix of scaling values. Both $\mathbf{U}$ and $\mathbf{V}$ have have orthogonal columns and describe the original row entities (concepts) and column entities (images) separately. By SVD, the matrix X can be reconstructed approximately by less dimensions $k < n$ in the least squares manner. This can be simply done by choosing the first k largest singular values in $\mathbf{\Sigma}$ and corresponding orthogonal columns in $\mathbf{U}$ and $\mathbf{V}$. This yields the approximation as:

$$\mathbf{X} \approx \hat{\mathbf{X}} = \mathbf{U_k \Sigma_k V_k^T} \tag{5.9}$$

The reduced matrix not only retains the semantic relationship between original concepts and images, but also removes “noise” induced by similar concepts. Since $\mathbf{U_k}$ is an orthogonal matrix, it is not hard to calculate the projection of any sample vector C_j in the new concept space as:

$$\hat{C_j} = \mathbf{\Sigma_k^{-1} U_k^T} C_j \tag{5.10}$$

After the concept vectors are mapped to the new concept space, vector quantization is employed to represent similar vectors with the same index. This is performed by dividing the large set of vectors into groups having a number of points similar to each other. In this way, the sample vectors characterizing concept occurrences are modeled only by a group of discrete states which is referred to as *vocabulary.* Vector quantization is done by clustering sample sets in an n-dimensional space, to M clusters, where n is the number of space bases (k after LSA), while M is the vocabulary size.

For vector quantization, we applied a k-means clustering algorithm to categorize the samples in the k-dimensional space. To avoid local optimization of quantization error, we carried out 10 iterations of k-means clustering with different randomly initialized cluster centers. The clustering result with minimum square error is selected as the final vocabulary. One example of vocabulary construction is shown in Figure 5.8, in which sample points are projected in a $2-d$ concept space and clustered for a vocabulary of size 5.

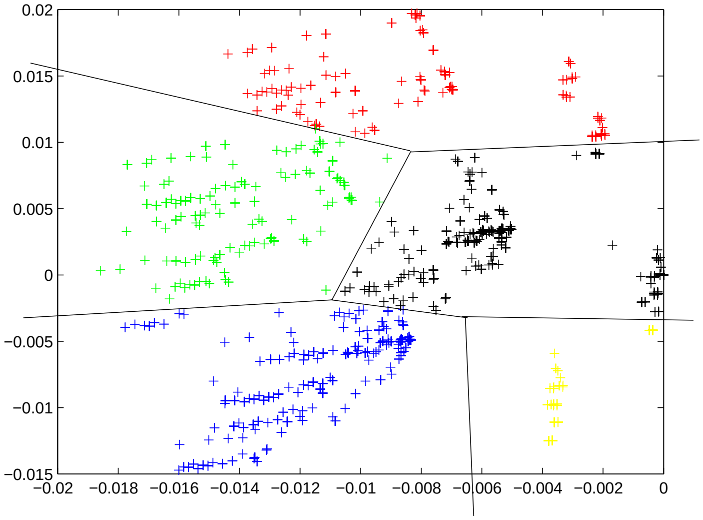


Figure 5.8: Vocabulary construction example in $2D$ space.

5.3.3 Concept-Spatial HMM Activity Classification

5.3.3.1 HMM Model Structure

In our activity detection, each lifelog event/activity segmentation is treated as an instance of an underlying activity type and is constructed by a series of SenseCam images. A Hidden Markov Model [134] is a very efficient machine learning tool to model time-varying patterns. In our activity classification the HMM treats the event instance as mutually independent sets of concepts generated by a latent state in a time series. The model structure as shown in Figure 5.9 is used in modeling the temporal pattern of dynamic concept appearances in an activity.

In Figure 5.9, one ‘Cooking’ event is demonstrated by the change of states and observation sequences, through the time line. The fully connected state transition model is shown in Figure 5.10:

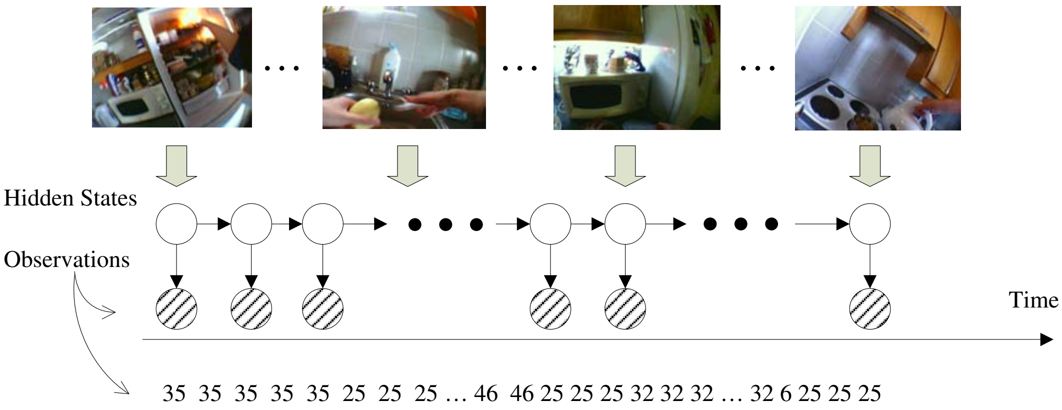


Figure 5.9: HMM structure for activity modeling.

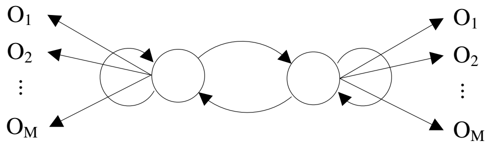


Figure 5.10: Two states transition model.

5.3.3.2 Parameter Training

The choice of k and M which determines the amount of dimension reduction in concept space and vocabulary size will affect the performance of our algorithm. The choice of k should be large enough to reflect the real structure in a new concept space, meanwhile should be small enough to avoid sampling errors or unimportant details introduced in the original matrix. It is a similar case in selecting a proper value for M for which the representation of observation and modeling complexity should also be balanced. Finding proper choices of k and cluster number M in a theoretical way is beyond the scope of our work and is an open issue in the information retrieval and machine learning community. In our work, we regard k and M as two parameters and test the best combination with the criterion of retrieval performance namely mean average precision (MAP).

As an evaluation matric, MAP is often used as a reflection of query performance in video retrieval. Average precision (AP) is defined as $AP = \frac{1}{min(R,k)} \sum_{j=1}^{k} \frac{R_j}{j} I_j$, where R is the number of relevant segmentations for a specific event topic, R_j is the number of relevant segmentations in the Top-j ranked results. $I_j = 1$ if the video shot ranked at $j - th$ position is relevant, and $I_j = 0$ otherwise. MAP is the mean AP of all event topics for a query.

We trained an HMM model for each activity class, that is, for each activity type we train the model with multiple observation sequences and find the optimal parameters. This is done using the Baum-Welch algorithm which optimally estimates the probability of the HMM model by iteratively re-estimating the model parameters. In our experiment we cross-validated the HMM models on training data with *leave-one-out* cross validation. After a specific number of iterations, the best initialized HMM parameters are selected and the HMM model is trained on all training data sets for the activity type. The models of different activity types are then evaluated on the final testing data to assess retrieval performance. The detailed model training and parameter searching will be presented in the experiment evaluation section, Section 5.5.1.

5.4 Semantic Representation: a VSM-Like Paradigm

The large amount of multimedia data collected in lifelogging poses severe difficulties in retrieval and representation for long-term lifelogs. The commonly accepted approaches for keyframe selection are based on analyzing low-level features. However, this way of representation selection often fails in properly reflecting higher level event semantics. The semantic gap between low-level features and event semantics needs also be bridged to achieve an event-centric representation. In the algorithm described below, we employed high level features as the measurement rather than the low-level features aiming to select the keyframe which is most relevant to the event semantics. This mechanism of event representation is proposed by leveraging concepts detected from image classification and has been proven to have the advantage of being informative and is of higher visual quality, as shown later in Section 5.5.3.

As a widely used search model, the Vector Space Model (VSM) [14] is known as one of the most popular models in information retrieval. In VSM, all entities including documents, queries and terms, are represented as vectors [150]. Using term vectors as the basis in vector space, both document and query vectors are built as linear combinations of the term vectors. The evaluation is then done by analyzing the correlation between the vectors as the relationship between query and document. In this section, we employ the VSM model as the representation for events.

Following the algorithm in section 5.2.2.2, event semantics is represented in the form of high-level features by a concept vector within which the concepts are ranked according to uniqueness. Assuming that event $e = s_1, s_2...s_N$ has the concept vector C_e, each image s_i has concept vector C_i. Both C_e and C_i are ranked in terms of the methodology in Section 5.2.2.2. Then the keyframe is chosen as satisfying:

$$s^* = argmax_{s_i \ \in \ e, \quad 1 \leq \ i \ \leq \ N} \ sim(C_i, C_e) \tag{5.11}$$

where $sim(C_i, C_e)$ is defined in Equation 5.6. The matrix calculates semantic similarities for each image with the event concept vector and then the most semantically similar image is selected as the keyframe. The advantages of this approach are described as follows:

- Semantically representative. The image is selected to be the most similar to the event semantics, so it must best represent the meaning of the event.

- Informative. The concept vector is ranked in terms of concept uniqueness. The concept which is more specific under the event topic is ranked higher. Then the keyframe must contain the most relevant concepts with the event topic.

- High visual quality. All the concepts are detected directly from the image by classifiers. The confidences for concepts detected from poor visual quality images are low. The keyframe with more concepts detected must thus have good quality.

- Wider visual field. Since SenseCam is worn around the neck by the user while collecting the data, the lens is often blocked by clothes or even arms; this will cause images with a narrower visual field. The semantically selected image will decrease the risk of choosing images that are partially blocked.

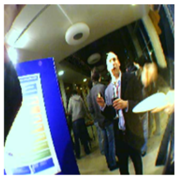
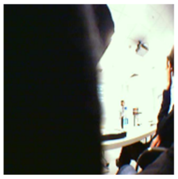
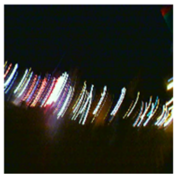
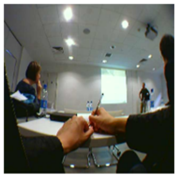
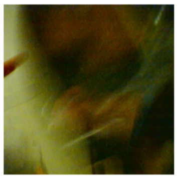
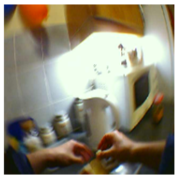
ID	Keyframe (LLF)	Keyframe (HLF)	ID	Keyframe (LLF)	Keyframe (HLF)
1			4		
2			5		
3			6		

Figure 5.11: Semantic representation for events.

To illustrate the advantages of this approach, Figure 5.11 demonstrates examples from which the keyframes using low-level features (LLFs) employed in [51] and high-level features (HLFs) are compared. Six events are randomly selected from one day based on the automatic segmentation of events [53]. The representations selected by high-level features have obviously better image quality than the ones selected based on low-level features, especially for events 1, 5 and 6. Objects are hardly recognizable in the LLF representation for event 1 and 6 due to motion blur. The images with higher quality often have more detail and concept information, so they are naturally selected as better representations using HLFs. In events 2, 3 and 4, the HLF representations are better than the LLF ones because of wider visual fields. Even during darkness, the HLF selection approach will choose images with more detail and better quality as shown for event 5.

5.5 Experiment and Evaluation

This section will describe the detailed experiments for the algorithms we proposed in this chapter, which are high-level activity classification, ontology-based multi-concept detection and semantic keyframe selection. All of these algorithms are aiming to deal with the challenges in indexing or representing the lifelogged data at event level, by leveraging the concepts detected at image level. The evaluations are carried out separately to test their performances.

5.5.1 Activity Classification Evaluation

5.5.1.1 Evaluation Data set

In the activity classification evaluation experiment, we carried out the assessment of our algorithm on data sets using both clean concept annotation and erroneous concept annotation. The data sets we used in our experiment are event samples of the 23 activity types we investigated in Chapter 4. Due to the limited number of positive samples of each activity type, we use the first 50% of each sample as training sample and another 50% as testing sample. The event types with more than 5 positive samples are selected to evaluate our algorithm. This leads to 16 event types which are shown in Table 5.1 with sample number and number of images contained.

Table 5.1: Experimental data set for activity classification

Activity type	Eating	Drinking	Cooking	Clean/Tidy/Wash
Sample number	28	15	9	21
Image number	1484	188	619	411
Activity type	Watch TV	Child care	Food shopping	General shopping
Sample number	11	19	13	7
Image number	285	846	633	359
Activity type	Reading	Driving	Use phone	Taking bus
Sample number	22	20	12	9
Image number	835	1047	393	526
Activity type	Walking	Presentation (listen)	Use computer	Talking
Sample number	19	11	17	17
Image number	672	644	851	704

5.5.1.2 Evaluation on Clean Concept Annotation

The clean concept annotation means the concept annotations on each image for event samples are error-free. This is done by manually annotating the 85 concepts we proposed in Chapter 4 for the data sets. For annotation purposes, a concept annotation software tool was developed for users to inspect the SenseCam images and judge if the concept exists or not. The temporal relationship is kept during annotation by providing a series of SenseCam images within the same event. This helps to improve annotation speed for the user by selecting positive image samples and the unselected samples will be annotated as negative samples. Thus a group of images can be annotated in one click and the whole event can be annotated in several clicks for one concept annotation. The performance of activity classification on clean annotation is now described.

As we described in Section 5.3.3, the selection of parameters k and M will affect the performance of our algorithm. In our experiment, we evaluated the final retrieval performance with different settings of these parameters. The search graph of parameters k and M in order to tune MAP is shown in Figure 5.12, for which 3 states HMM model is used.

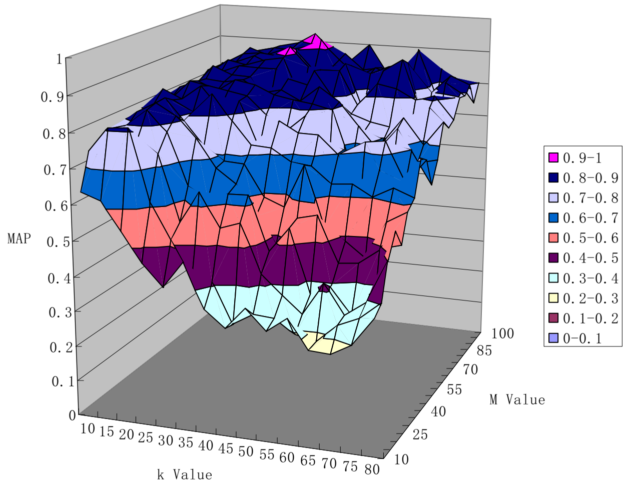


Figure 5.12: Search graph for MAP optimizing (3 states).

The search graph is built by varying k and M in the ranges [10..80] and [10..100] respectively. The best performances ($MAP \geq 0.9$) appear in the range [30..50] and [80..100] for k and M. When the value of k is increased, the value of M also needs to be increased to achieve better performance. The worst case happens when selecting a large k value and small M value, when more 'noise' is introduced from the concept space and the vocabulary clusters can not adapt to the 'noise'. The situation is better when k is low enough, say, $k = 20$, for which most choices of M have MAP above 0.8. Meanwhile, large M values can also complement the choice of k, when M is large enough ($M \geq 90$), most MAP remain at a satisfactory level, even though the best cases are in the range $k \in [30..50]$. A similar pattern can be seen when choosing different state numbers, e.g, two states as demonstrated in Figure 5.13.

In our training experiment, we trained and tested different settings of model parameters including the dimensions of concept space and the vocabulary size. After

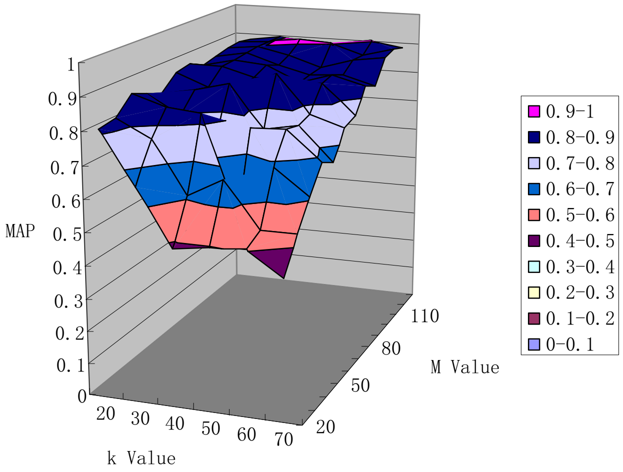


Figure 5.13: Search graph for MAP optimizing (2 states).

testing different combinations, we selected a concept space dimension of 35 and vocabulary size of 80 for further investigation, based on their performance. Different numbers of hidden states are tried in 5 runs (shown in Table 5.2) and the overall performance (average MAP) is considered in choosing the state number.

From Table 5.2, we find that 2 states achieves best overall performance which is then used to train HMM models for each type of activity. Because each HMM model can return the likelihood of an observation sequence, we perform activity classification by selecting the class of activity with highest likelihood for the input observation. The performance is then evaluated by precision and recall as shown in Table 5.3.

Among all these 16 activities investigated, the 'Driving', 'Food Shopping', 'Presentation (listen)' and 'Using computer' have the highest accuracy with both precision and recall being 1.00. Other activities like 'Reading', 'Taking bus', 'Using phone', 'Walking' and 'Watching TV' have an F-Score above 0.90. From the statistics reflected by Table 5.3, we find that the highest performances are achieved for activities in which the visual similarity of SenseCam images are high. The stability of concepts decided by image visual features makes it easier to detect these activities. As to the activities involving higher concept diversity, such as 'Child care', 'Cooking', 'Talk-

Table 5.2: State number searching (by MAP)

Number	Run1	Run2	Run3	Run4	Run5	Average
2	0.87118	0.89034	0.86201	**0.89533**	**0.87118**	0.**87801**
3	0.87089	**0.89352**	0.85792	0.89515	0.87089	0.87767
4	0.86629	0.88138	0.86006	0.89388	0.86629	0.87358
5	0.86151	0.87677	0.85807	0.89053	0.86151	0.86968
6	0.86989	0.88572	0.84547	0.88051	0.86989	0.87030
7	0.8548	0.88076	0.86071	0.88734	0.8548	0.86768
8	0.86948	0.87188	0.8545	0.8912	0.86948	0.87131
9	0.8482	0.87891	**0.86809**	0.88786	0.8482	0.86625
10	0.86023	0.87016	0.8559	0.87837	0.86023	0.86498
11	0.86665	0.87512	0.85716	0.87958	0.86802	0.86931
12	0.86299	0.8794	0.84177	0.87469	0.85733	0.86324
13	0.86538	0.87035	0.84766	0.88628	0.8625	0.86643
14	**0.87407**	0.87917	0.84238	0.87907	0.86712	0.86836
15	0.87063	0.87146	0.84914	0.87404	0.86087	0.86523
16	0.85593	0.86404	0.84667	0.89099	0.85285	0.86210
17	0.86959	0.87397	0.84208	0.89009	0.86399	0.86794
18	0.86947	0.86314	0.83931	0.88093	0.85556	0.86168
19	0.86617	0.86955	0.84656	0.87879	0.85313	0.86284
20	0.85914	0.86739	0.83312	0.87437	0.85962	0.85873

ing', etc., the overall accuracies are degraded but still remain at an acceptable level. Only 'Talking' and 'Drinking' has an F-Score lower than 0.80. Note that similar concept dynamics also introduces more misclassifications for activities like 'Drinking' and 'Eating'. In this evaluation, 1 out of 15 'Drinking' samples are detected as 'Eating' while 3 out of 28 'Eating' samples are classified as 'Drinking' activities. From Table 5.3, we notice that 'Talking' has the lowest recall 0.65. This is because 6 of these 17 'Talking' instances are misclassified as 'Drinking' (1 instance), 'General shopping' (1 instance), 'Walking' (3 instances) and 'Child care' (1 instance), due to very similar concepts like 'Face', 'Hand gesture', etc., which are the cues of 'Talking', but also frequently appear in other activities. These examples also show the influence of mapping ambiguity between activities and concepts on the final performance of activity classification. Even though in Table 5.3 the results are obtained based on clean concept detection without errors, the activity detection accuracies are still not

Table 5.3: Event detection results

Event type	Precision	Recall	F-Score
Child care	0.68	1.00	0.81
Clean/Tidy/Wash	0.86	0.86	0.86
Cooking	0.80	0.89	0.84
Drinking	0.75	0.80	0.77
Driving	1.00	1.00	1.00
Eating	0.95	0.75	0.84
Food shopping	1.00	1.00	1.00
General shopping	0.86	0.86	0.86
Presentation (listen)	1.00	1.00	1.00
Reading	1.00	0.95	0.98
Taking bus	1.00	0.89	0.95
Talking	0.85	0.65	0.73
Use computer	1.00	1.00	1.00
Use phone	0.92	1.00	0.96
Walking	0.86	0.95	0.90
Watch TV	1.00	0.82	0.90

perfect. This is because the mapping ambiguity from concepts to activities still exists especially for activities like ‘Clean/Tidy/Wash’, ‘Cooking’, ‘Drinking’, ‘Eating’, etc. When the detections of specific concepts are exactly the cues for activities, the detection performances of these activities are high. For example, the detection of ‘steering wheel’ has less uncertainty for ‘Driving’, the accuracy of ‘Driving’ is high based on the detection of ‘steering wheel’ and other concepts.

As described in Section 5.5.1.1, each event sample is divided into two halves, of which the first half is used as training data and the other is used as testing data. To evaluate the effect of this sampling method for training data and testing data, we also carried out the experiment on another sampling method which we call odd-and-even sampling to distinguish from half-and-half sampling. That is, in each event sample, we will use the images with odd number as training data while the images with even number are used as testing data. The performance comparison of the two sampling methods on the clean data set is shown in Figure 5.14:

For evaluation purposes, the training and testing are carried out for 10 runs with

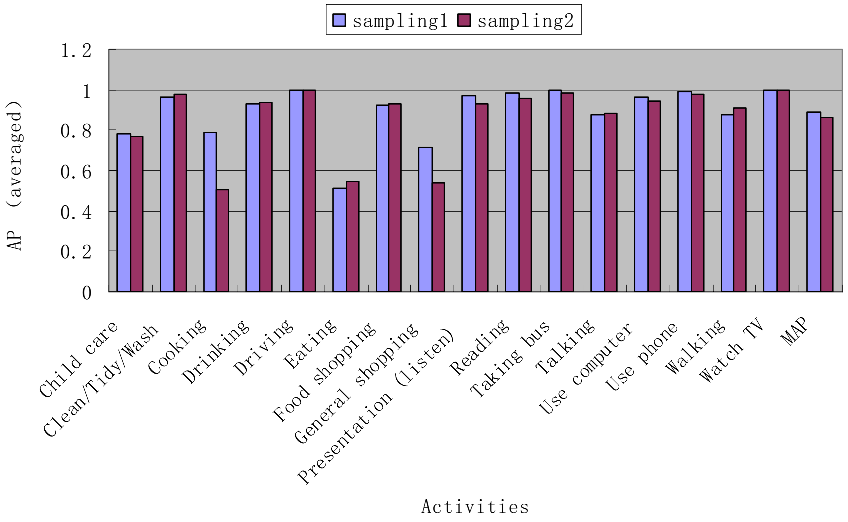


Figure 5.14: Comparison of two sampling methods (clean data).

each of the two sampling approaches. During the procedure, we used the same parameter settings as above, $k = 35$, $M = 80$, and 2 hidden states. The activity detection AP is calculated for each activity and then averaged on these 10 runs. The two sampling approaches are compared on the activity basis out of the 16 activities investigated. In Figure 5.14, the averaged AP for each activity and averaged MAP are shown. The half-and-half sampling and odd-and-even sampling are represented as $sampling1$ and $sampling2$ respectively in the figure. From Figure 5.14, there is no obvious difference between two sampling methods for most activities, compared on AP. Only two activities show obvious performance differences, which are 'Cooking' and 'General shopping'. The drop in performance for odd-and-even sampling shows that this sampling method can disrupt the intrinsic observation transition, especially for activities in which the observation of concepts changes frequently like 'Cooking'. For those activities in which concepts do not change so significantly, the performances of two sampling methods are almost the same, like 'Driving', 'Taking bus', 'Watch TV', etc. The overall performance is also dropped using odd-and-even sampling reflected by averaged MAP. The averaged MAP is 0.89 for $sampling1$ while it drops to 0.86 for *sampling*2. The performance difference shows that concept observation patterns can be changed by the odd-and-even sampling method. On the other hand, this also reflects that our algorithm can capture the pattern of concept dynamics and apply these patterns in activity classification for better performance. The evaluation of two sampling methods on erroneous concept detection will now be described.

5.5.1.3 Erroneous Concept Annotation

In order to assess the performance of our activity detection algorithm on automatically detected concepts which will have some errors in their detection, we carried out the evaluation by manually controlling the simulated concept detection accuracy, based on the groundtruth annotation. The simulation procedure is borrowed from [10], in which Aly *et al.* use Monte Carlo simulations to generate various accuracy performances for concept detection.

The notion of this simulation is based on the approximation of confidence score outputs from concept detectors as a probabilistic model of two Gaussians. In other words, both the densities for the positive and negative classes of a concept are simulated as Gaussian distributions. The concept detector performance is then controlled by modifying the models' parameters [10]. The method also assumes that the confidence scores of different detectors for a single object such as a shot are independent from each other. All concepts are also assumed to share the same mean μ_1 and standard deviation σ_1 for the positive class while the mean μ_0 and the standard deviation σ_0 are for the negative class. Then the performance of concept detection is affected by the intersection of the areas under the two probability density curves whose shapes can be controlled by changing the means or the standard deviations of the two classes for a single concept detector.

The implementation of the simulation involves the following processes. First, we simulate the confidence observations of concept detector as $N(\mu_0, \sigma_0)$ and $N(\mu_1, \sigma_1)$

for negative class and positive class respectively. The prior probability $P(C)$ for a concept C can also be obtained from the annotated collection. Then the sigmoid posterior probability function with the form of Equation 5.12 is fit for the generation of a specified number of S training examples.

$$P(C|o) = \frac{1}{1 + exp(Ao + B)} \tag{5.12}$$

After parameters A and B are decided, the posterior probability of the concept is returned using the sigmoid function for each shot with a random confidence score o drawn from the corresponding normal distribution. A more detailed description of the simulation approach can be found in [10] and [9].

In setting up the concept detectors with errors in our experiment, we modified the concept detection performance with the simulation based on the groundtruth annotation described in Table 5.1, for which each image is annotated with the existence of all concepts. During the simulation procedure, we fix the two standard deviations and the mean of the negative class. The mean of the positive class is changed in the range of [0.5 .. 10.0] to adjust the intersection area within the two normal curves, thus changing the detection performance. For each setting of parameters, we executed 20 repeated runs to avoid random performance and the averaged concept MAP and averaged activity detection MAP are both calculated.

5.5.1.4 Evaluation on Erroneous Concept Annotation

The evaluation on erroneous concept annotation is carried out by training and testing the activity detection algorithm described in Section 5.3.3, on the simulated concept detections with different accuracy. We increased the mean of the positive class μ_1 for each concept in our lexicon from 0.5 to 10.0 with step 0.5. For each value of μ_1, we execute 20 simulation runs, and for each run the concept detection MAP is calculated.

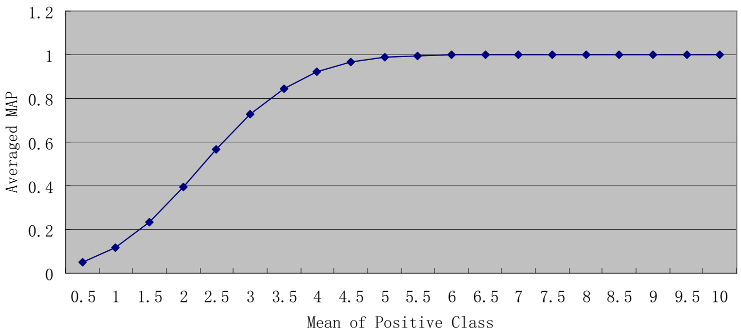


Figure 5.15: Averaged concept MAP with different positive class means.

In Figure 5.15, the concept MAP for all 20 runs are averaged and plotted with the increase of positive class mean μ_1. The x-axis shows the changes of μ_1 with the setting of the other parameters as $\sigma_0 = 1.0$, $\sigma_1 = 1.0$ and $\mu_0 = 0.0$. The y-axis depicts the value of averaged concept MAP for each μ_1. From Figure 5.15 we can see that an increasing of μ_1 achieves better concept detection performance. When μ_1 reaches the value 5.5, the concept detectors almost have the same performance with the ground truth and can be regarded as perfect.

For each run, the simulated concept annotations are analyzed by LSA first and projected to a new concept space with lower dimensions of $k = 35$. Vector quantization is then carried out in the new space by k-mean clustering and representing every SenseCam image with one observation from the vocabulary constructed. After vector quantization, the SenseCam image which was formerly represented with a 85-dimensional vector, is indexed with only the number of the cluster. In this step, we still choose $M = 80$ and achieve 80 clusters in the new concept space. The dynamic pattern of observations is modeled by the HMM model whose parameters are trained in the same process as described in Section 5.3.3. The testing is performed on the data set provided in Section 5.5.1.1.

Figure 5.16 depicts the changes of averaged activity detection MAP with respect to the μ_1 values, using the half-and-half sampling method for training and testing

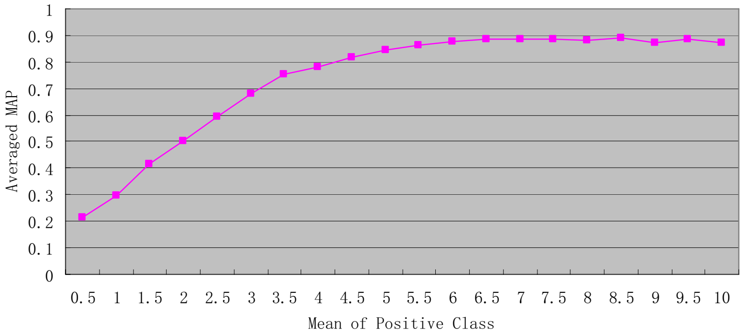


Figure 5.16: Averaged activity MAP with different positive class means.

data. The x-axis has the same meaning as it has in Figure 5.15 while the y-axis is the averaged MAP of activity detection over 20 runs. The activity detection performance increases with the improving of concept detection performance. Note that the activity detection performance does not drop significantly when the concept MAP is low. The smooth change of activity detection MAP shows that our algorithm is robust and tolerant to the errors introduced in automatic concept detection.

Similar to using clean concept annotation data, we also compared two sampling methods which are half-and-half sampling and odd-and-even sampling on simulated concept detectors. The concept detection simulation is performed by changing the mean of positive class μ_1 for each concept and 20 runs are carried out for each value of μ_1. For each simulation run, the evaluation procedure involved training and testing steps which are the same as using clean data and we use exactly the same parameter setting. Activity detection MAP is calculated in each run and then averaged on all 20 runs to obtain the overall performance on one simulation configuration. The performances of two sampling methods are compared and shown in Figure 5.17.

As shown in Figure 5.17, The x-axis shows the configurations of μ_1 varying from 0.5 to 10.0. The setting of the other parameters are the same as in Figure 5.15, that is $\sigma_0 = 1.0$, $\sigma_1 = 1.0$ and $\mu_0 = 0.0$. The averaged activity detection MAP over 20 runs is demonstrated in the y-axis. In Figure 5.17, two curves of $sampling1$ (half-and-

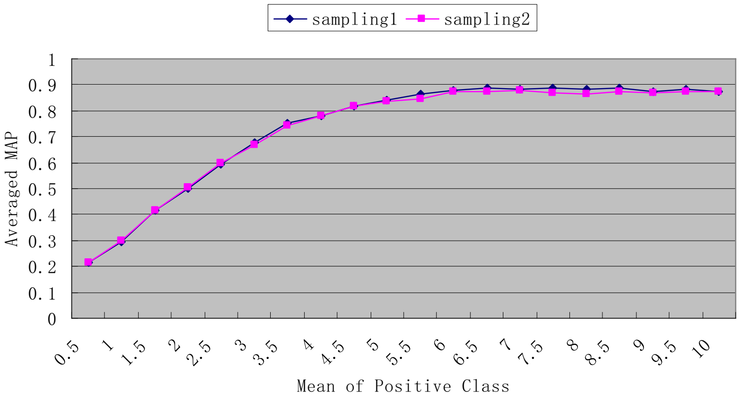


Figure 5.17: Comparison of two sampling methods (simulated data).

half) and *sampling*2 (odd-and-even) are the performances of two sampling methods. The overlap of two curves shows that there is no significant difference between two sampling methods, especially when $\mu_1 \leq 5.0$, for which the concept detection MAP is relatively low. While μ_1 increases, both of the performances of two samplings increase. When μ_1 is big enough, say, $\mu_1 \geq 6.5$, i.e. the concept detection MAP remains at a stable level (nearly perfect as shown in Figure 5.15), the curve of *sampling*1 remains higher than that of *sampling*2. This is consist with the comparison using clean data as described in Section 5.5.1.2. However, when the concept detection is not perfect ($\mu_1 \leq 5.0$), the erroneous concept appearance will change the underline concept observation patterns, therefore, two sampling approaches will perform at equal level. This can be depicted by the overlap of two curves when the value of μ_1 is small, as shown in Figure 5.17 especially when $0.5 \leq \mu_1 \leq 3.0$.

5.5.2 Ontology-based Multi-Concept Detection Evaluation

In Section 5.2.1, the motivation for using the semantic relationships among concepts in automatic detection has been discussed, together with our methodology for adjusting the detection confidence. The assessment of ontology-based multi-concept classification is discussed in this section. As we described in Algorithm 1, the whole procedure of the algorithm involves parameter training and confidence adjusting. In our evaluation, we randomly select one half from the corpus as training instances for the leaning of parameters A, B and C in Algorithm 1. Another half instances from the corpus are used for evaluation of ontology-based confidence adjustment.

To assess the effects of more comprehensive concepts on detection performance, we tested the distributions of accuracy and various multi-class margins. This is carried out by assigning only one disjoint concept for the target concept at one time. For evaluation purposes, we first obtained the ground truth of concepts 'Indoor' and 'Outdoor' for each image by means of user annotation, on a corpus consisting 10,226 SenseCam images. The baseline concept detection is performed by standard SVM classifiers as we describe in Section 5.5.3, which are also referred to as original one-per-class classifiers. Our ontology-based classification algorithm is then applied on the output of the baseline. The results of these two classification methods are both compared with ground truth annotations to calculate evaluation metrics like accuracy, AP and MAP whose definition have been given in Section 5.3.3.2.

Figure 5.18 shows the correlation of class-prediction accuracy on 'Indoor' concepts with the multi-class margin when a single disjoint concept is introduced. The disjoint concepts are modeled in concept ontology with the relationship of disjointness as described in Section 5.2.1. For example, 'Outdoor', 'Road', 'Sky', 'Vegetation', 'Tree' and 'Grass' are all typical disjoint concepts of 'Indoor'. These are the disjoint concepts which can be used in improving 'Indoor' detection and their confidences have different effects on the detection of 'Indoor', as shown in Figure 5.18. Among them, 'Outdoor' has the most significant influence on 'Indoor' accuracy while 'Grass' has the least effect. Even though they have influences on 'Indoor' to various degrees, they comply with the same distribution and can be fit by the form of Equation 5.12. The multi-class margin calculated by Equation 5.1 takes into account the effects from all of these disjoint concepts and applies them to adjust 'Indoor' detection confidence.

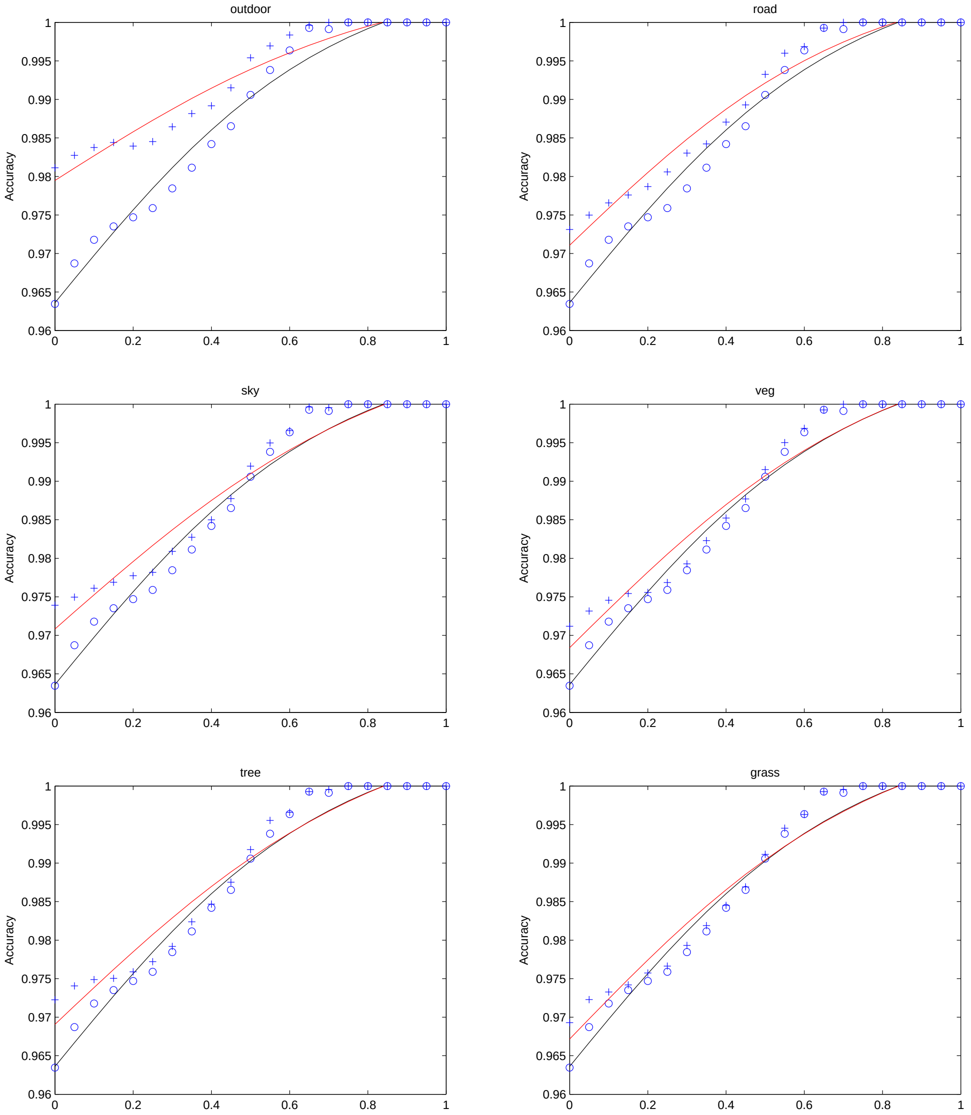


Figure 5.18: Accuracy improvement for 'Indoor' concept by single concepts.

Similarly, Figure 5.19 depicts the precision-recall curves for 'Outdoor' detection before and after applying the concept ontology. As we can see, the area under the precision-recall curve has been obviously increased by adjusted confidence, shown by the solid red line. The curve of the original confidence (show as blue dash-dot line) is located much lower under the red curve in the left part of the figure, when recall is less than 0.5. For example, when precision is at a high level of 0.7 the recall value is 0.35 for the curve of adjusted confidence, which is much higher than the baseline value 0.1. At a high level of precision in Figure 5.19, even when recall increases, the precision decreases much slower for adjusted confidence than using the original confidence as a criterion.

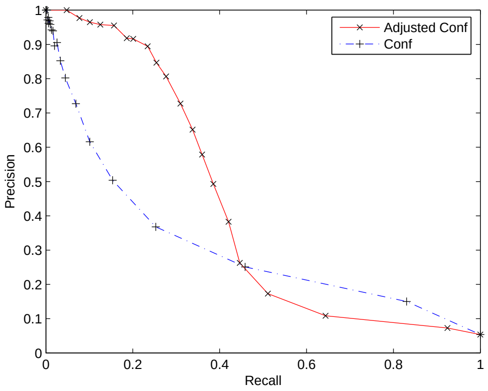


Figure 5.19: Precision-recall curve ('Outdoor').

A similar improvement of classification precision and recall can also be seen in Figure 5.20 for 'Outdoor'. In Figure 5.20, the x-axis depicts the values of original and adjusted concept detection confidence. The y-axis stands for the metric values of precision and recall. In Figure 5.20, we use two blue lines to represent precision and recall curves for adjusted confidence, while the black dotted line is the recall curve for original confidence. With the confidence values (including original and adjusted) increase, the classification precision becomes higher while the recall decreases. The performance of adjusted confidence is much higher than the baseline, reflected by two recall curves. After correcting the concept detection confidence by inherent ontological relationships, the precision of concept detection also remains satisfactory as shown by Figure 5.20. When the threshold of adjusted confidence values is larger than 0.5, the precision of 'Outdoor' remains above 0.8.

Another group of evaluations was carried out for a large number of target concepts, using the data set we simulated in Section 5.5.1.3. In this data set, the concept detectors for 85 concepts at different accuracy levels are simulated by changing the

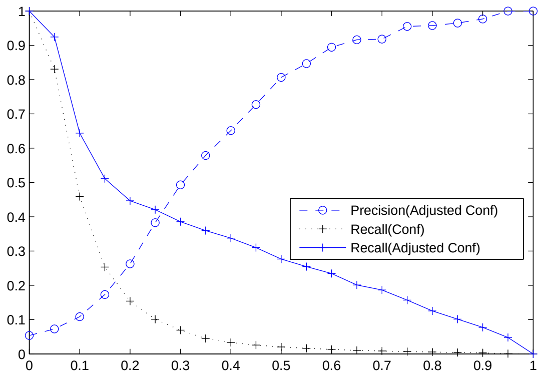


Figure 5.20: Effect of adjusted confidence ('Outdoor').

mean of positive class μ_1 for each concept classifier. The posterior probability of concept existence is returned as the simulated concept detection output and we use this value as the original classifier confidence. The purpose of this evaluation is to learn to adjust this simulated confidence value by employing our ontology model and then assess the final performance. The ontology we used in this evaluation for 85 concepts is demonstrated in Figure 5.3 and Figure 5.4, in Section 5.2.1. Two semantic relationships are modeled in this ontology which are subsumption and disjointness. Both of them are utilized to improve multi-concept detection accuracy.

For any configuration of μ_1, our evaluation is carried out with training and testing components for each run. In a single run, each concept is selected and the corresponding classification output is adjusted by considering the constructed ontology structure. After the parameter learning and confidence adjusting are finished, we calculate the AP for each concept. This is repeated for 5 runs on 5 unique data sets simulated in Section 5.5.1.3. The AP values are averaged over all 5 runs to obtain an evaluation from an overall point of view. After ontology inference, 52 of 85 concepts have disjoint concepts and can be adjusted for improved confidence. Out of these 52 concepts, 35 concepts have sample numbers larger than 100 in the corpus. Because enough samples are necessary to learn the parameters for confidence adjustment as shown in

Algorithm 1, we will then use these 35 concepts as the final evaluation concept sets.

In Figure 5.21, the improvement of averaged AP is depicted over all these 35 concepts. The positive y-axis value means that the performance is improved after employing concept semantics modeled in concept ontology while the negative value means the performance is degraded. As shown in Figure 5.21, most concept instances are upgraded with their corresponding detectors. Only 6 concepts have decreased detection performances after applying our algorithm. For concepts like 'Inside bus', 'Inside car', 'Road' and 'Path', the improvement reaches as high as more than 20%.

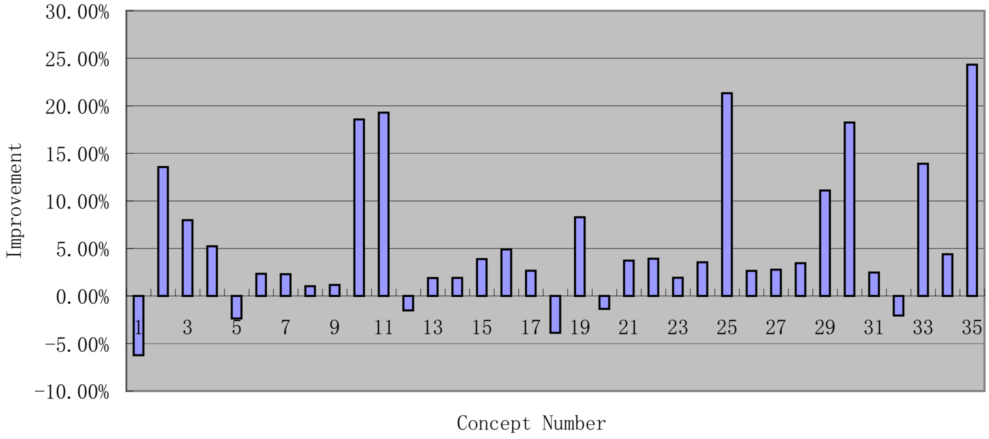


Figure 5.21: AP improvement by Disjoint Confidence Adjusting (DCA).

Beside concept AP, the MAP is also calculated for each run and then averaged. Figure 5.22 shows the MAP improvement when μ_1 is assigned different values varying from 0.5 to 10.0. Therefore, this figure also reflects the improvement at different performance levels of baseline concept detectors. When the baseline concept detectors have lower performance ($\mu_1 < 5$), all evaluations show obvious improvement by our algorithm. When $\mu_1 \geq 5$, i.e. the concept detection accuracy is already high (MAP nearly 1.0 as shown in Figure 5.15), there is definitely no space for the performance to be improved. The best improvement appears when μ_1 has the value of 1.5, at which the MAP improvement is higher than 6%. When μ_1 has smaller or large values, the improvement will become less obvious. This tells us that the ontology-based detection algorithm performs especially well when the performance of the original detectors is neither too good nor too poor. In the two extreme circumstances (very good or poor concept detectors), any added value of concept semantics will not be that significant.

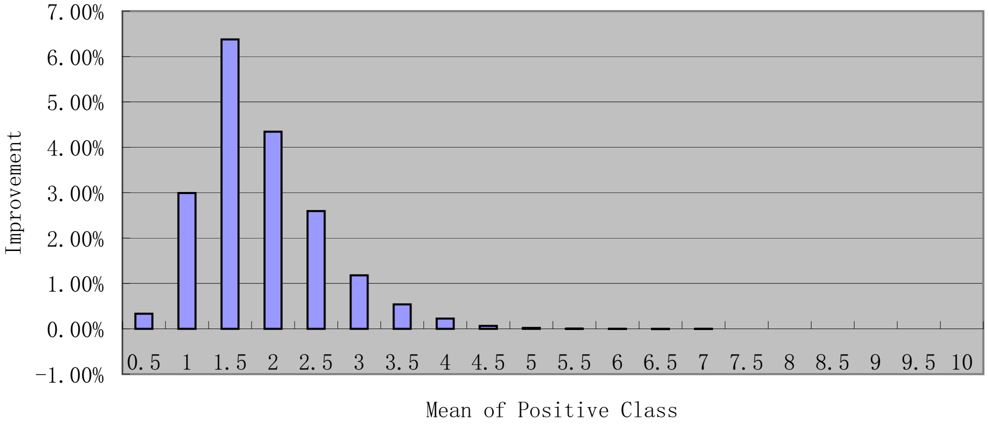


Figure 5.22: *MAP* improvement by Disjoint Confidence Adjusting (DCA).

5.5.3 Event Representation Evaluation

In order to evaluate the semantic selection of representing events, we built classifiers for a set of 27 concepts in constructing the concept space in our SenseCam-based event processing. The evaluation on these 27 concept detectors will not only simplify the assessment problem but also help to reuse the existing annotation efforts in our group since we already have a large corpus of annotation for these 27 concepts [34]. These 27 concepts are shown in Table 5.4 organized into categories of objects, scene/setting/site, people and events. Note that the the semantic keyframe selection algorithm is generic and can be extended to larger concept sets as well.

Following the state-of-the-art in concept detection, we employed the popular generic SVM learning algorithm for concept detection. Two MPEG-7 features were extracted for each image, Scalable Colour (12 bins) and Colour Layout (64 bins)

Table 5.4: SenseCam concept sets for keyframe evaluation

Objects	screen, steering wheel, car/bus/vehicles
Scene/Settings/Site	indoor, outdoor, office, toilet/bathroom, door, buildings, vegetation, road, sky, tree, grass, inside vehicle, view horizon, stair
People	face, people, hand
Event	reading, holding cup, holding phone, presentation, meeting, eating, shopping

forming 76-dimensional feature vectors. For the results presented in this paper, SVM-Light [81] was employed with the radial basis function (RBF) as a kernel, $K(\mathbf{a}, \mathbf{b}) = exp(-\gamma \|\mathbf{a} - \mathbf{b}\|^2)$. RBF kernel usually performs better than other kernels and has been employed in many state-of-the-art multimedia search engines [156, 159, 38] with it capability in learning nonlinear decision boundaries from a skewed high-dimensional space. The parameter settings were determined through iterative searching among parameter combinations. Classification models were trained for different concepts and work in isolation, yielding a 27-dimensional confidence vector for each image.

As mentioned earlier, SenseCam images have very different visual characteristics to the video keyframes used in the TRECVid benchmark [151, 152] and so we could not evaluate the performance of our concept detection on the TRECVid datasets. Thus an experiment was carried out on 6 participants' SenseCam image logs. The participants are all researchers in our lab and have been wearing SenseCam for varying lengths of time. The effect of interestingness-based semantic keyframe selection is compared with the baseline which is the selection of the middle image as a representation for an event, the same technique as is used for keyframe selection in video. Details of the data are shown in Table 5.5 indicating a total of 1,055 events composed of 96,217 individual images.

Concepts were first detected at the image level, followed by interestingness-based aggregation to model event semantics. We empirically choose the value $\xi = 200$ in Equation 5.4 considering the fact that most events have less than 200 images. Image-

Table 5.5: Experimental data set

	User1	User2	User3	User4	User5	User6
Events	300	248	242	168	70	27
Images	26,062	25,341	19,233	18,085	6,097	1,399

event semantic similarities are calculated to select the most similar image to the event semantics. In [51], a fusion of the Contrast and Saliency Measures in exploiting image quality show promising user judgement scores, which are no less satisfactory than more complicated fusions taking Colour Variance, Global Sharpness or Noise Measure into account. We employ the Contrast Measure and Saliency Measure from [51] as two measures to evaluate resulting keyframe quality. The Contrast and Saliency scores are calculated and normalized on a Max-Min scale respectively. To decrease the effect of external factors such as life patterns of individuals and characteristics of different SenseCam lenses, we analyze the results of our algorithm on a per-user basis.

Our semantic similarity measurement is tested on resulting Contrast and Saliency scores. Figure 5.23 shows the Contrast difference of selected keyframes by semantic similarity (SS) defined as Equation 5.6 and by Cosine similarity (COS) on one random user's dataset. The averaged Contrast scores over all event numbers are 0.477 and 0.459 using SS and COS measures respectively. From Figure 5.23, it is obvious that keyframes selected by the SS measure have better contrast quality. The same happens for the Saliency measure as shown in Figure 5.24, where averaged Saliency scores using the SS measure outperforms the COS measure by 15%. The semantic similarity also shows significant advantages over other measures like inner product, Euclidean and so on and we will not elaborate the details here because the comparisons are very much similar to those as described for Figure 5.23 and Figure 5.24.

In Figure 5.25, the improvement on average values of the Contrast and Saliency Measures with semantics-based representation are shown for each user. Both measurements are significantly enhanced over the baseline for all participants. Note that

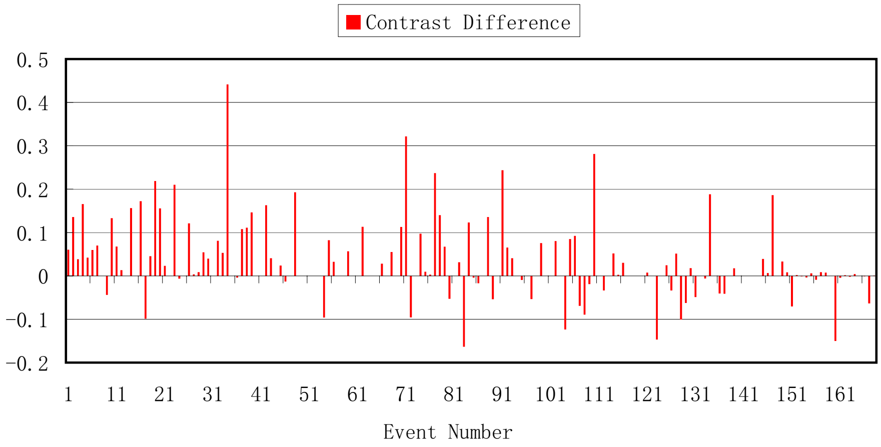


Figure 5.23: Contrast difference (SS-COS).

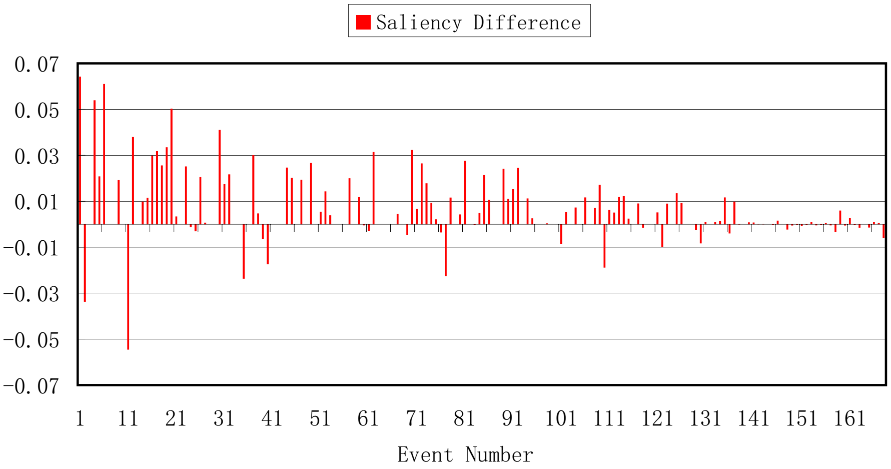


Figure 5.24: Saliency difference (SS-COS).

user5 is using an old SenseCam whose lens is scratched and the images blurred yet the semantics-based algorithm still performs well showing the robustness of our semantic modeling.

Modeling complexity is modified in our experiments by changing the selection of Top-k ranking of concept vectors to test the effect of event semantics on the selection of representative images. Figure 5.26 shows the dependence of keyframe quality on the semantics of events, by selecting the Top-k concepts. Results are depicted using an equally-weighted image quality value of Contrast (0.5) and Salience (0.5). For illustration, we randomly selected three participants' fused image quality scores and

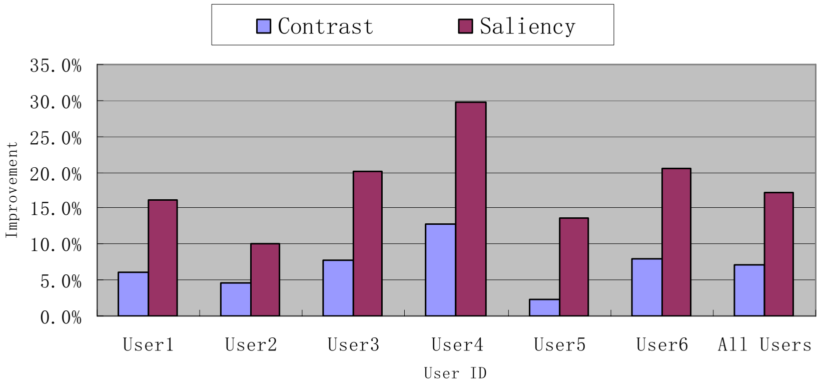


Figure 5.25: Contrast vs. Saliency Measure.

compared with their corresponding baseline values. With parameter k decreasing, the fused quality of semantics-based representation drops after k is less than 10. The correlation of quality score with choice of k demonstrates the impact of semantics of events on keyframe selection. When just a little semantics are employed, see $k \leq 2$, the quality score curves intersect with their own baselines, showing no obvious improvement. This also shows that our similarity measure is appropriate in deciding the relationship for concept-based semantics.

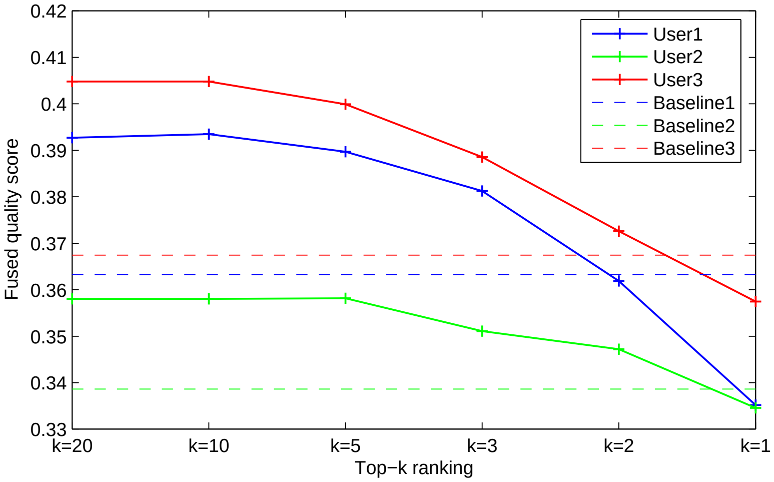


Figure 5.26: Correlation of quality with Top-k.

Figure 5.27 compares the number of concepts detected from each selected keyframe. When more concepts are used, e.g. $k = 20$ or 10, keyframes tend to contain more semantics about the events (nearly half have 3 or 4 concepts). Similar to image quality in Figure 5.26, the number of concepts in the representation decreases with smaller k values. Meanwhile, the representativeness of keyframes drops and less details about the represented events are found. When only the first concept is selected from the event concept vector, say $k = 1$, the semantics reflected in the semantics-based representation is almost the same as the baseline.

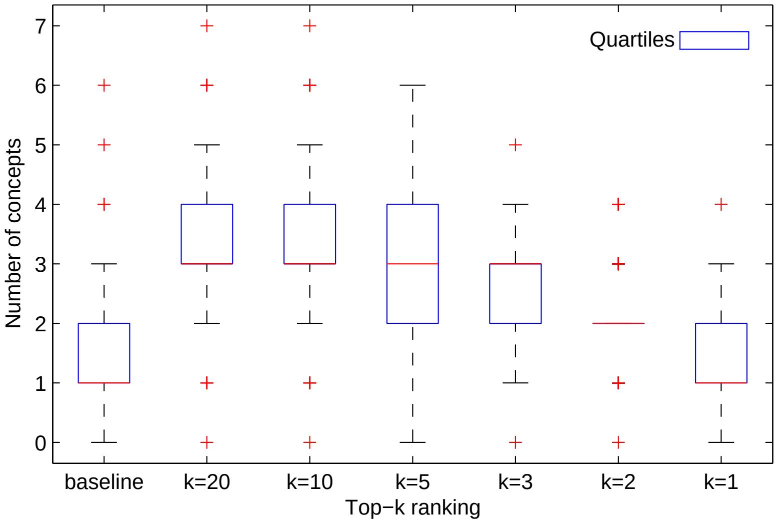


Figure 5.27: Concept number in single representation.

As demonstrated above, the image quality and potential concepts from the keyframe selected based on semantics shows strong correlation with the choice of k. When more semantic information is applied ($k \geq 5$), our algorithm performs well in selecting keyframes which are more representative and of better quality. Our interestingness-based event aggregation not only reflects semantics of events but also provides a computable platform in comparing semantic relationships such as similarity in the same concept space.

5.6 Summary

This chapter started with the discussion of event-level visual processing in lifelogging. Following the definition of a lifelogging event, the issues of adding semantics to lifelogged media at event level are proposed. To deal with the challenges of fusing image-level concepts, three algorithms are developed in this chapter. In ontology-based multi-concept classification, ontology modeling and inference are applied in order to incorporate concept relationships in multi-concept detection. Diverse concepts detected within events are then utilized for semantic event representation and high level activity detection. Inspired by traditional $tf \times IDF$ term weighting from the information retrieval field, an interestingness-based event level concept aggregation approach is proposed and applied in automatic keyframe selection from events. The semantically selected representation shows the advantages both in image quality and in semantic richness. An HMM-based activity detection algorithm is also proposed to recognize different activity types from the time-varying concept dynamics. The algorithm is evaluated on data sets with various performances of everyday concept detection and shown to be promising in indexing lifelogged visual media at the event level. From the output of these algorithms, we can find the added value based on concepts. As the high-level features, concepts can be incorporated and fused for more complex tasks like event classification and representation.

Chapter 6

Event Modeling and Semantic Enhancement

As defined in Chapter 5, an event is a transient occurrence of a happening of interest in the real-world. In lifelogging, this occurrence is observed and recorded within a computer system and there is much multimedia data collected for each individual event. One automatic way to help users to find an event of interest is keyword-based searching on labeled event indices. Since no consumer can afford the tedious effort of annotation for such a large amount of media data, in earlier chapters we investigated the selection of a metadata lexicon and automatic annotation of events by fusing semantic concept detection at event level which dealt with research questions (RQ1), (RQ2) and (RQ3). Even though this kind of image indexing based on visual information has shown to be effective to discriminate desired events from large volume of lifelogging archives, this single-dimensional semantic indexing fails in making full use of context information about events to provide more flexible measures. How to organize lifelogging events with multi-contextual metadata is an important issue for efficient event-centric retrieval and interpretation of lifelogs. Furthermore, keyframe-based event representation is the dominant means of multimedia representation. For lifelogging, when event captures are very rich in context information, a multi-dimensional contextual representation method is needed to enhance significant fractions of event aspects. The enhancement of lifelogging events/activities (RQ4) are the focus of this chapter.

By now, Semantic Web technologies have reached a sufficient level of maturity in terms of well-structured online knowledge repositories and semantic query/reasoning capabilities so that they can be used to enrich our understanding of daily events. In this chapter, we propose an event model based on a context-awareness application. We will address the issue of incorporating context semantics in one consistent event ontology model and we will perform semantic enrichment to make better sense of lifelogging events.

6.1 Semantic Representation and Model Language

The current WWW is an infrastructure for publishing arbitrary information online in the form of documents or web pages. This way of document publishing allows us to access digital resources beyond the physical or technical constraints. However, the document-based publishing platform fails to provide efficient content access facilities for newly-needed online services. The lack of standardized semantic description in documents and the limited meaning of document links make knowledge reuse very limited on the WWW as it is used currently. As a consequence, the current WWW is indeed a form of user-centric Web since its content can only be accurately interpreted by users rather than computers. Aside from WWW technologies which are document-driven, the Semantic Web [22] elaborates a data-driven infrastructure for data sharing and representation. Semantic Web technologies are evolving the ongoing Web into a more powerful and more reusable infrastructure for information sharing and knowledge management. By defining web data with meaningful information, the Semantic Web is more understandable and more reusable by machines than the current Web, making interoperation easier between software agents.

We believe that Semantic Web technologies and standards can facilitate the use of online information to interpret the semantics of lifelog events in the form of meaningful data rather than documents, based on the standardized information model in machine-readable languages to support data representation and inference. In this section, we will discuss the standard semantic representation and modeling language used in the Semantic Web. As we described in earlier sections, it is hard to represent the content of multimedia data directly due to the characteristics of multimedia data. Since multimedia data itself is difficult to organize for retrieval by precise matching, in modern information retrieval, descriptive metadata is extracted to model media content in a more structured way, hence decreasing the complexity of multimedia. The metadata is handled in multimedia retrieval systems together with media objects as a whole package. In another words, metadata is the structured semantics of multimedia content. When different persons have different understandings of the meaning of metadata, another question is how to make the applications interoperable. To address this issue, the standardized Semantic Web description languages, which define both syntactic representations and semantic contents, are needed to make metadata interoperable between applications.

6.1.1 Ontologies

An ontology is the core element of the Semantic Web adopted from philosophy and aiming to facilitate knowledge sharing and reuse between data consumers including Web users and machines. It is analogous to a database schema in a relational database or class diagram in object-oriented software engineering which are used to form an abstraction of domain knowledge. The difference is that, in the Semantic Web, an ontology is built up with concepts, relationships, and constraints, defined by statements. For the definition of an ontology, we quote the widely accepted definition [161] in the Semantic Web community as:

> "An ontology is a formal, explicit specification of a shared conceptualization."

As an abstract structure of domain knowledge, an ontology must be represented explicitly and concretely by formal logic-based models for machines to understand each other. This is done by using reserved vocabularies which are collections of predefined terms. The formal structures of ontologies are stored as documents on the Web consisting of the following fundamental components:

- **Classes:** A class is the abstraction of a set of resources that share common characteristics. For example, 'Event' can be a class representing the group of all events. By adding hierarchical relationships between classes, taxonomies are constructed in ontologies by specifying class subsumption. In the hierarchy, a class can subsume or be subsumed by other classes. A class subsumed by another is called a subclass, of the superclass which subsumes it. By linking two classes with a subsumption relationship, the properties of the superclass will be inherited by the subclass. For example, 'Car' is a subclass of 'Vehicle', so 'Car' has all the properties of 'Vehicle', like 'having motor engine', 'having four wheels', etc.

- **Individuals:** An individual is any resource that is a member of at least one class. Indeed, an individual is a concrete instance of class and can not be further specified. As the lowest level of abstraction in an ontology, instances are not necessarily to be included in ontologies. Individuals can be asserted to be members of classes explicitly in ontologies, though sometimes the membership can also be inferred indirectly from other assertions defined in ontologies.

- **Attributes:** An attribute is used to describe a resource (such as an instance or a class), by relating them to other instances, classes or data values. An attribute is also a resource that is used as a predicate in statements to describe subjects. In the Semantic Web, there are two main attribute types which are object properties and datatype properties. Just as the name implies, object properties link the subject described to other resources, and datatype properties link the subject to literal values.

Note that the statement forms the fundamental block of an ontology. A statement consists of a subject, predicate, and object which typically form a *triple*. The subject in a triple is the resource that is described by the statement, and the subject and the object are linked by the predicate to describe the relationship between them. For example, in the statement "Car is a subclass of vehicle", the subject "car" and the object ''vehicle" are connected by the predicate "a subclass of". This triple model naturally forms a directed graph, in which the subject and object in one statement are represented as nodes while the predicate is an edge starting from the subject and ending with the object. Though simple, the subject-predicate-object triple model achieves more flexible expressions by relating one statement to another, so that forms the web of data constituting the Semantic Web. Thousands, even billions of formal semantics on the Semantic Web are all aggregated by this triple model. The Semantic Web standard languages are actually the formalized syntax to assert the statements modeled by triples.

6.1.2 Resource Description Framework (RDF/RDFS)

The Resource Description Framework (RDF) [89] is the fundamental Semantic Web data model language formalizing semantics as statements. RDF original from the XML syntax and intended to represent metadata about Web resources, and was then developed as a language for expressing statements. Currently, RDF is usually referred to as a family of World-Wide-Web Consortium (W3C) specifications.

In RDF, the subject is also represented as resources. The asserted statements are modeled semantically by RDF about the identifiable resources. The resource description is actually arbitrary in RDF. Once the resource (an instance or a class) can be identifiable with its Uniform Resource Identifiers (URI), it can be represented by a semantic data model for asserting statements. The use of URIs allows us to identify a network-homed resource. Using the universal URI set of symbols, statements from different sources can be created to interlink, ultimately forming a graph of statements. There are various serialization formats for RDF such as RDF/XML [18], N3 [21], Turtle [19] and N-Triples [64].

The drawback with RDF's flexible modeling capability and expressiveness is that the meanings in RDF need to be specified by a vocabulary. RDF Schema (RDFS) is such a standard vocabulary that explicitly specifies the semantic of terms in RDF behind descriptions. RDF Schema (RDFS) provides a specific vocabulary for RDF that can be used to define taxonomies of classes and properties and simple domain and range specifications for properties. RDFS is itself expressed in RDF and is thus a member of the RDF specification family. RDF and the schema RDFS are used together to describe resources on the Web with concrete semantics. The combination has the the capability of providing vocabularies, taxonomies, and ontologies in the Semantic Web. Many RDF application will therefore reuse the metadata definitions by sharing RDF schemata.

As we mentioned above, the RDFS enables the definition of classes and properties. This is performed by denoting the resources with classes of `rdfs:Class` and `rdfs:Property`. Both `rdfs:Class` and `rdfs:Property` are subclasses of `rdfs:Resource` which is the most generic class denoting resources. So the classes and properties defined in any domain-specific schema will become instances of these two resources. The `rdf:type` property is used to classify the resources with classes or properties defined in a schema using RDFS. The definitions of subclass and sub-property hierarchies are enabled by `rdfs:subClassof` and `rdfs:subPropertyOf` properties offered by RDFS. More terms defined in RDFS for property domain (`rdfs:domain`) and range (`rdfs:range`) restrictions and other informal descriptions of classes and properties (`rdfs:comment`, `rdfs:label`, `rdfs:seeAlso`, etc.) can be found in [28].

6.1.3 OWL

OWL is the abbreviation for Web Ontology Language, a language for defining and instantiating Web ontologies. OWL provides an expressive language for defining ontologies that capture the semantics of domain knowledge. It extends the RDFS vocabulary with additional resources that can be used to build more expressive ontologies for the Web. Developed to augment the RDF and RDFS languages by additional vocabulary, OWL supports greater semantic interpretability of Web content. OWL is also syntactically expressed in RDF. As a vocabulary extension of RDF, OWL introduces extra restrictions aiming to make interpretation and inference more efficient with respect to the structure and contents of RDF documents. Complying with the OWL standard, ontology developers can take advantage of reasoning capabilities based on the classes and properties defined by OWL. Typical properties initiated and inferred by OWL are transitive properties, functional properties and inverse functional properties.

Aside from the complete OWL language (called OWL Full), OWL also provides two specific subsets for various needs by implementors and users. They are OWL Lite and OWL DL [114], which are described together with OWL Full as follows:

- **OWL Full:** OWL Full is the full set of OWL language. OWL Full allows free mixing of OWL with RDF Schema and, like RDF Schema, does not enforce a strict separation of classes, properties, individuals and data values [114]. The high flexibility of OWL Full scarifies its computational efficiency. It relaxes some of the constraints on OWL DL to make some useful features available, but violates the constraints of description logic reasoners.

- **OWL DL:** OWL DL contains the entire vocabulary of OWL Full, but as distinct from OWL Full, OWL DL puts constraints on the mixing with RDF and requires disjointness of classes, properties, individuals and data values [114]. The main reason for having the OWL DL sublanguage is that tool builders have developed powerful reasoning systems which support ontologies constrained by the restrictions required for OWL DL. These restrictions make OWL DL decidable and provide many of the capabilities of description logic which is an important subset of first-order logic. That is why this subset of OWL is named as OWL DL.

- **OWL Lite:** OWL Lite is a subset of OWL DL that supports only a basic set of the OWL language features. By providing limited expressivity, OWL Lite is particularly targeted to support the need of tool builders who want to start with a simple basic set of OWL language features.

6.2 Contextual Event Enhancement Architecture

In this section, we will elaborate our event enhancement architecture based on a multi-contextual event model. This notion of multi-contextual awareness is motivated by current needs for intelligent computing in lifelogging, for which a typical scenario is first illustrated.

6.2.1 An Illustrative Scenario

The large-scale proliferation of Web 2.0 and Semantic Web technologies provides a large amount of social media and machine-readable metadata which can be assimilated in interpreting event semantics in lifelogs. These kinds of online resources can be regarded as logical contexts compared to the situational contexts captured by sensor deployments like SenseCam. To provide a clear notion of how it is useful to encompass online resources, we consider a scenario in which a lifelogging user attends a conference presentation in Dublin City University (DCU).

Wearable devices record his locations and visual images. A mobile device would infer that he is currently in DCU according to the reasoning of spatial relationship with the university location. Image processing is also applied to detect that he is in a lecture room and sitting in front of a large projection. It is just lunch time when the lecture is over. The mobile device would search nearby restaurants and recommend the restaurant his friends often go to and some favorite dishes from them. After he orders from the menu, he can re-experience the conference events on his logging mobile device and he might think the lecture topic is very interesting and helpful to his current research so he would like to know more about the presenter. The name of the presenter is then interlinked to the online knowledge base about research expertise in his area and all the other papers published recently by the same presenter would be searched from the linked data base.

The above described scenario depicts a situation we often come across. However, current web applications can not realize this kind of customized service and integrate all these resources in the way described above. In addition to domain semantic modeling, well-structured knowledge bases and semantic query engines are also needed to adapt to multi-dimensional context-awareness.

6.2.2 Event Ontology Based on Multi-Context

Context information can be collected through the deployment of heterogeneous sensory devices in lifelogging. To model lifelogging events, context information should be contained in one consistent event model and this model should allow each context to be processed separately. That is because high uncertainty is often embedded in the context processing due to the occurrence of data loss (such as GPS signal dropouts) or detection defects (such as Bluetooth signal quality). To incorporate the above described Semantic Web resources and techniques, we introduce an event model based on a context-awareness application. Because every context contains and represents concrete concepts, we address our modular event model by incorporating context semantics, as shown in Figure 6.1. The event ontology is built by analyzing both the abstract conceptualization and relevant existing ontologies.

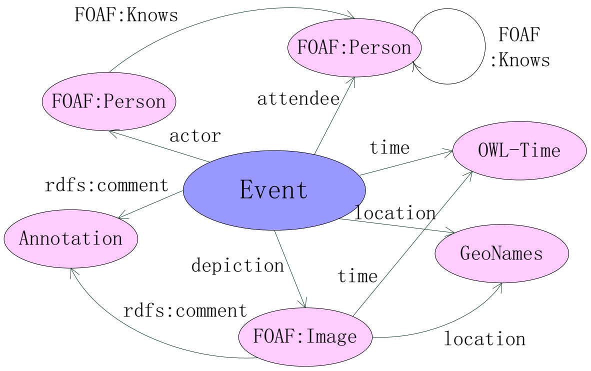


Figure 6.1: Lifelogging event ontology.

Motivated by heterogeneous lifelogging context collections, like images, GPS records and so on, we need an ontology to describe events represented by various document formats or sensor readings with detected contextual properties. The following concepts need to be specified in the context-aware event interpretation domain:

- Event: the occurrence as the intersection of time and space.
- Location: the geographical context of events.
- Time: the temporal context as a recall cue for events.
- Actor: the human who carried out the event, i.e., the lifelogger.

- Attendee: the human/humans who were present and might be involved in the event.
- Image: the class abstract for image document.
- Annotation: the class abstract for textual description of events.

```
:Event rdf:type owl:Class ;
    rdfs:subClassOf time:TemporalEntity ,
        [ rdf:type owl:Restriction ;
          owl:onProperty :hasLocation ;
          owl:minCardinality "1"^^xsd:nonNegativeInteger
        ] ,
        [ rdf:type owl:Restriction ;
          owl:onProperty :endAt ;
          owl:cardinality "1"^^xsd:nonNegativeInteger
        ] ,
        [ rdf:type owl:Restriction ;
          owl:onProperty :beginAt ;
          owl:cardinality "1"^^xsd:nonNegativeInteger
        ] .
```

Listing 6.1: Ontological event class definition

As shown in Figure 6.1, *Event* is the core class in the ontology. To keep consistent with the definition of event we presented in Chapter 5 – “the real-world occurrence at specific place and time”, we explicitly model the event class with spatial and temporal constraints in terms of OWL cardinality restrictions, as shown in Listing 6.1. The restriction `owl:cardinality` is used to confine that one event has exactly one value for the properties of starting time and ending time. In Listing 6.1, the restriction `owl:minCardinality` is stated on the property `:hasLocation` with respect to event class, indicating that any event instance needs to be related to at least one GPS location. In lifelogging, there are many cases when more than one GPS coordinate is needed to reflect the spatial characteristics of an event, such as "Walking", "Driving" and so on. More details of our event ontology are shown in Appendix B and the ontology is formalized in Turtle [19].

In the event ontology the contexts are integrated with the event class. Each context is represented with its own domain semantics and can be processed individually. This makes it flexible to process the whole event that might be represented by multiple media sources. Besides the content of events like event description and concepts, we can see from the event model that there are three main external contexts which are spatial context, temporal context and social context. For these contexts, there are already well established ontologies designed specifically to describe the domain semantics. We investigated the existing ontologies which may be reused and integrated into our context-aware event ontology and chose the OWL-Time and GeoNames ontologies to model spatial and temporal contexts respectively. In our architecture, the agents involved in the event including the actor and attendees are modeled by the FOAF (Fridend Of A Friend) ontology [5] which describes persons with their properties and relations. The visual information of events which answers the "What" question about events is depicted by SenseCam images and in this event model addressed by the `FOAF:Image` class.

6.2.3 EventCube: an Enhanced Album of Events

In designing our application for event enhancement, we mimic the behaviors of users in organizing personal digital photos. Users usually organize their digital photos in the way of an 'album'. In [140], users state that the most important feature of a photo organization tool is to automatically place photographs into albums. As shown in [130], albums are suggested to be more desirable for image organization and retrieval. Motivated by this notion, we propose a multi-contextual event enhancement architecture – EventCube, to enhance the context profiles of event models defined in Figure 6.1. Base on this event ontology, an event can be easily defined as a set of triples described in RDF. The event enhancement task is to find the relevant semantics from online knowledge repositories and social profiles, to improve the representation and subsequent recall of lifelogging events. The architecture of EventCube is demonstrated in Figure 6.2.

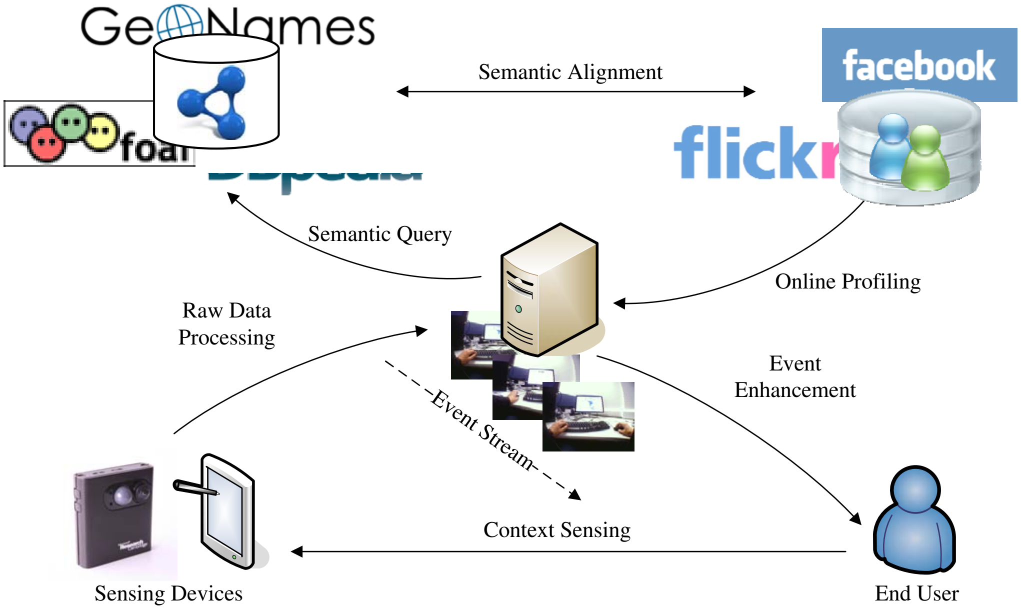


Figure 6.2: Event enhancement architecture.

For the application of this architecture to lifelogging, we employed SenseCam and Bluetooth-enabled mobile devices plus GPS modules as the context sensing devices. The processing of raw lifelogged sensor data into enhanced events can be described in three steps: First, the user uploads sensor readings to database. In this step, the SenseCam image streams are segmented into chunks and each chunk represents an event occurrence. Meanwhile, the keyframe is selected as a thumbnail for a single event. Second, the recorded GPS coordinates are clustered (to be described in Section 6.3.2) and stored, together with BlueTooth proximity records. In this step, the sensor readings are synchronized with the segmented events. Third, the online knowledge bases and social profiles are accessed and combined to retrieve the relevant semantics with current event contexts. The enhanced contents are provided as links to the end user for further navigation at his preference.

In contrast with image-based lifelogging which only utilize the visual processing such as event segmentation and keyframe selecting for event reexperience, EventCube makes full use of semantics inferred from various contexts to enhance the attributes of events like "Who", "What", "Where" and "When". Because the facets of events are handled in a combined event model, we name this multi-contextual enhancement as EventCube. Our image-based event processing and representation has been discussed in Chapter 5. Since the SenseCam images contains more information about "What" aspect of events, like event types, concept occurrences, etc., we address the encompassment of "Who", "Where" and "When" facets here. The corresponding three contextual aspects to be modeled and enhanced in the EventCube architecture include social context, spatial context and temporal context, which are described in detail as follows:

- **Social context modeling:** Social context includes information about event actor and attendee. The social context is a recall cue of the "Who" aspect for events. FOAF is an ontology to describe people profiles, their relationships and the corresponding information about things they create and are involved in. We will use the FOAF ontology to model the social context in our event ontology.
- **Spatial context modeling:** Spatial context as modeled in our event ontology includes the geographical context of event occurrences and plays the role of the "Where" cue for events. The Geo Vocabulary (World Geodetic System

1984 (WGS84) Ontology [1]) is a spacial RDF encoding widely adopted by many Semantic Web systems. It defines coordinates as instances of the `Point` class and uses predicates like `lat`, `long`, and `alt` to specify a `Point`'s latitude, longitude, and altitude settings. It is extended by the GeoNames ontology [2], in which places are modeled as geographical features for specific coordinates, as well as types and hierarchies of features. In addition, the GeoNames project also provides Web services [3] to access instance location features from the GeoNames database in various supported data format like XML, JSON, CSV, etc. The GeoNames ontology is employed in modeling spatial context in event ontologies and applied for latter spatial enhancement.

- **Temporal context modeling:** The temporal context for events is actually a span of time decided by the starting and ending time of the event occurrences. The OWL language supports time representation with standard XML Schema Definition (XSD) `date`, `time`, and `dateTime` types. But these typed literal values are too limited for event modeling. In our representation of an event's temporal context, we adopted the W3C OWL-Time ontology [4]. The OWL-Time ontology provides a vocabulary for expressing topological relations among temporal entities including instants and intervals, together with information about durations, and about datetime information. In our event temporal context modeling, the duration of an event is an instance of the `DurationDescription` class, and can also be combined with properties of `hasBeginning` and `hasEnd` to represent the starting and ending time of the event.

[1] http://www.w3.org/2003/01/geo/wgs84_pos#

[2] http://www.geonames.org/ontology/

[3] http://www.geonames.org/export/ws-overview.html

[4] http://www.w3.org/TR/owl-time/

6.3 Event Semantic Enhancement and Query

In addition to the semantics we can directly infer from event contexts, more knowledge is also needed by incorporating online information like digital gazetteer, real-time news, personal calendar, social networks, web pages, to name a few. Before interpreting event semantics using online information, we still need to solve the problem of accessing the existing semantics. Traditional organization and representations of information in the WWW pose a challenge for a computer to understand and fuse semantics from unstructured information. The representation of information in traditional WWW web pages is user-interpretable rather than computer-interpretable. In the meantime, lifelogging data is more challenging in terms of enrichment using external online information because it includes heterogeneous context information like locations, visual scene tracks, people around, etc.

6.3.1 Linked Open Data (LOD) and SPARQL

In our proposed semantic enrichment approach, the inferred context semantics from sensor readings are mediated to build a link between raw sensor data and relevant online semantic resources. The linked data cloud is such a comprehensive external knowledge repository which we can use to enrich our event interpretation. In our proposed event semantic enhancement, we will take advantage of the SPARQL query language which is the state-of-the-art semantic query language to access not only the local event semantic base but also external linked open data, to maximize the semantic interpretation of lifelogging events. This section will present a novel way of generating enhanced event semantics based on structured context metadata associated with events.

6.3.1.1 LOD-based Event Enhancement

The linked open data [4] can provide the resource-oriented knowledge base for next generation web services. It is realized by defining a URI standard for web semantics, through which a user can locate and use the digital resources using the mature HTTP/URI mechanism. The linked open data has changed the traditional way of linking documents, and instead, it tries to link the data and arbitrary information semantics in the formalized format of RDF. URIs are employed to identify any kind of resource such as object, concepts, properties and so on. Most datasets in the linked data can provide a domain-related semantic base to satisfy the needs of semantic interpretation of events in terms of browsing, navigation and semantic query.

In our lifelogging event interpretation application, we will introduce the following datasets into the semantic logging system:

- **DBpedia** : As the linked data version of Wikipedia, DBpedia is one of the most important datasets in the data cloud. The DBpedia data set currently provides information about more than 3.4 million things consisting of 312,000 persons, 413,000 places, 94,000 music albums, 49,000 films, 15,000 video games, 140,000 organizations and so on.

- **Geonames** : It is necessary in the system to interpret the semantics of event location. Geonames can provide information about over 6.2 million places and geographic features. Each Geonames toponym has a unique URL with a corresponding RDF web service.

- **DBLP bibliography** : Bibliographic information is structured in DBLP about scientific papers. The DBLP dataset now contains about 800,000 articles, 400,000 authors, and approximate 15 million triples.

- **FOAF profiles**: FOAF (Friend Of A Friend) projects provides a machine-readable ontology describing persons, their properties and relations. This vocabulary is one of the most widely used ontologies and is applied in modeling millions of RDF triples on the web. These datasets which are extracted from FOAF files or exported from other datasets can be used to interpret the Actor/Attendee information which can answer the “Who” aspect of events.

6.3.1.2 Enable the Semantic Query

To query the semantics constructed in RDF syntax, a semantic query language is required. SPARQL is a W3C recommendation for the semantic web query language. It is a recursive acronym for SPARQL protocol and RDF query language. SPARQL can be used to express queries across diverse data sources, whether the data is stored natively as RDF or viewed as RDF via middleware [132]. Its efficiency and flexibility attract wide community support and the linked open data endpoints. In summary, a large number of datasets provide access services through SPARQL queries and return results efficiently.

With a syntax similar to SQL, SPARQL is a relatively user-friendly language. SPARQL is a graph matching query language by which the semantics of interest is described as a subgraph. The query engine will match the subgraph in the whole data model (which is also an RDF graph), then the results matched are returned. SPARQL can also be used to construct new RDF graphs based on information in the queried graphs.

An example SPARQL query is shown in Listing 6.2, in which the structure of a SPARQL query can be highlighted. As we can see from Listing 6.2, SPARQL also allows the namespace abbreviation with predefined prefixes to make queries more readable. The SPARQL query contains two important components: the `SELECT` and `WHERE` syntaxes. The `SELECT` syntax defines which variable (or variables) to be returned while the `WHERE` syntax defines the premises to be satisfied by the required variables. Actually, the `WHERE` clause constructs a graph pattern which needs to be matched against the RDF repository. The results of SPARQL queries can be results sets or RDF graphs [132]. Note that specific languages can be indicated by appending `@language` to the end of a string. Here in Listing 6.2 we specify that the place name is described in English, denoted by `@en`.

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
SELECT  DISTINCT ?Abstract ?WebSite
WHERE {
    ?place <http://dbpedia.org/ontology/abstract> ?Abstract.
    ?place <http://dbpedia.org/property/website> ?WebSite.
    ?place rdfs:label "Dublin␣City␣University"@en.
    FILTER langMatches(lang(?Abstract), ’en’)
    }
```

Listing 6.2: SPARQL query example

6.3.2 Location Enhancement

Semantic enhancement of events in lifelogging needs an effective measure to locate the user because location is one key recall cue of event. Although there are some alternative measures such as Wifi SSID or GSM cell tower ID to choose from, we use GPS to record the wearer’s location considering two reasons. First, GPS is more accurate than others methods. Second, GPS is infrastructure independent which means we do not need extra devices or support and can detect location anywhere in the world. One shortfall of GPS is that it does not work inside buildings or other places in which satellite signals are unreachable. The GPS records are then used for the enhancement of location context in our application, for which the location enhancement algorithm consists of location clustering, reverse geocoding and LOD semantic query.

Location clustering is first explored on GPS coordinate records using k-means clustering [13]. We chose 100 meters as clustering radius and the filtering time span is 10 minutes. This is decided according to the fact that the distance between significant places important to users are usually farther than 100 meters and the dropouts of GPS signal cased by temporary signal block can be filtered through a 10 minutes time window. The location clustering can be described as the following steps:

1. Randomly select one GPS coordinate $P_0(x_0, y_0)$ as the original circle center, with radius $r = 100$. Choose all recorded GPS coordinates within this circle as candidates and label them as $P_i(x_i, y_i)$.

2. Calculate the centroid $P(x, y)$ of all chosen candidates $P_i(x_i, y_i)$, where $x = \sum_{i=1}^{N} x_i/N$, $y = \sum_{i=1}^{N} y_i/N$ and N is the total number of coordinate candidates located in the current circle.

3. Replace $P_0(x_0, y_0)$ with $P(x, y)$ as the new center of circle, repeat Step 1 and 2 until the distance of successive circle centers is under a predefined threshold ϵ. Save $P(x, y)$ as one significant event location and remove the coordinates $P_i(x_i, y_i)$ within the current circle.

4. Repeat Step 1, 2 and 3 for the remaining GPS coordinates until all coordinates are removed. Note that the coordinates within the same cluster (in the same circle) are all regarded as recorded in the same location and the coordinate of the location is the cluster centroid $P(x, y)$.

One example of location clustering is illustrated in Figure 6.3 using one full day's GPS records (Day_4 in Table 6.4). In Figure 6.3, blue dots represent recorded GPS coordinates while red circles are the clusters as a result of applying our algorithm. Three significant places are detected from the whole day's location traces, two of which are the lifelogger's living accommodation and lab on the DCU campus (on the

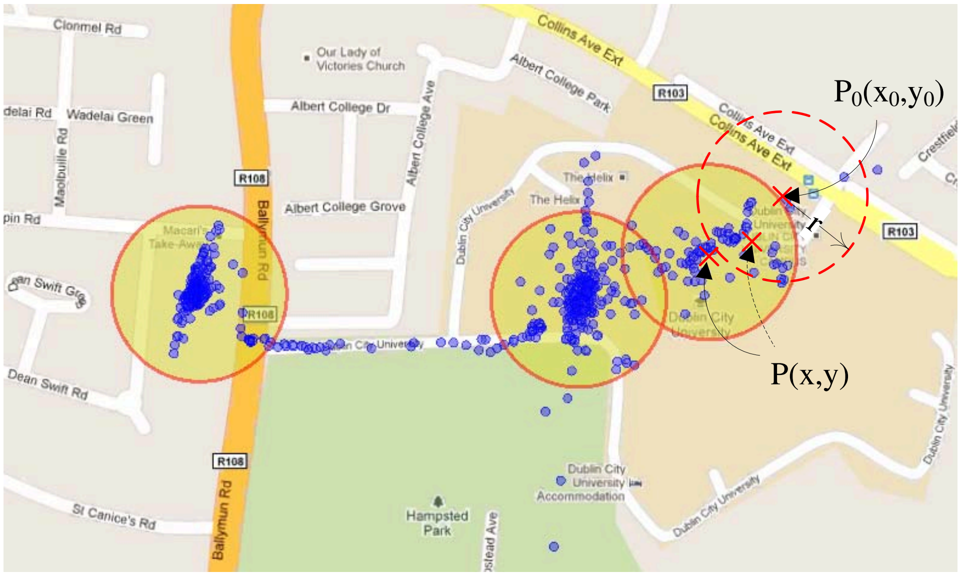


Figure 6.3: Location clustering diagram (from left to right: shop, accommodation and lab).

right) and the other one is a shop nearby (on the left). We can find that location clustering can detect the places where the lifelogger spent longer time (usually $>$ 10 minutes) while the places where the lifelogger spent little time are not considered as significant places, such as walking between these places. The diagram illustrating the procedure for clustering is also shown in Figure 6.3. The clustering starts with a random selected GPS coordinate $P_0(x_0, y_0)$ (shown as the center of dashed circle) and moves to a transitive centroid $P(x, y)$ (pointed at with the dashed arrow). The calculation is iterated to update the centroid coordinate $P(x, y)$ until stable. The final center of the circle is then regarded as the coordinate of detected place.

Reverse geocoding is necessary because the GPS coordinates contain no meaningful information to end users. No user can have an understanding of something like the following, “In the morning last Monday you were at (53.3854,-6.2574) and then went to (53.3884,-6.2564) at 1 pm.”. Reverse geocoding is used herein to translate the latitude/longitude pairs to human-readable address names. This step is performed by returning the closest addressable location, though the returned location may be some distance from the retrieved latitude/longitude pair. In Listing 6.3, the reverse geocoding snippet for “Dublin City University” is demonstrated using GeoNames web service. The returned result is an XML file containing different features of the location, such as location name, country name and distance to the retrieved coordinate (53.384954,-6.256542), etc. Some features have been truncated in Listing 6.3 and the snippet is abridged for readability.

```
    <?xml version="1.0" encoding="UTF-8" standalone="no" ?>
- <geonames>
    - <geoname>
        <toponymName>Dublin City University</toponymName>
        <name>Dublin City University</name>
        <lat>53.38541</lat>
        <lng>-6.25777</lng>
        <geonameId>6496673</geonameId>
        <countryCode>IE</countryCode>
        <countryName>Ireland</countryName>
        <distance>0.09593</distance>
    </geoname>
  </geonames>
```

Listing 6.3: GeoNames reverse geocoding

After the place name is obtained, we query the relevant semantics in DBpedia’s RDF repository. This is done through a SPARQL query by specifying the place name, as shown in Listing 6.2. The example in Listing 6.2 will retrieval the abstract description and web site link of a place specified with the name “Dublin City University”. If available, the returned results will include the information which can be matched with the `WHERE` clause. Results returned for the query example in Listing 6.2 are shown in

Table 6.1. Since the pre-selected information about the target place might limit the user's interest, we query all the semantics (properties and values) in our enhancement application and provide links for user to navigate the returned RDF graphs with a browser.

Table 6.1: Example results of SPARQL query (Listing 6.2)

Abstract	WebSite
"Dublin City University (abbreviated as DCU) is a university situated between Glasnevin, Santry, Ballymun and ..."@en	http://www.dcu.ie/

Current reverse geocoding web services label the given GPS coordinates with semantic tags by returning the nearest place names. However, due to the accuracy of GPS and different sizes of places, the nearest place is not guaranteed to be the right answer for the target event. In such cases, other places near the given GPS coordinates are also likely to be the right ones, or at least helpful for the user to recall the geographical information of the region where the event took place. To deal with this issue, we provide the nearby places as a ranked list for the user and enhance the selected places at the user's preference. We rank the place list according to their popularity analyzed from Flikr social tags. The assumption held for this processing is that the better known places could be easier for recall when the user reminisces about an event. In addition, the most popular places are usually a benchmark of the region, so the user can benefit from it and realize where he was during the event.

6.3.3 Social Context Enhancement

As shown in the event ontology depicted in Figure 6.1, the actor and attendee contexts together reflect the agent aspect of a lifelogging event. While these two contexts answer "Who" is carrying out the event and "Who" else is involved, social context enhancement tries to enrich the social profiles of these agents. In our performing of social context enhancement, the FOAF profile and lifelogger's personal information in Facebook are combined.

FOAF profiles are datasets in LOD and contain personal information modeled in RDF. While FOAF profiles contain information about millions of persons including relevant or irrelevant persons to the event, lifeloggers' social profiles like Facebook contain more semantics which have been customized and might have higher correlation with lifelogging events. When a user reexperiences his lifelogging events, social information can improve the understanding of the "Who" aspect. The combination of FOAF profiles and Facebook involves the following procedures: First, the XML feeds from Facebook need to be transformed to a form of RDF. Second, the FOAF profiles and Facebook are integrated in the same data model for which the same vocabularies like the FOAF ontology are needed for consistent semantic representations. Third, the RDF statements are populated to the event model for social context enhancement.

Facebook is one typical social media sharing web site by which registered users can establish a social networking profiles including shared friend information, pictures, messages, and so on. A user's social network information can be accessed through a web service API to retrieve the XML results as a stream over HTTP. To facilitate efficient semantic modeling, especially the event ontology we built in Section 6.2.2, as well as the SPARQL query language, we need to convert the XML-based profile representation to a more extensible RDF model.

As one form of machine-readable data format XML is used to interchange data between applications which need to convey information to diverse end users. Due to its simplicity and flexibility, a large number of data sources are formalized in XML and many XML processing tools have been developed. From XML, The transformed RDF model also makes it easier to combine semantics together into a common knowledge model. The output of transformation of Facebook XML results is an RDF/XML containing the information reflected by the XML source document. The RDF/XML file can then be handled as an RDF model and be output to other RDF file format such as Turtle, N3 and N-Triples. Because the transformed model needs to be integrated with FOAF profiles queried from LOD, we simplify the semantic alignment by converting the Facebook XML directly to semantics modeled by the FOAF ontology. The detailed description of semantic transformation for social context enhancement will be described in Section 6.4.

6.4 Event Semantic Enhancement Experiment

6.4.1 Experiment Setup

The event enhancement experiment is expanded based on the event ontology and enhancement architecture we described in Section 6.2. The experiment has two main procedures, which are lifelogging event recording and event semantic retrieval.

6.4.1.1 Event Recording Setup

SenseCam is employed in our experiment to collect images and movement data as well as temperature and light levels with its on-board camera and sensors. Among these heterogenous sensor readings collected by SenseCam, we only use images in our event enhancement application. However, the other sensor readings especially accelerometer recordings are helpful for SenseCam to decide when to trigger image capturing, hence are stored together with SenseCam images into our database.

GPS recording and Bluetooth detection are implemented on an Nokia n810 internet tablet with client software built to communicate with an external GPS module. A GPS data steam is received and recorded every 10 seconds. The nearby Bluetooth unique addressea and friendly device names are logged with a time stamp. The Bluetooth detection time interval is 20 seconds. Note that all these sensor readings including SenseCam images are recorded with time stamps and then synchronized through the same time line when stored in our database.

6.4.1.2 Retrieval Environment Setup

One user in our group has been wearing the above recoding devices for one month for our event enhancement experiment purpose. We process the storage and retrieval on a daily basis, which means the user uploads the SenseCam data collection and GPS plus Bluetooth readings after one day's continuous recording. The enhancement on such lifelogged event data involves the combination of semantics from two spaces: physical space and information space.

For retrieval of physical information recorded by ambient sensing devices such as SenseCam, GPS and Bluetooth, we apply the SenseCam browser [53] to segment a whole day's SenseCam data streams into individual events. The events are also indexed with relevant images and keyframes are selected for visual representation of events. Since the SenseCam browser does not provide facilities to deal with spatial or social contexts collected by GPS and Bluetooth, we process these context recordings separately with external applications and also upload the results into the database for later retrieval. The combination of event segmentation, location clustering and Bluetooth records can form a whereabouts log as shown in Table 6.2.

Event snippets of a typical working day are illustrated in Table 6.2. After location clustering, the cluster centroids are used to represent the final coordinates for significant places as we described in Section 6.3.2. As to 'Traveling' events, one single cluster is not enough to reflect the whole traveling trail. The starting location and ending location are both used to model such events. This is also applicable to some events during which GPS signals are lost. In such cases, the location where signal dropout started and the signal was resumed are recorded as the starting and ending locations for such events. As shown from Table 6.2, the Bluetooth MAC addresses have no semantic meaning at all and are of no use for event reminiscence. Mobile

Table 6.2: The whereabouts log

Event	Lat/Long	Starting	BT MAC Address	BT Device Name
79	53.38,-6.26	10:08	00:17:F2:BA:17:F9 00:23:12:5B:B0:99 ...	Daragh Byrne' iMac NeilOHare-MacBook ...
80	—	12:29	00:1B:EE:3F:BE:0F 00:26:5D:F5:CB:AE ...	Nokia 7373 SGH-J700I ...
81	53.38,-6.25	12:45	00:17:F2:BA:17:F9 00:23:6C:BB:6A:C3 ...	Daragh Byrne' iMac cdvpminiColum ...
82	—	13:20	9C:18:74:EF:15:65 00:16:BC:D5:A7:4A ...	Nokia N97 Madge ...
...	...	...	...	...

device owners often set their Bluetooth device names in a more friendly way, such as “Daragh Byrne' iMac”, “NeilOHare-MacBook” and so on. These friendly device names are cues for a user to realize “Who” he was with, or “Where” he went, during the specific event.

Besides the local ambient information access, the retrieval environment also includes the information space constructed by online semantic repositories and users' social profiles. The retrieval of online knowledge bases such as datasets in LOD aims to fulfill the task of enhancing the “Who” and “Where” aspects of events. Most LOD datasets have provided SPARQL query endpoints for the sharing of domain semantics. In our event interpretation using such query-based data resources, we employ the SPARQL semantic query language and the data sources are listed in Table 6.3. In the list, GeoNames, Flickr and Facebook have no SPARQL endpoints provided and we use their web services for information access. Note that datasets of DBpedia, GeoNames and DBLP in Table 6.3 are all members of LOD.

Table 6.3: Online data sources employed

Dataset	Web Service Endpoints	Event Aspects
DBpedia	http://dbpedia.org/sparql	Who, Where
GeoNames	http://www.geonames.org/export/	Where
Flickr	http://www.flickr.com/services/api/	Where
DBLP (Hannover):	http://dblp.l3s.de/d2r/	Who
Facebook	http://api.facebook.com/1.0/	Who

6.4.2 Aligning Semantics for Social Enhancement

The Facebook web service provides the access to users' social networking profiles through XML data streams. However, the FAOF profiles queried from DBpedia repository are all modeled in RDF. Because RDF/XML is indeed an XML syntax to describe RDF triples, exposing semantics to RDF from XML can be done by an XML transformation. Extensible Stylesheet Language Transformations (XSLT) is a XML processing tool to convert data representation between different XML documents. XSL includes an XML vocabulary for specifying formatting and specifies the styling of an XML document by using XSLT to describe how a document is transformed into another XML document that uses the formatting vocabulary [45]. The XSLT template rules are used to specify the mapping between the elements of the source XML document and elements of the output document. Applying an XSLT document to a source XML document and generating a new XML document is typical XML processing, and more details about XSLT template rules and XML transformation can be found in [45].

In our experiment, we apply XSLTs to align semantics between XML-based and RDF-based data representation. The effectiveness is also shown in our experiment. For example, Listing 6.4 is the snippet of original XML source feed from the Facebook web service. After transformation by employing XSLT, the returned RDF-based semantic model is shown in Listing 6.5, in which the statement triples are all formalized in Turtle. Both Listing 6.4 and Listing 6.5 have been abridged for readability. From

Listing 6.5 we can find that the information is reformatted in a more readable style.

```
<?xml version="1.0" encoding="UTF-8"?>
  <user>
    <uid>692153372</uid>
      <affiliation>
        <nid>16779809</nid>
        <name>DCU</name>
        <type>college</type>
      </affiliation>
    <birthday>June 1</birthday>
    <name>Cathal Gurrin</name>
    <pic>http://profile.ak.fbcdn.net/...4189469_s.jpg</pic>
  </user>
```

Listing 6.4: XML-based data source (abridged)

```
@prefix rdf:<http://www.w3.org/1999/02/22-rdf-syntax-ns#>.
@prefix foaf:<http://xmlns.com/foaf/0.1/>.
@prefix ec:<http://www.clarity-centre.org/EventCube#>.
<http://www.clarity-centre.org/EventInterpretation#user692153372>
      rdf:type foaf:Person ;
      ec:hasAffiliation
              [ rdf:type foaf:Organization ;
                foaf:name "DCU" ] ;
      foaf:birthday "June␣1" ;
      foaf:depiction "http://profile.ak.fbcdn.net/...4189469_s.jpg" ;
      foaf:name "Cathal␣Gurrin" .
```

Listing 6.5: Aligned semantics in Turtle (abridged)

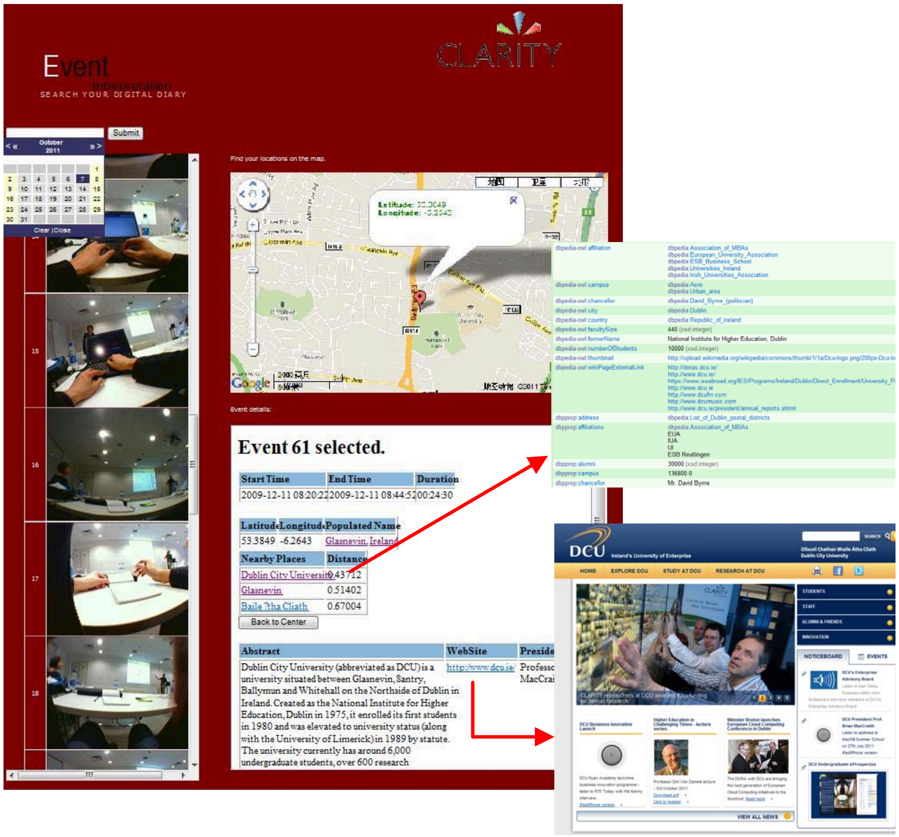
While the result shown in Listing 6.5 accurately reflects the information contained in the XML source document, it is not hard for us to notice that the new representations of semantics use the prevalent vocabularies such as `rdf`, and `foaf`, which are widely adopted for semantic modeling in DBpedia knowledge base and other FOAF profiles. The common semantic description also make it easier to combine various knowledge sources for a more comprehensive event enhancement.

6.4.3 Event-Centric Enhancement Application Overview

In our lifelogging semantic enhancement, the event is still the basic unit for us to reveal underlying semantics. This notion is also reflected by the event ontology we built in Figure 6.1. In this section, we apply this notion into an event-centric enhancement application tool. The application tool is built for the purpose of event context enhancement and event semantic visualization. The enhancement tool is a browser-based application with a SenseCam event viewer, geospatial map and contextual enhancement browser embedded, as shown in Figure 6.4.

The event viewer lists event keyframes sequentially allowing the user to view his events on a day by day basis. The calendar on the left corner of Figure 6.4 provides the user with the selection of a specific day. After the user selects a target date he wants to review, the event viewer will list all events which have been segmented for the day. Event representations are organized in a temporal order for the whole day to reflect the progress of events. Figure 6.4 illustrates temporal progress when the lifelogger attended a presentation. The sequence includes starting-up the laptop, listening to the presentation, taking notes, etc., all of which can be visualized in the event viewer.

When the user wants to step through the details of event contexts, he can click the event keyframe and contextual information will be enhanced and visualized in the geospatial map and contextual information browser. Figure 6.4 demonstrates



Event 61 selected.

Start Time	End Time	Duration
2009-12-11 08:20:22	2009-12-11 08:44:52	00:24:30

Latitude	Longitude	Populated Name
53.3849	-6.2643	Glasnevin, Ireland

Nearby Places	Distance
Dublin City University	0.43712
Glasnevin	0.51402
Baile ?tha Cliath	0.67004

Back to Center

Abstract	WebSite	Preside
Dublin City University (abbreviated as DCU) is a university situated between Glasnevin, Santry, Ballymun and Whitehall on the Northside of Dublin in Ireland. Created as the National Institute for Higher Education, Dublin in 1975, it enrolled its first students in 1980 and was elevated to university status (along with the University of Limerick) in 1989 by statute. The university currently has around 6,000 undergraduate students, over 600 research	http://www.dcu.ie/	Professo MacCrai

dbpedia-owl:affiliation	dbpedia:Association_of_MBAs dbpedia:European_University_Association dbpedia:ESB_Business_School dbpedia:Universities_Ireland dbpedia:Irish_Universities_Association
dbpedia-owl:campus	dbpedia:Acre dbpedia:Urban_area
dbpedia-owl:chancellor	dbpedia:David_Byrne_(politician)
dbpedia-owl:city	dbpedia:Dublin
dbpedia-owl:country	dbpedia:Republic_of_Ireland
dbpedia-owl:facultySize	440 (xsd:integer)
dbpedia-owl:formerName	National Institute for Higher Education, Dublin
dbpedia-owl:numberOfStudents	10000 (xsd:integer)
dbpedia-owl:thumbnail	http://upload.wikimedia.org/wikipedia/commons/thumb/1/1a/Dcu-logo.png/200px-Dcu-lo
dbpedia-owl:wikiPageExternalLink	http://doras.dcu.ie/ http://www.dcu.ie/ https://www.iesabroad.org/IES/Programs/Ireland/Dublin/Direct_Enrollment/University_P http://www.dcu.ie http://www.dcufm.com http://www.dcumusic.com http://www.dcu.ie/president/annual_reports.shtml
dbpprop:address	dbpedia:List_of_Dublin_postal_districts
dbpprop:affiliations	dbpedia:Association_of_MBAs EUA IUA UI ESB Reutlingen
dbpprop:alumni	30000 (xsd:integer)
dbpprop:campus	136800.0
dbpprop:chancellor	Mr. David Byrne

Figure 6.4: Event enhancement interface (left: event viewer; right: map and enhancement browser).

the enhancement and visualization of spatial context for the event of listening to a presentation. After the user picks the event he is interested in, the corresponding GPS location is queried and located on the map. The enhanced context information is acquired through the aforementioned methodology in this chapter. Two categories of enhanced context are visualized in the information browser, which are location context and social context. To enhance the location context, the relevant place names are retrieved according to the event GPS coordinates. The abstract information is shown in the browser as a brief description for the most relevant named place. The browser also provides the user with further details of these places through links from their web pages or RDF triple repositories, as shown in Figure 6.4. Social context is enhanced and visualized in the same manner with brief information and links to external semantics. Social context enhancement also utilizes the DBLP dataset to allow the user to drill down into more detailed personal information if available, in addition to DBpedia. The temporal context is visualized with a time stamp indicating the starting time, ending time and duration of the selected event.

After the mobile devices are carried by user as wearable devices for one month, we selected 25 consecutive days of lifelogged data to evaluate our methodology of event semantic enhancement. The final dataset includes 38,026 images, 327,244 GPS records and 45,898 Bluetooth detections involving 958 unique devices. An excerpt of three days event enhancement records are shown in Appendix C. For simplicity, we only demonstrate the top three place names and Bluetooth friendly names in the table. Detailed results of our enhanced contexts are described in Section 6.4.4.

6.4.4 Assessing Context Enhancement

In enhancing the relevant contexts of events, we first apply the location clustering algorithm described in Section 6.3.2 into finding the significant places. Since SenseCam images are collected simultaneously together with GPS coordinates, the ground truth of places where the user stayed for a relatively long time in one day can be judged by looking through the event keyframe representations. The GPS records for the selected 25 consecutive days are first validated in order to filter the invalid coordinates such as empty GPS records logged when satellite signals are invisible. Finally, 59,164 GPS coordinates are selected for location clustering and each day's locations are clustered with about 2,400 coordinates on average. The clustered significant places are judged and shown in Table 6.4.

Clustering and judgement are carried out on a day basis, as shown in Table 6.4. We define *true positive* as the number of place clusters detected by our algorithm and where events actually happened as detected. *False positive* stands for the number of clusters detected but no events happened there, *false negative* is the number of

Table 6.4: Evaluating location clusters

Day ID	**Clusters**	**True Positive**	**False Positive**	**False Negative**
Day_1	2	2	0	0
Day_2	1	1	0	0
Day_3	1	1	0	0
Day_4	3	2	1	1
Day_5	3	2	1	2
Day_6	1	1	0	0
Day_7	1	1	0	0
Day_8	2	1	1	1
Day_9	3	2	1	1
Day_10	2	2	0	0
Day_11	3	3	0	0
Day_12	2	2	0	0
Day_13	1	1	0	0
Day_14	3	3	0	0
Day_15	2	2	0	0
Day_16	2	2	0	3
Day_17	2	2	0	0
Day_18	1	1	0	0
Day_19	1	1	0	0
Day_20	1	1	0	0
Day_21	4	3	1	0
Day_22	1	1	0	0
Day_23	1	1	0	0
Day_24	2	2	0	3
Day_25	1	1	0	0
Total	**46**	**41**	**5**	**11**

undetected clusters but some events turned out to happen there. Because we did not consider the detection of sublocations, we assume the sublocations within the same cluster belong to the same place and have the same place name. Under this assumption, the nearby places (distance less than 100 meters) like different rooms in the same lab, the meeting room in the same building are all regarded as the same place.

In total, there are 46 clusters detected for 25 days' events. According to user judgement, the *precision* and *recall* of our clustering algorithm are 0.891 and 0.788 respectively. The location clustering algorithm has relatively low recall compared with its high precision. That's because of the dropouts of GPS signals, especially in some places surrounded by tall buildings. On Day_16 and Day_24, more places in Dublin city center are missed by our algorithm because not enough GPS records are collected for those indoor events. However, the noise caused by GPS accuracy is handled better by our clustering algorithm which is reflected by low false positive values. In most cases, the error of GPS location is under 100 meters and can be filtered by clustering.

Several days of location clustering results are visualized in Figure 6.5 on a map. The recorded GPS locations are represented with blue dots in Figure 6.5 while the red circles stand for clustered significant places. To demonstrate the performance of our place detection algorithm, we illustrate the results with several routine days (on the left of figure) for Day_1, Day_10, Day_14 and Day_17 in Table 6.4. A more interesting day when the lifelogger spent the whole day in Dublin city center (Day_5) is illustrated on the right of Figure 6.5. These sample results show that our location clustering algorithm works well for a small geographical region and can also be scaled to broader ranges. The converted human-readable names for detected places in Figure 6.5 are also shown in our enhanced event records, as listed in Appendix C. One more trial of location clustering is also tested on another researcher in our group who is more

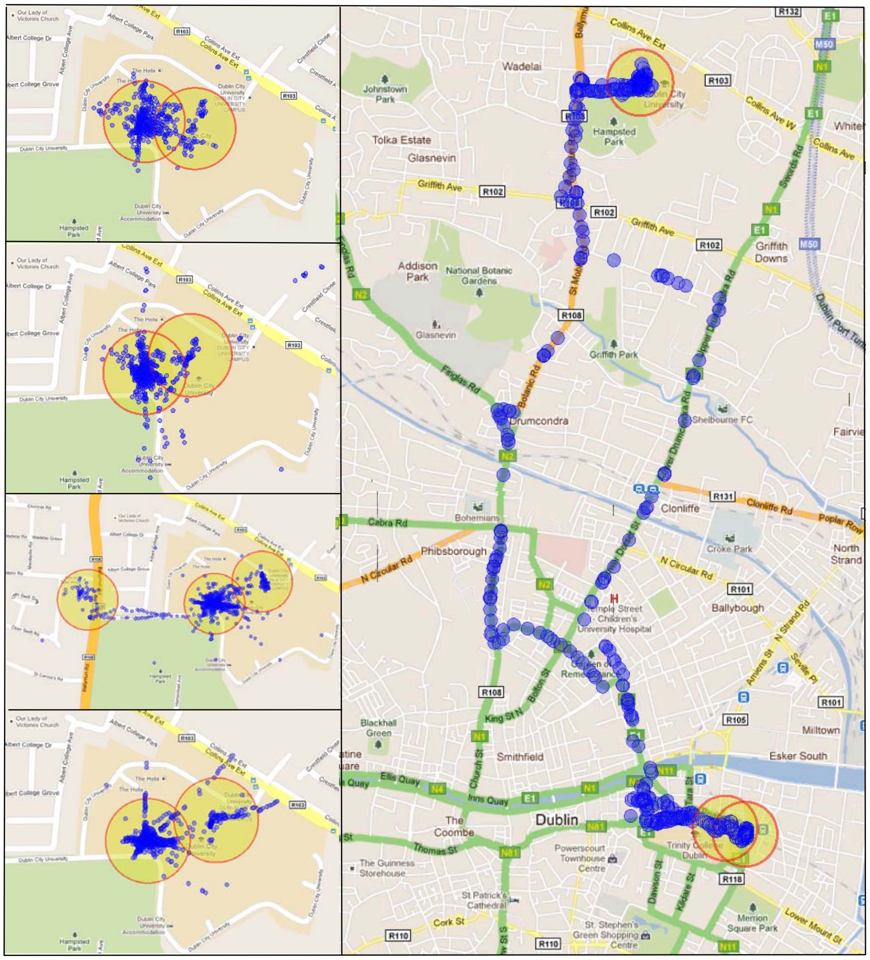


Figure 6.5: Day samples of place detection. (left: routine working days; right: one day in city center)

'active' in traveling than the lifelogger whose records has been illustrated in Figure 6.5. His one typical day's GPS logs and detected significant places are demonstrated in Appendix D.2.

The running of our location clustering was also carried out on a whole month's GPS records for a single individual. The clustered results are shown in Appendix D.1. We find that the noise of a longer time span will accumulate and cause more incorrectly detected significant places, as shown in Appendix D.1. In addition to its lower accuracy, the detection on a one month time scale is also more computationally complex than that on the day basis. In our experiment, the average elapsed time for

Table 6.5: Enhanced samples for places

Place Name	**Abstract**	**Home Page**
Dublin City Univ.	a university situated between Glasnevin, Santry, Ballymun and . . .	www.dcu.ie
Trinity College	formally known as the College of the Holy and Undivided Trinity of . . .	www.tcd.ie
Glasnevin	a largely residential neighborhood of Dublin, Ireland . . .	–
Baile Átha Cliath	capital and largest city of Ireland . . .	www.dublincity.ie
Croke Park	the principal stadium and headquarters of the Gaelic Athletic Association (GAA) . . .	www.crokepark.ie
Book of Kells	an illuminated manuscript Gospel book in Latin, containing the four Gospels of . . .	–
Westin	an upscale hotel chain . . .	–
Merrion Square	a Georgian square on the southside of Dublin city centre . . .	–
Leinster House	the name of the building housing the Oireachtas, the national parliament of Ireland . . .	–
The Spire	a 1964 novel by . . .	–
Marino	a Northside suburb located in Dublin . . .	–

processing one day's GPS data is only 0.695 second but clustering one month's data takes 818.196 seconds, tested on a desktop PC (2.66GHz Dual Core Processor, 4.00GB Memory(RAM)). Though some mis-detected places on accumulated GPS noise can be filtered by applying sophisticated time constraints, we argue that detection of significant places using our algorithm on a day basis is more suitable for users to reminisce about events. Mis-detected places or non-relevant places with respect to the date picked by the user will only cause confusion to him/her. In our later discussion, the contexts are still processed and enhanced on a day by day basis.

The enriched location context by DBpedia is shown in Table 6.5, in which the abstracts (defined by `dbpedia-owl:abstract` predicate), home pages (defined by `foaf:homepage`) are demonstrated for simplicity, if available.

After applying the SPARQL query, the relevant semantics about various places are retrieved from DBpedia. Besides abstract and home page, there might be dozens of properties queried from DBpedia for location enhancement. The relevant properties about the target place also include the type of the place, the exact geospatial location information, affiliation, image, etc., which are all provided as links as an enhancement interface for users to navigate, as shown in Figure 6.4. As reflected by Table 6.5, we did not apply place name disambiguation before applying the enhancement. "The Spire" is enhanced as a novel in Table 6.5, which is not the true interpretation of its meaning as a tourist attraction. However, the `dbpedia-owl:wikiPageDisambiguates` property allows users to navigate various options of resources with the same name "The Spire" and choose the right one, which is described as "the Monument of Light . . . on O'Connell Street in Dublin, Ireland".

Similar to location enhancement, social context enhancement is also performed and visualized in the enhanced browser shown in Figure 6.4. While most benchmark locations can be queried from DBpedia datasets, not many persons involved in the event can be enhanced by DBpedia so we enhance the social context by combining different resources of DBpedia, DBLP and the lifelogger's Facebook social profiles. As illustrated in Appendix C, not all of the Bluetooh records are useful in enriching the social context of events. In our application, we allow the lifelogger to edit the real friend names to be mapped to the Bluetooth friendly names. Social context is then enhanced by querying relevant information from the aforementioned data sources by interlinking the friend's name to those data sets.

Table 6.6 shows some samples for enhanced social contexts in Appendix C. For simplicity, we only illustrate the person abstracts obtained from DBpedia in the table. The column of DBLP shows the number of records in DBLP datasets reflected by the number of `dc:creator` or `foaf:maker` properties queried from DBLP. The semantics retrieved from the Facebook data source are obtained by aligning from XML streams

Table 6.6: Enhanced samples for social context

Bluetooth Name	DBpedia	DBLP	Facebook
Daragh Byrne's 24inch iMac	–	23	–
NeilOHare-MacBook	–	13	Drogheda, Ireland
Alan Smeaton's MacBook Pro	Alan Smeaton is an author and academic at Dublin City University . . .	227	–
cdvpmini-AlansOffice	Alan Smeaton is an author and academic at Dublin City University . . .	227	–
cdvpminiColum	–	12	–
Pete	a British multimedia artist living in Newfoundland, Canada . . .	23	–
Jiang	–	30	Pengxian, China
Dermot Diamond's Computer	–	21	–

to RDF models as described in Section 6.4.2. Similarly, for the Facebook column, we only demonstrate the hometown defined by `ec:hasLocation` in the aligned RDF models using our event ontology.

As we can see, our approach to semantic enhancements can utilize the information retrieved from various sources by applying SPARQL which is a state-of-the-art semantic query language, and aligning semantics to standardized RDF model. The populated personal profiles provide a comprehensive tool for the user to realize the detailed aspects about the social contexts of events. We believe this kind of semantic enhancement based on various well-structured knowledge could also be a solution for more complicated and customized services like the scenario we illustrated in Section 6.2.1. The same problem caused by lack of name disambiguation is mis-enhancement for some commonly-used names in datasets. For example, the recorded person 'Pete' (Peter as real name), who was a colleague of the lifelogger in the same lab, is incorrectly enhanced as a British multimedia artist by querying DBpedia. The characteristics of Bluetooth also cause another artifact for social context enhancement. Bluetooth has a range of about 10 meters and in some cases it can penetrate walls. This means that even some Bluetooth devices are not physically proximate, i.e., should not be regarded as involved in the event, they are still logged as the social context. In our enhancement experiment, we rank the Bluetooth records in terms of their frequency during the time span of selected event. In this way, accidentally logged device proximities can be ranked lower and have less chance to be enhanced.

6.5 Summary

This chapter elaborated our methodology of how to improve the interpretation of event contexts with external knowledge bases, which is also called event enhancement. Because more and more semantics are modeled and formalized in LOD datasets by Semantic Web technologies, we investigated the application of modern Semantic Web technologies, such as domain semantic ontological modeling, online triple store accessment, etc., into our event enhancement. This chapter first discussed the prevailing technologies for semantic modeling and ontologies, as well as standard languages for Web semantic representation and sharing. These technologies are then exploited to build an event ontology for our lifelogging event modeling with multiple contexts embraced into one consistent model. Based on this model, we also discussed location and social context enhancement by accessing semantics from various online data sources including the Linked Open Data (LOD), social media and lifeloggers' own Facebook profiles. In this step, SPARQL was used for efficient data query from LOD datasets. Finally, in our experiments, we built an application tool for lifelogging event enhancement to assimilate the aforementioned data sources for event contexts enrichment. For a consistent semantic representation using our event ontology, the semantic alignment of different data sources was also discussed in the experiment part.

Chapter 7

Conclusions and Future Work

In this thesis, we tackled the comprehensive area of event processing for visual lifelogging. Aiming to fill in the semantic gap between raw media data and lifelogging users' expectations, we focused on semantic interpretation of events to build a mapping from lifelogging data collections to high-level semantics. Our understanding of lifelogging events is based on the notion that the sensor readings are all descriptions of the event rather than the event itself. With this notion, we believe that the semantics of events can be maximally interpreted to provide an efficient tool for use as a memory aid, medical analysis of activities of daily living (ADL), market research or even future context-aware web services.

Our methodology is twofold in dealing with semantic interpretation of events. Our semantic mining comes not only from the visual media such as SenseCam images which are the direct reflection of event semantics, but also from external online knowledge repositories which play indirect roles in event interpretation. The essential elements with regard to semantic interpretation from these two different semantic sources are as follows:

Visual semantic detection: The task of visual semantic detection is to index local lifelogged collections such as SenseCam image archives, with human-understandable features. We adopted state-of-the-art concept-based multimedia processing for this task.

- High-level feature detection
- Multi-concept fusion
- Event semantic representation
- Concept-based event classification

Semantic enhancement: Semantic enhancement uses external knowledge bases for context enrichment of events. To fulfill this task, we employed the linked open data cloud as the main data sources from the Semantic Web.

- Multi-contextual event modeling
- Semantic query

Our research questions are examined when applying the above tasks to lifelogging event interpretation. The corresponding research questions derived from these tasks are now revisited as follows:

(RQ1) What concept ontology needs to be defined to satisfy the needs for indexing everyday multimedia in lifelogging ?

(RQ2) How can we automatically select proper concepts for a given activity topic ? How can we perform semantic reasoning in the lifelogging domain ?

(RQ3) How can we classify different activities and represent them when there are severe visual diversities ?

(RQ4) How can we enhance the semantics of lifelogging activities using Semantic Web technologies ?

Generally speaking, the research questions **(RQ1)**, **(RQ2)** and **(RQ3)** are raised for the task of visual semantic detection while **(RQ4)** deals with the semantic enhancement task. Before applying concept-based multimedia indexing to lifelogging visual images, **(RQ1)** and **(RQ2)** need to be answered because the appropriately selected lexicon and automatic reasoning on it are needed to facilitate efficient semantic description in a given domain. In concept-based information retrieval, a user's expectation needs to be mapped to a group of high-level feature detectors. The answer to question **(RQ2)** also addresses the automatic mapping between everyday activity and concepts. While more concepts might be involved in everyday lifelog media and these concepts are usually not independent to each other, **(RQ3)** is brought forward to fuse any erroneous concept detections for further applications of activity detection and semantic representation. **(RQ4)** is proposed to deal with the issue of applying cutting-edge Semantic Web technologies into contextual enhancement of events. Trying to answer these research questions, different algorithms are developed and demonstrated to be effective in Chapter 3, Chapter 4, Chapter 5 and Chapter 6, which are the main contributions of this thesis.

7.1 Main Contributions

A density-based semantic concept selection algorithm was introduced in Chapter 3 for the purpose of topic-related automatic selection. Semantic Web technology has come to a certain level of maturity for modeling domain semantics as ontology graphs connected by various concept relationships. Our density-based concept selection algorithm utilizes concept similarity reasoned from these ontologies and applies them to decreased mapping ambiguity between everyday activity and concepts. In Chapter 4, a user experiment was also carried out to generate a set of concepts with respect to these activities in the lifelogging activity domain. The effective performance of automatic concept selection has been demonstrated with two comprehensive ontologies, which are *WordNet* and *ConceptNet*. Various ontological similarity measures including lexical similarity and contextual similarity, are investigated on these two ontologies. The experiments on both lifelogging and TRECVid lexicons show that density-based concept selection can utilize the global similarity of concepts in concept selection and ranking, then archive satisfactory performance. To the best of our knowledge, the investigation of comprehensive ontological similarities for lifelogging domain concepts, as we reported in Chapter 3 and Chapter 4, has never been done before.

Event-level concept fusion and activity classification are another contribution of this thesis, addressed in Chapter 5. Since image-level concept detection is prone to be erroneous and can not reflect the semantics at event level when these images are visually diverse, an interestingness-based concept aggregation approach is proposed and has been applied into selection of event keyframes. The better image quality of selected event representations demonstrates the efficacy of concept fusion. Ontological multi-concept classification is also discussed in Chapter 5. By explicitly modeling concept relationships with a Semantic Web ontology language, the utilization of concept semantics to concept detection has been demonstrated to be effective in improving the traditional one-per-class concept detection performance. A HMM-based activity classification algorithm is proposed in Chapter 5 to make use of image-level concept appearance patterns, in order to decide the type of activity at the event level. The performance of this HMM-based activity detection algorithm is assessed on concept detectors with various levels of detection accuracy. The algorithm is shown to be robust to concept detection errors and is also shown to be effective in activity classification based on the learned time-varying concept dynamics.

The third contribution of the thesis is applying external semantic repositories from the Semantic Web into enhancing the interpretation of lifelogging events. This is covered in Chapter 6 from a multi-context point of view. Chapter 6 modeled event class and corresponding contextual semantics in an event ontology with a Semantic Web description language. In this event model, prevailing ontologies are reused in order to incorporate context semantics extracted from raw sensor readings, with external knowledge. Based on this lifelogging event model, event enhancement can be performed by querying the most relevant context semantics from online knowledge repositories of linked open data, through Semantic Web technologies. In Chapter 6, we illustrated our methodology for enhancing location and social contexts of events. We accessed various Semantic Web datasets like DBpedia and DBLP through state-of-the-art semantic query language – SPARQL in our enhancement tool. The enhanced and structured event semantics, derived from raw sensor data like GPS and Bluetooth records, has demonstrated the effectiveness of Semantic Web technologies in enriching lifelogging events.

These main contributions tackled the four research questions we just revisited. Semantic Web technologies have been employed in all three contributions, at different levels of abstraction. Since not one single technology, either Multimedia Retrieval or Semantic Web, can successfully fulfill the task of semantic interpretation of events in lifelogging, Semantic Web technologies have been assimilated in our contributions to address the research questions together with traditional Multimedia Retrieval technologies like supervised machine learning, unsupervised machine learning, Vector Space Model (VSM), etc. As answers to the research questions, the contributions of this thesis have supported our hypotheses formulated at the beginning of thesis, that is, “Semantic Web technologies can support the interpretation of event semantics in lifelogging”.

7.2 Future Work

Our algorithms and models have shown their merits to some extent in fulfilling event semantic interpretation tasks. But not all of them are free of limitations. In Chapter 5, an everyday concept ontology is applied to adjust the confidence values returned by traditional binary concept detection. This is carried out by learning concept correlations by fitting models of sigmoid functions. In this way, concept relationships are used indirectly for a multi-concept classification purpose. A similar limitation is also faced with in our algorithm for activity classification, where Latent Semantic Analysis (LSA) is introduced to project the dataset to a new concept space. The semantic relationships of concepts has been assumed and then handled implicitly with a factor analysis approach of Singular Value Decomposition (SVD). This way of adopting concept semantics is still a data-driven approach, together with the multi-concept confidence adjusting based on sigmoid-learning. It is not hard for us to see the possible limitations that are faced with when scaled up to larger datasets. Since concept semantics can be modeled explicitly in ontologies, an approach directly applying concept relationships to these tasks will have higher scalability and efficiency. This kind of knowledge-driven semantic adoption will be part of our future work.

Concept detection is the basis for further processing of event interpretation. The 27 concept detectors have been introduced in our event level semantic fusion and shown to be effective. Though representative in evaluating our algorithms, these detectors can not reflect more semantics in real-world application. Experiments on larger concept detection data sets will be another topic for future work.

In this thesis, we dealt with event interpretation from internal and external aspects. Internal semantics are extracted from SenseCam images by our detection and fusion algorithms, while external semantics are accessed from online knowledge bases. Both of these two parts of semantics are essential for a better understanding of lifelogging events. What current work still lacks is an effective approach to incorporate them. Though our event ontology provides a way for representing these semantics, a more powerful tool is also needed in future work, to validate and link up these semantics associatively.

Bibliography

[1] http://lastlaugh.inf.cs.cmu.edu/lscom/. Last accessed: Sep. 2010.

[2] http://www.statistics.gov.uk/StatBase/ssdataset.asp?vlnk=7038&More=Y. Last accessed: Sep. 2010.

[3] http://www.statistics.gov.uk/StatBase/ssdataset.asp?vlnk=9497&Pos=&ColRank=1&Rank=272. Last accessed: Sep. 2010.

[4] http://www.w3.org/DesignIssues/LinkedData.html. Last accessed: Sep. 2010.

[5] http://xmlns.com/foaf/spec/. Last accessed: Oct. 2011.

[6] LSCOM lexicon definitions and annotations version 1.0. Technical report, Columbia University, March 2006.

[7] Bill Adams, Arnon Amir, Chitra Dorai, Sugata Ghosal, Giridharan Iyengar, Ro Jaimes, Christian Lang, Ching yung Lin, Apostol Natsev, Milind Naphade, Chalapathy Neti, Harriet J. Nock, Haim H. Permuter, Raghavendra Singh, John R. Smith, Savitha Srinivasan, Belle L. Tseng, Ashwin T. V, and Dongqing Zhang. IBM Research TREC-2002 video retrieval system. In *Proceedings of the TREC-2002*, pages 289–298, Gaithersburg, Maryland, 2002.

[8] Akiko Aizawa. An information-theoretic perspective of tf-idf measures. *Information Processing and Management*, 39:45–65, January 2003.

[9] Robin Aly and Djoerd Hiemstra. Concept detectors: How good is good enough? In *Proceedings of the 17th ACM international conference on Multimedia*, MM '09, pages 233–242, New York, NY, USA, 2009. ACM.

[10] Robin Aly, Djoerd Hiemstra, Franciska de Jong, and Peter Apers. Simulating the future of concept-based video retrieval under improved detector performance. *Multimedia Tools and Applications*, pages 1–29, 2011.

[11] Robin Aly, Djoerd Hiemstra, and Arjen de Vries. Reusing annotation labor for concept selection. In *CIVR '09: Proceeding of the ACM International Conference on Image and Video Retrieval*, pages 1–8, New York, NY, USA, 2009. ACM.

[12] Arnon Amir, Marco Berg, Shih-Fu Chang, Giridharan Iyengar, Ching-Yung Lin, Apostol (Paul) Natsev, Chalapathy Neti, Harriet Nock, Milind Naphade, Winston Hsu, John R. Smith, Belle Tseng, Yi Wu, and Donqing Zhang. IBM research TRECVid-2003 video retrieval system. In *NIST TRECid-2003*, 2003.

[13] Daniel Ashbrook and Thad Starner. Using GPS to learn significant locations and predict movement across multiple users. *Personal Ubiquitous Comput.*, 7:275–286, October 2003.

[14] Ricardo Baeza-Yates and Berthier Ribeiro-Neto. *Modern Information Retrieval*. Addison Wesley, 1st edition, May 1999.

[15] Satanjeev Banerjee and Ted Pedersen. An adapted Lesk algorithm for word sense disambiguation using WordNet. In *CICLing '02: Proceedings of the Third International Conference on Computational Linguistics and Intelligent Text Processing*, pages 136–145, London, UK, 2002. Springer-Verlag.

[16] Ling Bao and Stephen S. Intille. Activity recognition from user-annotated acceleration data. *Pervasive Computing*, pages 1–17, 2004.

[17] Deborah Barreau, Abe Crystal, Jane Greenberg, Anuj Sharma, Michael Conway, John Oberlin, Michael Shoffner, and Stephen Seiberling. Augmenting memory for student learning: Designing a context-aware capture system for biology education. *Proceedings of the American Society for Information Science and Technology*, 43(1):251, 2006.

[18] Dave Beckett. RDF/XML syntax specification. *W3C Recommendation*, February 2004.

[19] Dave Beckett. Turtle-Terse RDF triple language. *W3C Technical Report*, November 2007.

[20] Serge Justin Belongie, Jitendra M. Malik, and Jan Puzicha. Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24:509–522, April 2002.

[21] Tim Berners-Lee. Notation 3 specification. *W3C Technical Report*, March 2006.

[22] Tim Berners-Lee, James Hendler, and Ora Lassila. The Semantic Web. *Scientific American*, 284(5):34–43, May 2001.

[23] Emma Berry, Narinder Kapur, Lyndsay Williams, Steve Hodges, Peter Watson, Gavin Smyth, James Srinivasan, Reg Smith, Barbara Wilson, and Ken Wood. The use of a wearable camera, SenseCam, as a pictorial diary to improve autobiographical memory in a patient with limbic encephalitis: A preliminary report. *Neuropsychological Rehabilitation*, 17(4-5):582–601, August 2007.

[24] Alberto Del Bimbo and Pietro Pala. Visual image retrieval by elastic matching of user sketches. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19:121–132, February 1997.

[25] Henk Blanken, Arjen P. de Vries, Henk Ernst Blok, and Ling Feng (Eds.). *Multimedia Retrieval*. Springer, 1st edition, 2007.

[26] Michael Blighe, Sorin Sav, Hyowon Lee, and Noel E. O'Connor. Mo Músaem Fíorúil: A web-based search and information service for museum visitors. In *International Conference on Image Analysis and Recognition*, pages 485–496, 2008.

[27] Mark Blum, Alex (Sandy) Pentland, and Gehrard Tröster. InSense: Interest-based life logging. *Multimedia, IEEE*, 13(4):40 –48, Oct.-Dec. 2006.

[28] Dan Brickley and R. V. Guha. RDF vocabulary description language 1.0: RDF Schema. *W3C Technical Report*, February 2004.

[29] M. G. Brown, Jonathan Trumbull Foote, Gareth J. F. Jones, Koren Sparck Jones, and Stephen J. Young. Automatic content-based retrieval of broadcast news. In *Proceedings of the third ACM international conference on Multimedia*, MULTIMEDIA '95, pages 35–43, New York, NY, USA, 1995. ACM.

[30] Michael Bukhin and Michael DelGaudio. WayMarkr: Acquiring perspective through continuous documentation. In *MUM '06: Proceedings of the 5th international conference on mobile and ubiquitous multimedia*, page 9, New York, NY, USA, 2006. ACM.

[31] Christopher J. C. Burges. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2(2):121–167, 1998.

[32] Vannevar Bush. As we may think. *The Atlantic Monthly*, 1945.

[33] Daragh Byrne, Aiden R. Doherty, Gareth J. F. Jones, Alan F. Smeaton, Sanna Kumpulainen, and Kalervo Järvelin. The SenseCam as a tool for task observation. In *Proceedings of the 22nd British HCI Group Annual Conference on*

People and Computers: Culture, Creativity, Interaction - Volume 2, BCS-HCI '08, pages 19–22, Swinton, UK, 2008. British Computer Society.

[34] Daragh Byrne, Aiden R. Doherty, Cees G. M. Snoek, Gareth J. F. Jones, and Alan F. Smeaton. Everyday concept detection in visual lifelogs: validation, relationships and trends. *Multimedia Tools Appl.*, 49(1):119–144, 2010.

[35] Daragh Byrne and Gareth J.F. Jones. Towards computational autobiographical narratives through human digital memories. In *Proceeding of the 2nd ACM international workshop on story representation, mechanism and context*, SRMC '08, pages 9–12, New York, NY, USA, 2008. ACM.

[36] Daragh Byrne, Barry Lavelle, Aiden R. Doherty, Gareth J. F. Jones, and Alan F. Smeaton. Using Bluetooth and GPS metadata to measure event similarity in SenseCam images. In *The 5th International Conference on Intelligent Multimedia and Ambient Intelligence*, Salt Lake City, Utah, USA, 2007.

[37] Murray Campbell, Alexander Haubold, Shahram Ebadollahi, Dhiraj Joshi, Milind R. Naphade, Apostol (Paul) Natsev, Joachim Seidl, John R. Smith, Katya Scheinberg, Jelena Tesic, and Lexing Xie. IBM research TRECVid-2006 video retrieval system. In *TREC Video Retrieval Evaluation Proceedings*, 2006.

[38] Murray Campbell, Er Haubold, Shahram Ebadollahi, Dhiraj Joshi, and Milind R. Naphade.

[39] Shih-Fu Chang, Winston Hsu, Wei Jiang, Lyndon Kennedy, Dong Xu, Akira Yanagawa, and Eric Zavesky. Evaluating the impact of 374 visualbased LSCOM concept detectors on automatic search. In *Proceedings of the 4th TRECVid Workshop*, Gaithersburg, USA, November 2006.

[40] Rachel Chilvers, Susan Corr, and Singlehurst Hayley. Investigation into the occupational lives of healthy older people through their use of time. *Australian Occupational Therapy Journal*, 57(1):24–33, 2010.

[41] Michael G. Christel and Alexander G. Hauptmann. The use and utility of high-level semantic features in video retrieval. In *CIVR'05: Proceedings of the International Conference on Video Retrieval*, pages 134–144, Dublin, Ireland, 2005.

[42] Michael G. Christel and Alexander G. Hauptmann. Exploring concept selection strategies for interactive video search. In *ICSC '07: Proceedings of the International Conference on Semantic Computing*, pages 344–354, Washington, DC, USA, 2007. IEEE Computer Society.

[43] Michael G. Christel, Milind R. Naphade, Apostol (Paul) Natsev, and Jelena Tesic. Assessing the filtering and browsing utility of automatic semantic concepts for multimedia retrieval. In *CVPRW '06: Proceedings of the 2006 Conference on Computer Vision and Pattern Recognition Workshop*, pages 117–124, Washington, DC, USA, 2006. IEEE Computer Society.

[44] Tat-Seng Chua, Shi-Yong Neo, Ke-Ya Li, Gang Wang, Rui Shi, Ming Zhao, Huaxin Xu, Qi Tian, Sheng Gao, and Tin Lay Nwe. TRECVid 2004 search and feature extraction task by NUS PRIS. In *NIST TRECVid*, 2004.

[45] James Clark. XSL transformations (XSLT). *W3C Recommendation*, November 1999.

[46] Ciarán Ó Conaire, Damien Connaghan, Philip Kelly, Noel E. O'Connor, Mark Gaffney, and John Buckley. Combining inertial and visual sensing for human action recognition in tennis. In *Proceedings of the first ACM international*

workshop on analysis and retrieval of tracked events and motion in imagery streams, ARTEMIS '10, pages 51–56, New York, NY, USA, 2010. ACM.

[47] Ciaran O Connaire, Noel O'Connor, Alan F. Smeaton, and Gareth Jones. Organising a daily visual diary using multi-feature clustering. In *SPIE Electronic Imaging - Multimedia Content Access: Algorithms and Systems*, San Jose, CA, 2007.

[48] Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407, 1990.

[49] Anind K. Dey and Gregory D. Abowd. Towards a better understanding of context and context-awareness. In *HUC'99: Proceedings of the 1st international symposium on Handheld and Ubiquitous Computing*, pages 304–307, London, UK, 1999. Springer-Verlag.

[50] Aiden Doherty and Alan F. Smeaton. Automatically augmenting lifelog events using pervasively generated content from millions of people. *Sensors*, 10(3):1423–1446, 2010.

[51] Aiden R. Doherty, Daragh Byrne, Alan F. Smeaton, Gareth J. F. Jones, and Mark Hughes. Investigating keyframe selection methods in the novel domain of passively captured visual lifelogs. In *CIVR'08: Proceedings of the 2008 international conference on content-based image and video retrieval*, pages 259–268, Niagara Falls, Canada, 2008. ACM.

[52] Aiden R. Doherty, Niamh Caprani, Ciarán Ó Conaire, Vaiva Kalnikaite, Cathal Gurrin, Alan F. Smeaton, and Noel E. O'Connor. Passively recognising human activities through lifelogging. *Computers in Human Behavior*, 27:1948–1958, September 2011.

[53] Aiden R. Doherty and Alan F. Smeaton. Automatically segmenting lifelog data into events. In *WIAMIS '08: Proceedings of the 2008 Ninth International Workshop on Image Analysis for Multimedia Interactive Services*, pages 20–23, Washington, DC, USA, 2008. IEEE Computer Society.

[54] Micah Dubinko, Ravi Kumar, Joseph Magnani, Jasmine Novak, Prabhakar Raghavan, and Andrew Tomkins. Visualizing tags over time. In *WWW '06: Proceedings of the 15th international conference on World Wide Web*, pages 193–202, New York, NY, USA, 2006. ACM.

[55] Nathan Eagle and Alex (Sandy) Pentland. Reality mining: Sensing complex social systems. *Personal Ubiquitous Comput.*, 10(4):255–268, 2006.

[56] Jianping Fan, Ahmed K. Elmagarmid, Xingquan Zhu, Walid G. Aref, and Lide Wu. ClassView: Hierarchical video shot classification, indexing, and accessing. *IEEE Trans. on Multimedia*, 6:70–86, 2004.

[57] Andrew Farmer, Oliver Gibson, Paul Hayton, Kathryn Bryden, Christina Dudley, Andrew Neil, and Lionel Tarassenko. A real-time, mobile phone-based telemedicine system to support young adults with type 1 diabetes. *Informatics in Primary Care*, 13(3):171–178, 2005.

[58] Rowanne Fleck and Geraldine Fitzpatrick. Supporting collaborative reflection with passive image capture. In *Supplementary Proceedings of COOP'06*, pages 41–48, Carry-le-Rouet, France, 2006.

[59] David Frohlich, Allan Kuchinsky, Celine Pering, Abbe Don, and Steven Ariss. Requirements for photoware. In *Proceedings of the 2002 ACM conference on computer supported cooperative work*, CSCW '02, pages 166–175, New York, NY, USA, 2002. ACM.

[60] Shravan Gaonkar, Jack Li, Romit Roy Choudhury, Landon Cox, and Al Schmidt. Micro-Blog: Sharing and querying content through mobile phones and social participation. In *Proceeding of the 6th International Conference on mobile systems, applications, and services*, MobiSys '08, pages 174–186, New York, NY, USA, 2008. ACM.

[61] Jim Gemmell, Aleks Aris, and Roger Lueder. Telling stories with MyLifeBits. In *ICME 2005: IEEE International Conference on Multimedia and Expo, 2005*, pages 1536 –1539, Amsterdam, The Netherlands, July 2005.

[62] Theo Gevers and Arnold W. M. Smeulders. PicToSeek: Combining color and shape invariant features for image retrieval. *IEEE Transactions on Image Processing*, 9(1):102–119, January 2000.

[63] King-Shy Goh, Edward Chang, and Kwang-Ting Cheng. SVM binary classifier ensembles for image classification. In *Proceedings of the tenth International Conference on information and knowledge management*, CIKM '01, pages 395–402, New York, NY, USA, 2001. ACM.

[64] Jan Grant and Dave Beckett. RDF test cases. *W3C Technical Report*, February 2004.

[65] Cathal Gurrin, Daragh Byrne, Noel O'Connor, Gareth Jones, and Alan F. Smeaton. Architecture and challenges of maintaining a large-scale, context-aware human digital memory. In *VIE 2008 - The 5th IET Visual Information Engineering 2008 Conference*, Xi'An, China, 2008.

[66] Zoltán Gyöngyi, Hector Garcia-Molina, and Jan Pedersen. Combating web spam with trustrank. In *Proceedings of the thirtieth international conference on very large data bases*, VLDB '04, pages 576–587. VLDB Endowment, 2004.

[67] Alexander Haubold, Apostol Natsev, and Milind Naphade. Semantic multimedia retrieval using lexical query expansion and model-based reranking. *IEEE International Conference on Multimedia and Expo*, pages 1761–1764, 2006.

[68] Claudia Hauff, Robin Aly, and Djoerd Hiemstra. The effectiveness of concept based search for video retrieval. In *Workshop Information Retrieval (FGIR 2007)*, pages 205–212, Halle-Wittenberg, Germany, 2007.

[69] Alexander Hauptmann, Rong Yan, and Wei-Hao Lin. How many high-level concepts will fill the semantic gap in news video retrieval? In *CIVR '07: Proceedings of the 6th ACM international conference on image and video retrieval*, pages 627–634, New York, NY, USA, 2007. ACM.

[70] Jeffrey Hightower. From position to place. In *Proceedings of the Workshop on Location-Aware Computing*, pages 10–12, 2003.

[71] Jeffrey Hightower, Sunny Consolvo, Anthony Lamarca, Ian Smith, and Jeff Hughes. Learning and recognizing the places we go. In *UbiComp 2005: Ubiquitous Computing*, pages 159–176. Tokyo, Japan, 2005.

[72] Matthew Hill, Gang Hua, Apostol Natsev, John R. Smith, Lexing Xie, Bert Huang, Michele Merler, Hua Ouyang, and Mingyuan Zhou. IBM Rsesearch TRECVid-2010 video copy detection and multimedia event detection system. In *NIST TRECVID Workshop*, 2010.

[73] Graeme Hirst and David St-Onge. Lexical chains as representation of context for the detection and correction malapropisms. *WordNet: An Electronic Lexical Database (Language, Speech, and Communication)*, May 1998.

[74] Steve Hodges, Lyndsay Williams, Emma Berry, Shahram Izadi, James Srinivasan, Alex Butler, Gavin Smyth, Narinder Kapur, and Ken Wood. SenseCam:

A retrospective memory aid. In *Proc. 8th International Conference on Ubicomp*, pages 177–193, Orange County, CA, USA, 2006.

[75] L. Hollink, M. Worring, and A. Th. Schreiber. Building a visual ontology for video retrieval. In *Proceedings of the 13th annual ACM international conference on Multimedia*, MULTIMEDIA '05, pages 479–482, New York, NY, USA, 2005. ACM.

[76] Anthony Hoogs, Jens Rittscher, Gees Stein, and John Schmiederer. Video content annotation using visual analysis and a large semantic knowledgebase. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2:327–334, 2003.

[77] Tetsuro Hori and Kiyoharu Aizawa. Context-based video retrieval system for the life-log applications. In *Proceedings of the 5th ACM SIGMM international workshop on Multimedia information retrieval*, MIR '03, pages 31–38, New York, NY, USA, 2003. ACM.

[78] Bouke Huurnink, Katja Hofmann, and Maarten de Rijke. Assessing concept selection for video retrieval. In *MIR '08: Proceeding of the 1st ACM international conference on Multimedia information retrieval*, pages 459–466, New York, NY, USA, 2008. ACM.

[79] Jay J. Jiang and David W. Conrath. Semantic similarity based on corpus statistics and lexical taxonomy. In *International Conference Research on Computational Linguistics*, pages 19–33, September 1997.

[80] Yu-Gang Jiang, Xiaohong Zeng, Guangna Ye, Subh Bhattacharya, Dan Ellis, Mubarak Shah, and Shih-Fu Chang. Columbia-UCF TRECVid2010 multimedia event detection: Combining multiple modalities, contextual concepts, and temporal matching. In *NIST TRECVID Workshop*, 2010.

[81] Thorsten Joachims. Making large-scale support vector machine learning practical. *Advances in Kernel Methods*, pages 169–184, 1999.

[82] Crystal H. Kaczkowski, Peter J. H. Jones, Jianying Feng, and Henry S. Bayley. Four-day multimedia diet records underestimate energy needs in middle-aged and elderly women as determined by doubly-labeled water. *Journal of Nutrition*, 130(4):802–5, 2000.

[83] Daniel Kahneman, Alan B. Krueger, David A. Schkade, Norbert Schwarz, and Arthur A. Stone. A survey method for characterizing daily life experience: The day reconstruction method. *Science*, 306(5702):1776–1780, 2004.

[84] Svebor Karaman, Jenny Benois-Pineau, Remi Megret, Julien Pinquier, Yann Gaestel, and Jean-Francois Dartigues. Activities of daily living indexing by hierarchical HMM for dementia diagnostics. In *The 9th International Workshop on Content-Based Multimedia Indexing (CBMI)*, pages 79–84, Madrid, Spain, June 2011.

[85] Shuaib Karim, Amin Andjomshoaa, and A. Min Tjoa. Exploiting SenseCam for helping the blind in business negotiations. In *Computers Helping People with Special Needs*, pages 1147–1154. Springer, 2006.

[86] Philip Kelly, Aiden R. Doherty, Alan F. Smeaton, Cathal Gurrin, and Noel E. O'Connor. The colour of life: Novel visualisations of population lifestyles. In *Proceedings of the international conference on Multimedia*, MM '10, pages 1063–1066, New York, NY, USA, 2010. ACM.

[87] Lyndon S. Kennedy, Apostol (Paul) Natsev, and Shih-Fu Chang. Automatic discovery of query-class-dependent models for multimodal search. In *Proceedings of the 13th annual ACM international conference on Multimedia*, MULTIMEDIA '05, pages 882–891, New York, NY, USA, 2005. ACM.

[88] Hee-Seung Kim, Nam-Cho Kim, and Sung-Hee Ahn. Impact of a nurse short message service intervention for patients with diabetes. *Journal of Nursing Care Quality*, 21(3):266–271, 2006.

[89] Graham Klyne and Jeremy J. Carroll. Resource Description Framework (RDF): Concepts and abstract syntax. *W3C Recommendation*, February 2004.

[90] Christoph Kuhmunch. On the detection and recognition of television commercials. In *Proceedings of the 1997 International Conference on Multimedia Computing and Systems*, pages 509–516, Washington, DC, USA, 1997. IEEE Computer Society.

[91] Sanna Kumpulainen, Kalervo Järvelin, Sami Serola, Aiden R. Doherty, Daragh Byrne, Alan F. Smeaton, and Gareth J.F. Jones. Data collection methods for analyzing task-based information access in molecular medicine. In *MobiHealthInf 2009 - 1st International Workshop on Mobilizing Health Information to Support Healthcare-related Knowledge Work*, 2009.

[92] George Lakoff. *Women, Fire, and Dangerous Things.* University of Chicago Press, April 1990.

[93] Thomas K. Landauer, Peter W. Foltz, and Darrell Laham. An introduction to latent semantic analysis. *Discourse Processes*, (25):259–284, 1998.

[94] Mary Law, Sandy Steinwender, and Leanne Leclair. Occupation, health and well-being. *Canadian Journal of Occupational Therapy*, 65(2):81–91, 1998.

[95] Claudia Leacock and Martin Chodorow. Combining local context and WordNet similarity for word sense identification. *An Electronic Lexical Database*, pages 265–283, 1998.

[96] Hyowon Lee, Alan F. Smeaton, Noel E. O'Connor, Gareth J. F. Jones, Michael Blighe, Daragh Byrne, Aiden R. Doherty, and Cathal Gurrin. Constructing a SenseCam visual diary as a media process. *Multimedia Syst.*, 14(6):341–349, 2008.

[97] Beitao Li and Kingshy Goh. Confidence-based dynamic ensemble for image annotation and semantics discovery. In *MULTIMEDIA '03: Proceedings of the eleventh ACM international conference on Multimedia*, pages 195–206, New York, NY, USA, 2003. ACM.

[98] Jia Li and James Z. Wang. Automatic linguistic indexing of pictures by a statistical modeling approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25:1075–1088, September 2003.

[99] Xirong Li, Dong Wang, Jianmin Li, and Bo Zhang. Video search in concept subspace: A text-like paradigm. In *Proceedings of the 6th ACM international conference on Image and video retrieval*, CIVR '07, pages 603–610, New York, NY, USA, 2007. ACM.

[100] Dekang Lin. An information-theoretic definition of similarity. In *Proceedings of the 15th International Conference on Machine Learning*, pages 296–304. Morgan Kaufmann, 1998.

[101] Wei-Hao Lin and Alexander G. Hauptmann. Which thousand words are worth a picture? Experiments on video retrieval using a thousand concepts. In *IEEE International Conference on Multimedia and Expo*, pages 41–44, Los Alamitos, CA, USA, 2006. IEEE Computer Society.

[102] Hugo Liu and Push Singh. Commonsense reasoning in and over natural language. In *Proceedings of the 8th International Conference on Knowledge-Based*

Intelligent Information and Engineering Systems, Wellington, New Zealand, 2004. Springer.

[103] Hugo Liu and Push Singh. ConceptNet – a practical commonsense reasoning tool-kit. *BT Technology Journal*, 22:211–226, October 2004.

[104] Hangzai Luo and Jianping Fan. Building concept ontology for medical video annotation. In *Proceedings of the 14th annual ACM international conference on Multimedia*, MULTIMEDIA '06, pages 57–60, New York, NY, USA, 2006. ACM.

[105] Wei Y. Ma and B. S. Manjunath. NeTra: a toolbox for navigating large image databases. *Multimedia Systems*, 7(3):184–198, May 1999.

[106] Bangalore S. Manjunath and Wei-Ying Ma. Texture features for browsing and retrieval of image data. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18:837–842, August 1996.

[107] Bangalore S. Manjunath, Jens-Rainer Ohm, Vinod V. Vasudevan, and Akio Yamada. Color and texture descriptors. *IEEE Transactions on Circuits and Systems for Video Technology*, 11(6):703–715, 2001.

[108] Steve Mann. Wearable computing: A first step toward personal imaging. *Computer*, 30(2):25–32, 1997.

[109] Steve Mann. 'WearCam' (the wearable camera): Personal imaging systems for long–term use in wearable tetherless computer–mediated reality and personal photo/videographic memory prosthesis. In *Proceedings of the 2nd IEEE International Symposium on Wearable Computers*, ISWC '98, pages 124–131, Washington, DC, USA, 1998. IEEE Computer Society.

[110] Steve Mann, James Fung, Chris Aimone, Anurag Sehgal, and Daniel Chen. Designing EyeTap digital eyeglasses for continuous lifelong capture and sharing of personal experiences. In *Proc. CHI 2005 Conference on Computer Human Interaction*, Portland, Oregon, USA, 2005. ACM Press.

[111] Kryss McKenna, Kieran Broome, and Jacki Liddle. What older people do: Time use and exploring the link between role participation and life satisfaction in people aged 65 years and over. *Australian Occupational Therapy Journal*, 54(4):273–284, 2007.

[112] Rémi Mégret, Vladislavs Dovgalecs, Hazem Wannous, Svebor Karaman, Jenny Benois-Pineau, Elie El Khoury, Julien Pinquier, Philippe Joly, Régine André-Obrecht, Yann Gaëstel, and Jean-François Dartigues. The IMMED project: wearable video monitoring of people with age dementia. In *Proceedings of the international conference on Multimedia*, MM '10, pages 1299–1302, New York, NY, USA, 2010. ACM.

[113] Rémi Mégret, Daniel Szolgay, Jenny Benois-Pineau, Philippe Joly, Julien Pinquier, Jean-François Dartigues, and Catherine Helmer. Wearable video monitoring of people with age dementia : Video indexing at the service of healthcare. In *International Workshop on Content-Based Multimedia Indexing*, CBMI '08, pages 101–108, London, UK, 2008.

[114] Guus Schreiber Mike Dean. OWL Web ontology language reference. *W3C Recommendation*, February 2004.

[115] George A. Miller. WordNet: a lexical database for english. *Communications of the ACM*, 38(11):39–41, 1995.

[116] PlaceLab (MIT). http://architecture.mit.edu/house_n/placelab.html. Last accessed: Aug. 2011.

[117] Nicolas Moënne-Loccoz, Bruno Janvier, Stéphane Marchand-Maillet, and Eric Bruno. Managing video collections at large. In *CVDB '04: Proceedings of the 1st international workshop on Computer vision meets databases*, pages 59–66, New York, NY, USA, 2004. ACM.

[118] Mor Naaman, Yee Jiun Song, Andreas Paepcke, and Hector Garcia-Molina. Automatic organization for digital photographs with geographic coordinates. In *Proceedings of the 4th ACM/IEEE-CS joint conference on Digital libraries*, JCDL '04, pages 53–62, New York, NY, USA, 2004. ACM.

[119] Milind Naphade, John R. Smith, Jelena Tesic, Shih-Fu Chang, Winston Hsu, Lyndon Kennedy, Alexander Hauptmann, and Jon Curtis. Large-scale concept ontology for multimedia. *IEEE Multimedia*, 13(3):86–91, July 2006.

[120] Milind R. Naphade and Thomas S. Huang. A probabilistic framework for semantic video indexing, filtering, and retrieval. *IEEE Transactions on Multimedia*, 3:141–151, 2001.

[121] Milind R. Naphade, Lyndon Kennedy, John R. Kender, Shih-Fu Chang, John R. Smith, Paul Over, and Alex Hauptmann. A light scale concept ontology for multimedia understanding for TRECVid 2005. Technical report, IBM Research, 2005.

[122] Apostol (Paul) Natsev, Alexander Haubold, Jelena Tesic, Lexing Xie, and Rong Yan. Semantic concept-based query expansion and re-ranking for multimedia retrieval. In *MULTIMEDIA '07: Proceedings of the 15th international conference on Multimedia*, pages 991–1000, New York, NY, USA, 2007. ACM.

[123] Apostol (Paul) Natsev, Milind R. Naphade, and Jelena Tesic. Learning the semantics of multimedia queries and concepts from a small number of examples.

In *Proceedings of the 13th annual ACM international conference on Multimedia*, MULTIMEDIA '05, pages 598–607, New York, NY, USA, 2005. ACM.

[124] Shi-Yong Neo, Jin Zhao, Min-Yen Kan, and Tat-Seng Chua. Video retrieval using high level features: Exploiting query matching and confidence-based weighting. In *CIVR'06: Proc. Conf. Image and Video Retrieval*, pages 143–152. Springer-Verlag, Tempe, AZ, USA, 2006.

[125] Paul Over, Tzveta Ianeva, Wessel Kraaij, and Alan F. Smeaton. TRECVid 2005 – an overview. In *Proceedings of TRECVid 2005*, 2005.

[126] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The PageRank citation ranking: Bringing order to the web. Technical report, Stanford Digital Library Technologies Project, 1998.

[127] Donald J. Patterson, Lin Liao, Krzysztof Gajos, Michael Collier, Nik Livic, Katherine Olson, Shiaokai Wang, Dieter Fox, and Henry Kautz. Opportunity Knocks: a system to provide cognitive assistance with transportation services. In *International Conference on Ubiquitous Computing (UbiComp)*, pages 433–450, Heidelberg, 2004. Springer.

[128] Siddharth Patwardhan, Satanjeev Banerjee, and Ted Pedersen. Using measures of semantic relatedness for word sense disambiguation. In *CICLing'03: Proceedings of the 4th international conference on Computational Linguistics and Intelligent Text processing*, pages 241–257, Berlin, Heidelberg, 2003. Springer-Verlag.

[129] Giuseppe Pirró and Jérôme Euzenat. A feature and information theoretic framework for semantic similarity and relatedness. In *Proceedings of the 9th international semantic web conference on the semantic web*, ISWC'10, pages 615–630, Berlin, Heidelberg, 2010. Springer-Verlag.

[130] John C. Platt. AutoAlbum: Clustering digital photographs using probabilistic model merging. In *Proceedings of the IEEE workshop on Content-based access of image and video libraries*, CBAIVL '00, pages 96–100, Washington, DC, USA, 2000. IEEE Computer Society.

[131] Jay M. Ponte and W. Bruce Croft. A language modeling approach to information retrieval. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '98, pages 275–281, New York, NY, USA, 1998. ACM.

[132] Eric Prud'hommeaux and Andy Seaborne. SPARQL query language for RDF. *W3C Recommendation*, January 2008.

[133] Zhengwei Qiu, Aiden R. Doherty, Cathal Gurrin, and Alan F. Smeaton. Mining user activity as a context source for search and retrieval. In *STAIR'11: International Conference on Semantic Technology and Information Retrieval*, Kuala Lumpur, Malaysia, June 2011.

[134] Lawrence R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. In *Readings in speech recognition*, pages 267–296. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1990.

[135] Roy Rada, Hafedh Mili, Ellen Bicknell, and Maria Blettner. Development and application of a metric on semantic nets. *IEEE Transactions on Systems, Man and Cybernetics*, 19(1):17–30, 1989.

[136] Sasank Reddy, Andrew Parker, Josh Hyman, Jeff Burke, Deborah Estrin, and Mark Hansen. Image browsing, processing, and clustering for participatory sensing: lessons from a dietsense prototype. In *EmNets'07: Proceedings of the 4th workshop on Embedded networked sensors*, pages 13–17, Cork, Ireland, 2007. ACM Press.

[137] Philip Resnik. Using information content to evaluate semantic similarity in a taxonomy. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence*, pages 448–453, San Francisco, CA, USA, 1995. Morgan Kaufmann Publishers Inc.

[138] Philip Resnik. Semantic similarity in a taxonomy: An information-based measure and its application to problems of ambiguity in natural language. *Journal of Artificial Intelligence Research*, 11:95–130, 1999.

[139] Ray Richardson and Alan F. Smeaton. Using WordNet in a knowledge-based approach to information retrieval. Technical Report CA-0395, Dublin City University, 1995.

[140] Kerry Rodden. How do people organise their photographs? In *Proceedings of the BCS IRSG Colloquium*, 1999.

[141] Kerry Rodden and Kenneth R. Wood. How do people manage their digital photographs? In *Proceedings of the SIGCHI conference on Human factors in computing systems*, CHI '03, pages 409–416, New York, NY, USA, 2003. ACM.

[142] Yong Rui, Anoop Gupta, and Alex Acero. Automatically extracting highlights for TV baseball programs. In *Proceedings of the eighth ACM international conference on Multimedia*, MULTIMEDIA '00, pages 105–115, New York, NY, USA, 2000. ACM.

[143] Yong Rui, Thomas S. Huang, and Shih-Fu Chang. Image retrieval: current techniques, promising directions and open issues. *Journal of Visual Communication and Image Representation*, 10(4):39–62, April 1999.

[144] Marçal Rusinol, David Aldavert, Dimosthenis Karatzas, Ricardo Toledo, and Josep Lladós. Interactive trademark image retrieval by fusing semantic and

visual content. In *Proceedings of the 33rd European conference on Advances in information retrieval*, ECIR'11, pages 314–325, Berlin, Heidelberg, 2011. Springer-Verlag.

[145] Gerard Salton. Dynamic document processing. *Commun. ACM*, 15:658–668, July 1972.

[146] Gerard M Salton, Andrew K C Wong, and C. S. Yang. A vector space model for automatic indexing. *Commun. ACM*, 18:613–620, November 1975.

[147] Nuno Seco, Tony Veale, and Jer Hayes. An intrinsic information content metric for semantic similarity in WordNet. In *ECAI'2004, the 16th European Conference on Artificial Intelligence*, Valencia, Spain, 2004.

[148] Abigail Sellen, Andrew Fogg, Mike Aitken, Steve Hodges, Carsten Rother, and Ken Wood. Do life-logging technologies support memory for the past? An experimental study using SenseCam. In *Proc. CHI 2007, ACM Press*, pages 81–90, New York, NY, USA, 2007.

[149] Bageshree Shevade, Hari Sundaram, and Min-Yen Kan. A collaborative annotation framework. In *Proc. International Conference on Multimedia and Expo 2005*, pages 1346–1349, Amsterdam, 2005.

[150] Ilmerio R. Silva, Joao Nunes Souza, and Karina S. Santos. Dependence among terms in vector space model. In *Proceedings of the International Database Engineering and Applications Symposium*, pages 97–102, Washington, DC, USA, 2004. IEEE Computer Society.

[151] Alan F. Smeaton, Paul Over, and Wessel Kraaij. Evaluation campaigns and TRECVid. In *MIR '06: Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval*, pages 321–330, New York, NY, USA, 2006. ACM Press.

[152] Alan F. Smeaton, Paul Over, and Wessel Kraaij. High-level feature detection from video in TRECVid: a 5-year retrospective of achievements. In Ajay Divakaran, editor, *Multimedia Content Analysis, Theory and Applications*, pages 151–174. Springer Verlag, Berlin, 2009.

[153] Alan F. Smeaton and Ian Quigley. Experiments on using semantic distances between words in image caption retrieval. In *Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '96, pages 174–180, New York, NY, USA, 1996. ACM.

[154] Arnold W. M. Smeulders, Marcel Worring, Simone Santini, Amarnath Gupta, and Ramesh Jain. Content-based image retrieval at the end of the early years. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22:1349–1380, December 2000.

[155] John R. Smith and Shih-Fu Chang. Visually searching the web for content. *IEEE MultiMedia*, 4:12–20, July 1997.

[156] Cees G. M. Snoek, Bouke Huurnink, Laura Hollink, Maarten De Rijke, Guus Schreiber, and Marcel Worring. Adding semantics to detectors for video retrieval. *IEEE Transactions on Multimedia*, 9(5):975–986, 2007.

[157] Cees G. M. Snoek, Jan C. van Gemert, Theo Gevers, Bouke Huurnink, Dennis C. Koelma, Michiel van Liempt, Ork de Rooij, Koen E. A. van de Sande, Frank J. Seinstra, Arnold W. M. Smeulders, Andrew H. C. Thean, Cor J. Veenman, and Marcel Worring. The MediaMill TRECVid 2006 semantic video search engine. In *Proceedings of the 4th TRECVid Workshop*, Gaithersburg, USA, November 2006.

[158] Cees G. M. Snoek, Marcel Worring, Jan M. Geusebroek, Dennis C. Koelma, Frank J. Seinstra, and Arnold W. M. Smeulders. The semantic pathfinder:

Using an authoring metaphor for generic multimedia indexing. *IEEE Trans. Pattern Anal. Mach. Intell.*, 28(10):1678–1689, 2006.

[159] Cees G. M. Snoek, Marcel Worring, Jan C. van Gemert, Jan-Mark Geusebroek, and Arnold W. M. Smeulders. The challenge problem for automated detection of 101 semantic concepts in multimedia. In *Proceedings of the 14th annual ACM international conference on Multimedia*, MULTIMEDIA '06, pages 421–430, New York, NY, USA, 2006. ACM.

[160] Evaggelos Spyrou, Hervé Le Borgne, Theofilos Mailis, Eddie Cooke, Yannis Avrithis, and Noel O'Connor. Fusing MPEG-7 visual descriptors for image classification. In *Proceedings of the 15th international conference on Artificial neural networks: formal models and their applications*, ICANN'05, pages 847–852, Berlin, Heidelberg, 2005. Springer-Verlag.

[161] Rudi Studer, V. Richard Benjamins, and Dieter Fensel. Knowledge engineering: Principles and methods. *Data and Knowledge Engineering*, 25(1-2):161–197, March 1998.

[162] Jan C. van Gemert, Jan-Mark Geusebroek, Cor J. Veenman, Cees G. M. Snoek, and Arnold W. M. Smeulders. Robust scene categorization by learning image statistics in context. In *Proceedings of the 2006 Conference on Computer Vision and Pattern Recognition Workshop*, CVPRW '06, pages 105–112, Washington, DC, USA, 2006. IEEE Computer Society.

[163] Vladimir N. Vapnik. *Statistical Learning Theory.* Wiley-Interscience, 1989.

[164] Sunil Vemuri and Walter Bender. Next-generation personal memory aids. *BT Technology Journal*, 22(4):125–138, 2004.

[165] Sunil Vemuri, Chris Schmandt, Walter Bender, Stefanie Tellex, and Brad Lassey. An audio-based personal memory aid. In *Ubicomp*, pages 400–417, 2004.

[166] Da-Hong Wang, Michiko Kogashiwa, and Shohei Kira. Development of a new instrument for evaluating individuals' dietary intakes. *Journal of the American Dietetic Association*, 106(10):1588 – 1593, 2006.

[167] Huan Wang, Song Liu, and Liang-Tien Chia. Does ontology help in image retrieval?: a comparison between keyword, text ontology and multi-modality ontology approaches. In *Proceedings of the 14th annual ACM international conference on Multimedia*, MULTIMEDIA '06, pages 109–112, New York, NY, USA, 2006. ACM.

[168] Peng Wang and Alan F. Smeaton. Aggregating semantic concepts for event representation in lifelogging. In *Proceedings of the International Workshop on Semantic Web Information Management*, SWIM '11, pages 8:1–8:6, New York, NY, USA, 2011. ACM.

[169] Xiao-Yong Wei and Chong-Wah Ngo. Ontology-enriched semantic space for video search. In *MULTIMEDIA '07: Proceedings of the 15th international conference on Multimedia*, pages 981–990, New York, NY, USA, 2007. ACM.

[170] Utz Westermann and Ramesh Jain. Toward a common event model for multimedia applications. *IEEE MultiMedia*, 14:19–29, January 2007.

[171] Thijs Westerveld, Arjen P. de Vries, Alex van Ballegooij, Franciska de Jong, and Djoerd Hiemstra. A probabilistic multimedia retrieval model and its evaluation. *EURASIP Journal on Applied Signal Processing*, 2003(2):186–198, 2003.

[172] Ken Wood, Rowanne Fleck, and Lyndsay Williams. Playing with SenseCam. In *Proc. Playing with Sensors (W3) at UbiComp 2004*, Nottingham, UK, 2004.

[173] Yi Wu, Belle L. T Tseng, and John R. Smith. Ontology-based multi-classification learning for video concept detection. In *ICME'04: IEEE International Conference on Multimedia and Expo*, volume 2, pages 1003–1006, June 2004.

[174] Zhibiao Wu and Martha Palmer. Verb semantics and lexical selection. In *Proceedings of the 32nd annual meeting on Association for Computational Linguistics*, pages 133 –138, Stroudsburg, PA, USA, 1994.

[175] Lexing Xie, Hari Sundaram, and Murray Campbell. Event mining in multimedia streams. *Proceedings of the IEEE*, 96(4):623–647, April 2008.

[176] Rong Yan, Jun Yang, and Alexander G. Hauptmann. Learning query-class dependent weights in automatic video retrieval. In *Proceedings of the 12th annual ACM international conference on Multimedia*, MULTIMEDIA '04, pages 548–555, New York, NY, USA, 2004. ACM.

[177] WeiQi Yan, Declan Kieran, Setareh Rafatirad, and Ramesh Jain. A comprehensive study of visual event computing. *Multimedia Tools and Applications*, 55:443–481, 2011.

[178] Hui Yang, Tat-Seng Chua, Shuguang Wang, and Chun-Keat Koh. Structured use of external knowledge for event-based open domain question answering. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, SIGIR '03, pages 33–40, New York, NY, USA, 2003. ACM.

[179] Meng Yang, Barbara M. Wildemuth, and Gary Marchionini. The relative effectiveness of concept-based versus content-based video retrieval. In *MULTIMEDIA '04: Proceedings of the 12th annual ACM international conference on Multimedia*, pages 368–371, New York, NY, USA, 2004. ACM.

[180] J. M. Zacks, T. S. Braver, M. A. Sheridan, D. I. Donaldson, A. Z. Snyder, J. M. Ollinger, R. L. Buckner, and M. E. Raichle. Human brain activity time-locked to perceptual event boundaries. *Nature Neuroscience*, 4(6):651–655, 2001.

[181] Amit Zunjarwad, Hari Sundaram, and Lexing Xie. Contextual wisdom: social relations and correlations for multimedia event annotation. In *Proceedings of the 15th international conference on Multimedia*, MULTIMEDIA '07, pages 615–624, New York, NY, USA, 2007. ACM.

Appendix A

TRECVid Topic Set

ID	Topic Description
0149	shots of Condoleeza Rice
0150	shots of Iyad Allawi, the former prime minister of Iraq
0151	shots of Omar Karami, the former prime minister of Lebannon
0152	shots of Hu Jintao, president of the People's Republic of China
0153	shots of Tony Blair
0154	shots of Mahmoud Abbas, also known as Abu Mazen, prime minister of the Palestinian Authority
0155	shots of a graphic map of Iraq, location of Bagdhad marked - not a weather map,
0156	shots of tennis players on the court - both players visible at same time
0157	shots of people shaking hands
0158	shots of a helicopter in flight
0159	shots of George W. Bush entering or leaving a vehicle (e.g., car, van, airplane, helicopter, etc) (he and vehicle both visible at the same time)
0160	shots of something (e.g., vehicle, aircraft, building, etc) on fire with flames and smoke visible
0161	shots of people with banners or signs
0162	shots of one or more people entering or leaving a building
0163	shots of a meeting with a large table and more than two people
0164	shots of a ship or boat
0165	shots of basketball players on the court
0166	shots of one or more palm trees

ID	Topic Description
0167	shots of an airplane taking off
0168	shots of a road with one or more cars
0169	shots of one or more tanks or other military vehicles
0170	shots of a tall building (with more than 5 floors above the ground)
0171	shots of a goal being made in a soccer match
0172	shots of an office setting, i.e., one or more desks/tables and one or more computers and one or more people
0173	shots with one or more emergency vehicles in motion (e.g., ambulance, police car, fire truck, etc.)
0174	shots with a view of one or more tall buildings (more than 4 stories) and the top story visible
0175	shots with one or more people leaving or entering a vehicle
0176	shots with one or more soldiers, police, or guards escorting a prisoner
0177	shots of a daytime demonstration or protest with at least part of one building visible
0178	shots of US Vice President Dick Cheney
0179	shots of Saddam Hussein with at least one other person's face at least partially visible
0180	shots of multiple people in uniform and in formation
0181	shots of US President George W. Bush, Jr. walking
0182	shots of one or more soldiers or police with one or more weapons and military vehicles
0183	shots of water with one or more boats or ships
0184	shots of one or more people seated at a computer with display visible
0185	shots of one or more people reading a newspaper
0186	shots of a natural scene - with, for example, fields, trees, sky, lake, mountain, rocks, rivers, beach, ocean, grass, sunset, waterfall, animals, or people; but no buildings, no roads, no vehicles
0187	shots of one or more helicopters in flight
0188	shots of something burning with flames visible
0189	shots of a group including least four people dressed in suits, seated, and with at least one flag
0190	shots of at least one person and at least 10 books
0191	shots containing at least one adult person and at least one child
0192	shots of a greeting by at least one kiss on the cheek

Appendix A

ID	Topic Description
0193	shots of one or more smokestacks, chimneys, or cooling towers with smoke or vapor coming out
0194	shots of Condoleeza Rice
0195	shots of one or more soccer goalposts
0196	shots of scenes with snow
0197	shots of one or more people walking up stairs
0198	shots of a door being opened
0199	shots of a person walking or riding a bicycle
0200	shots of hands at a keyboard typing or using a mouse
0201	shots of a canal, river, or stream with some of both banks visible
0202	shots of a person talking on a telephone
0203	shots of a street market scene
0204	shots of a street protest or parade
0205	shots of a train in motion
0206	shots with hills or mountains visible
0207	shots of waterfront with water and buildings
0208	shots of a street at night
0209	shots with 3 or more people sitting at a table
0210	shots with one or more people walking with one or more dogs
0211	shots with sheep or goats
0212	shots in which a boat moves past
0213	shots of a woman talking toward the camera in an interview - no other people visible
0214	shots of a very large crowd of people (fills more than half of field of view)
0215	shots of a classroom scene with one or more students
0216	shots of a bridge
0217	shots of a road taken from a moving vehicle through the front windshield
0218	shots of one or more people playing musical instruments such as drums, guitar, flute, keyboard, piano, etc.
0219	shots that contain the Cook character in the Klokhuis series
0220	grayscale shots of a street with one or more buildings and one or more people
8001	Military formations engaged in tactical warfare, or part of a parade

Appendix A

ID	Topic Description
8002	Government or Civilian leaders at various locations such as press conference, indoors, outdoors, meeting other leaders, addressing crowds, rallies, in parliament or legislative buildings, at photo opportunities etc.
8004	Crowds protesting on streets in urban or rural backgrounds with or without posters/banners etc.
8006	Funeral Procession or Scenes from a funeral or from a cemetery/crematorium/burial site with participants chanting slogans, and/or armed militia and/or masked people present, people carrying pictures of the dead
8007	People on street expressing sorrow by crying, beating their chests, chanting
8008	Military vehicles or helicopters
8011	Police firing weapons
8012	People touching a coffin
8016	Armed guards at checkpoints with barricade on roads
8017	Injured or dead people lying on the ground in any location such as in front of a mosque, on a street, in open grounds, in water etc
8018	Presidential Candidates
8019	Vice-presidential Candidates
8020	Indoor Debate with Speakers at Podium
8021	Town-hall Style Gathering
8022	U.S. Maps depicting the electoral vote distribution (blue vs. red state)
8027	Indoor scene with speaker addressing audience waving flags and cheering
8029	Person greeting people or crowd
8030	Two people on stage in a debate
8031	People posing for pictures with cameras flashing
8034	Soldier sniping at target
8036	Armed men on the city streets
8039	Tanks rolling on streets
8040	Tanks rolling in desert
8041	Armed uniformed soldiers walking on city lanes
8047	Cars burning on city streets or in the desert. May also have overturned cars by the side of roads
8052	Person People not in uniform firing weapons

Appendix A

ID	Topic Description
8053	Armed Soldiers firing weapons
8059	Person or people not in uniform, firing weapon and hiding behind wall of house or building
8067	Battles/Violence in Mountains
8070	Armored Vehicles driving through barren landscapes
8074	Refugee Camps with women and children visible
8079	Convoy of several vehicles on makeshift roads
8080	Empty Streets with buildings in state of dilapidation
8087	Man firing shoulder fired missile in air
8091	Armed Guards standing outside large buildings
8093	Protests turning violent with people throwing missiles, burning objects and clashing with armed military personnel
8094	Military personnel standing guard with shields
8099	Military meeting in an indoor setting with flag visible
8100	Vehicles with flags passing on streets
8101	An open air rally with a high podium and people attending
8103	Rebels with guns on streets or in jeeps
8107	People on the streets being interviewed by a reporter speaking into a microphone
8109	Scenes of battle between rebels and military in urban setting
8114	Dead uniformed soldiers
8119	Destroyed aircrafts and helicopters
8121	Demonstrators marching on streets with banners and signs against military brutality
8125	Clashes of demonstrators with police with police using teargas shells and water guns to push people back
8127	Violence on the streets with crowd pelting stones at police while running away from the advancing police
8128	Rebels brandishing and firing weapons in the air
8131	Heart wrenching scenes of people who have become extremely weak due to absence of adequate food, and water

Appendix B

EventCube Ontology (in Turtle)

```
#Application ontology for EventCube, formatted in Turtle

@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix owl: <http://www.w3.org/2002/07/owl#> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix time: <http://www.w3.org/2006/time#> .
@prefix xml: <http://www.w3.org/XML/1998/namespace> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix geo: <http://www.w3.org/2003/01/geo/wgs84_pos#> .
@prefix : <http://www.clarity-centre.org/EventCube#> .

###    Annotation properties

:hasAffiliation rdf:type owl:AnnotationProperty .
:hasAnnotation rdf:type owl:AnnotationProperty ;
               rdfs:domain :Event ,
                           :Image ;
               rdfs:subPropertyOf rdfs:comment .
```

```
###   Object Properties

:beginAt rdf:type owl:ObjectProperty ;
         rdfs:domain :Event ;
         rdfs:range time:Instant .

# Used for image time
:captureAt rdf:type owl:ObjectProperty ;
           rdfs:domain :Image ;
           rdfs:range time:Instant .

:endAt rdf:type owl:ObjectProperty ;
       rdfs:domain :Event ;
       rdfs:range time:Instant .

# Associates an event with actor and attendee
# Both represent people and from Facebook/FOAF profiles
:hasActor rdf:type owl:ObjectProperty ;
          rdfs:domain :Event ;
          rdfs:range foaf:Person .
:hasAttendee rdf:type owl:ObjectProperty ;
          rdfs:domain :Event ;
          rdfs:range foaf:Person .

:hasAffiliation rdf:type owl:ObjectProperty ;
          rdfs:range foaf:Organiztion ;
          rdfs:domain foaf:Person .

# Associates an event with images
:hasImage rdf:type owl:ObjectProperty ;
          rdfs:domain :Event ;
          rdfs:range :Image ;
          rdfs:subPropertyOf foaf:depiction .

# Associates an event/image with a location
:hasLocation rdf:type owl:ObjectProperty ;
          rdfs:domain :Event ,
                      :Image ;
          rdfs:range :Location .

###   Data properties

:city rdf:type owl:DatatypeProperty ;
          rdfs:domain :Location .

:country rdf:type owl:DatatypeProperty ;
          rdfs:domain :Location .

# Add annotations to event or image if necessary
:hasAnnotation rdf:type owl:DatatypeProperty .

# This field is provided for email
:hasEmailAddress rdf:type owl:DatatypeProperty ;
           rdfs:range xsd:string ;
           rdfs:domain foaf:Person .

#  This predicate is a string which describes the home town
:isFrom rdf:type owl:DatatypeProperty ;
           rdfs:range xsd:string ;
           rdfs:domain foaf:Person .

###   Classes
```

```
# An event has a start time, an end time and a location
:Event rdf:type owl:Class ;
       rdfs:subClassOf time:TemporalEntity ,
           [ rdf:type owl:Restriction ;
             owl:onProperty :hasLocation ;
             owl:minCardinality "1"^^xsd:nonNegativeInteger
           ] ,
           [ rdf:type owl:Restriction ;
             owl:onProperty :endAt ;
             owl:cardinality "1"^^xsd:nonNegativeInteger
           ] ,
           [ rdf:type owl:Restriction ;
             owl:onProperty :beginAt ;
             owl:cardinality "1"^^xsd:nonNegativeInteger
           ] .

# Lifelogging image
:Image rdf:type owl:Class ;
            owl:equivalentClass foaf:Image .

# A location is a point where an event takes place
:Location rdf:type owl:Class ;
            rdfs:subClassOf geo:Point .
```

Appendix C

Event Enhancement Record

Event ID	Location Enhancement	Bluetooth Name
61	Dublin City University	Kirsty-lvs-davexXx
	Glasnevin	SGH-S400i
	Baile Átha Cliath	SGH-X680
62	Mceniff Skylon	Annie tigers
	Dublin Skylon Hotel	W995
	Croke Park	S5230
63	The Westin Dublin	JUN
	Book of Kells	K750i
	Westin	Science Gallery Workshop MacBook (5)
64	Trinity College	JUN
	Merrion Square	
	Leinster House	
65	Trinity College	JUN
	Merrion Square	Jose
	Leinster House	
66	Trinity College	JUN
	Merrion Square	McGrovern
	Leinster House	
67	Trinity College	JUN
	Merrion Square	Zordon
	Leinster House	SGH-ZV60
68	Trinity College	JUN
	The Westin Dublin	Nokia 6300

Event ID	Location Enhancement	Bluetooth Name
	Book of Kells	
69	Trinity College	JUN
	The Westin Dublin	K800i
	Book of Kells	
70	Trinity College	JUN
	The Westin Dublin	Nokia 5800 XpressMusic
	Book of Kells	LG KU990i
71	Trinity College	JUN
	The Westin Dublin	SGH-D908i
	Book of Kells	Nokia 5800 XpressMusic
72	Trinity College	Nokia 2630
	The Westin Dublin	Lar good lookin
	Book of Kells	JUN
73	Trinity College	Science Gallery Workshop MacBook (37)
	Merrion Square	Ciaran Fowley's Computer
	Leinster House	ScienceGalleryWorkshop
74	Trinity College	JUN
	Merrion Square	BT-GPS-37E394
	Leinster House	SGH-D908i
75	Trinity College	SGH-D908i
	Merrion Square	JUN
	Leinster House	Nokia 6230i
76	Trinity College	JUN
	Merrion Square	K750i
	Leinster House	SGH-D908i
77	Trinity College	JUN
	Merrion Square	K750i
	Leinster House	SGH-D908i
78	Trinity College	Nokia 6230i
	Merrion Square	JUN
	Leinster House	K750i
79	Trinity College	K750i
	Merrion Square	JUN
	Leinster House	Nokia 6230i
80	Trinity College	JUN
	Merrion Square	K750i

Event ID	Location Enhancement	Bluetooth Name
	Leinster House	Nokia 6230i
81	Trinity College	JUN
	Merrion Square	Nokia 6230i
	Leinster House	Nif
82	Trinity College	Nif
	Merrion Square	Ciaran Fowley's Computer
	Leinster House	Nokia 6230i
83	Merrion Square	Nif
	Leinster House	Ciaran Fowley's Computer
	Arlington Hotel	S5230
84	Merrion Square	Ciaran Fowley's Computer
	Leinster House	Sparks.Mobile
	Arlington Hotel	
85	Merrion Square	Ciaran Fowley's Computer
	Leinster House	Moomoo
	Arlington Hotel	
86	Merrion Square	Ciaran Fowley's Computer
	Leinster House	Bananaphone
	Arlington Hotel	CHUBBS
87	Merrion Square	Greener
	Leinster House	Bananaphone
	Arlington Hotel	Mine
88	Merrion Square	Bananaphone
	Leinster House	Victorios B.I.G
	Arlington Hotel	Greener
89	Merrion Square	Bananaphone
	Leinster House	
	Arlington Hotel	
90	Academy	Frederick Walter West
	The Spire	ZORAN:@MARKOSKI
	Lynams Hotel	SGH-J700I
91	Dublin City University	
	Glasnevin	
239	Dublin City University	Daragh Byrne's 24inch iMac
	Glasnevin	NeilOHare-MacBook
		cdvpminiColum

Event ID	Location Enhancement	Bluetooth Name
240	Dublin City University	Alan Smeaton’s MacBook Pro
	Glasnevin	Daragh Byrne’s 24inch iMac
		cdvpmini-AlansOffice
241	Dublin City University	Alan Smeaton’s MacBook Pro
	Glasnevin	Pete
		Dermot Diamond’s Computer
242	Dublin City University	Daragh Byrne’s 24inch iMac
	Glasnevin	cdvpminiColum
		NeilOHare-MacBook
243	Dublin City University	Daragh Byrne’s 24inch iMac
	Glasnevin	cdvpminiColum
		Jiang
244	Dublin City University	Nokia 7373
	Glasnevin	
245	Dublin City University	Nokia 7373
	Glasnevin	Deco
246	Dublin City University	Nokia 7373
	Glasnevin	
247	Dublin City University	Nokia 7373
	Glasnevin	
	Baile Tha Cliath	
248	Dublin City University	Nokia 7373
	Glasnevin	Nokia 7610
	Baile Tha Cliath	
249	Dublin City University	Nokia 7373
	Glasnevin	
	Baile Átha Cliath	
250	Dublin City University	DPS1
	Glasnevin	Nokia 3120 classic
	Baile Tha Cliath	Nokia 6288
251	Dublin City University	Quacksalot

Event ID	Location Enhancement	Bluetooth Name
	Glasnevin	Flanders
	Baile Átha Cliath	SGH-J700I
252	Dublin City University	Sandra
	Glasnevin	Cock with no balls
	Baile Átha Cliath	Nokia 7373
253	Dublin City University	Nokia 7373
	Glasnevin	Sandra
	Baile Átha Cliath	Nic phone
273	Dublin City University	Daragh Byrne's 24inch iMac
	Glasnevin	cdvpminiColum
		Stop lukin at my bluetooth!
274	Dublin City University	Daragh Byrne's 24inch iMac
	Glasnevin	cdvpminiColum
275	Dublin City University	Daragh Byrne's 24inch iMac
	Glasnevin	cdvpminiColum
		N95
276	Dublin City University	Daragh Byrne's 24inch iMac
	Glasnevin	cdvpminiColum
		N95
277	Dublin City University	Daragh Byrne's 24inch iMac
	Glasnevin	cdvpminiColum
278	Dublin City University	cdvpminiColum
	Glasnevin	Daragh Byrne's 24inch iMac
279	Dublin City University	Daragh Byrne's 24inch iMac
	Glasnevin	cdvpminiColum
		Conors fne
280	Dublin City University	Bowers2
	Glasnevin	RAI N95
	Baile Átha Cliath	LFC2005
		Malleerrooo

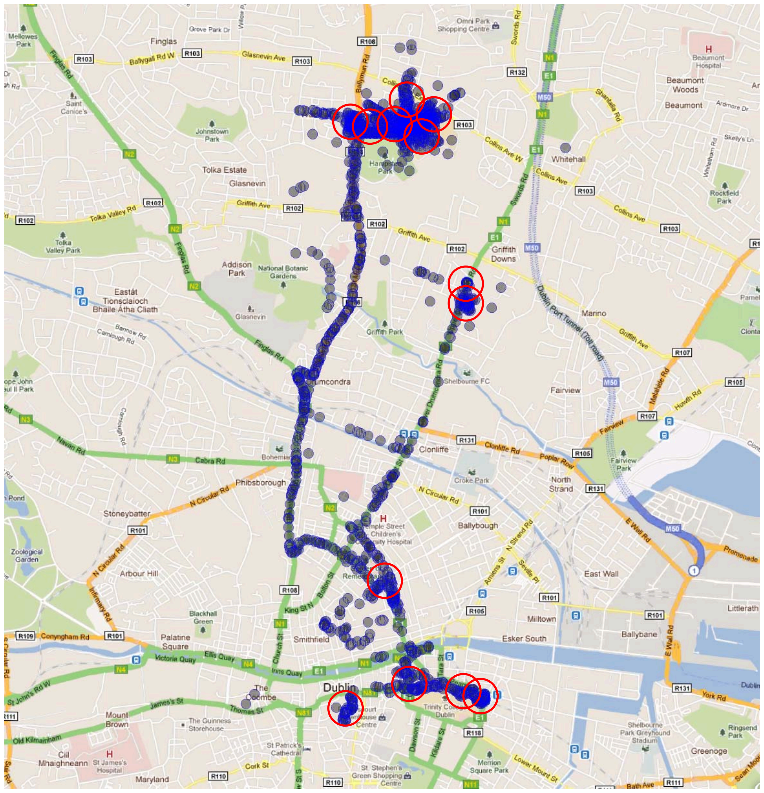
Appendix C

Event ID	Location Enhancement	Bluetooth Name
281	Mceniff Skylon	N95
	Dublin Skylon Hotel	Yuki
	Croke Park	Nokia N97
282	Marino	N95
	Mceniff Skylon	RAI N95
	Croke Park	Yuki
283	Mceniff Skylon	N95
	Dublin Skylon Hotel	Nokia N97
	Croke Park	RAI N95
284	Mceniff Skylon	Yuki
	Dublin Skylon Hotel	Nokia N97
	Croke Park	N95
285	Mceniff Skylon	Yuki
	Dublin Skylon Hotel	Nokia N97
	Croke Park	RAI N95
286	Mceniff Skylon	Yuki
	Dublin Skylon Hotel	Nokia N97
	Croke Park	Steve
287	Mceniff Skylon	Yuki
	Dublin Skylon Hotel	Nokia N97
	Croke Park	Up the Dubs
288	Mceniff Skylon	Nokia
	Dublin Skylon Hotel	Anto
	Croke Park	Nokia CK-7W
289	Mceniff Skylon	Anto
	Dublin Skylon Hotel	Lucyy
	Croke Park	Nokia 5800 XpressMusic
290	Dublin City University	Nokia 7373
	Glasnevin	
	Baile Átha Cliath	
291	Dublin City University	Nokia 7373
	Glasnevin	
	Baile Átha Cliath	

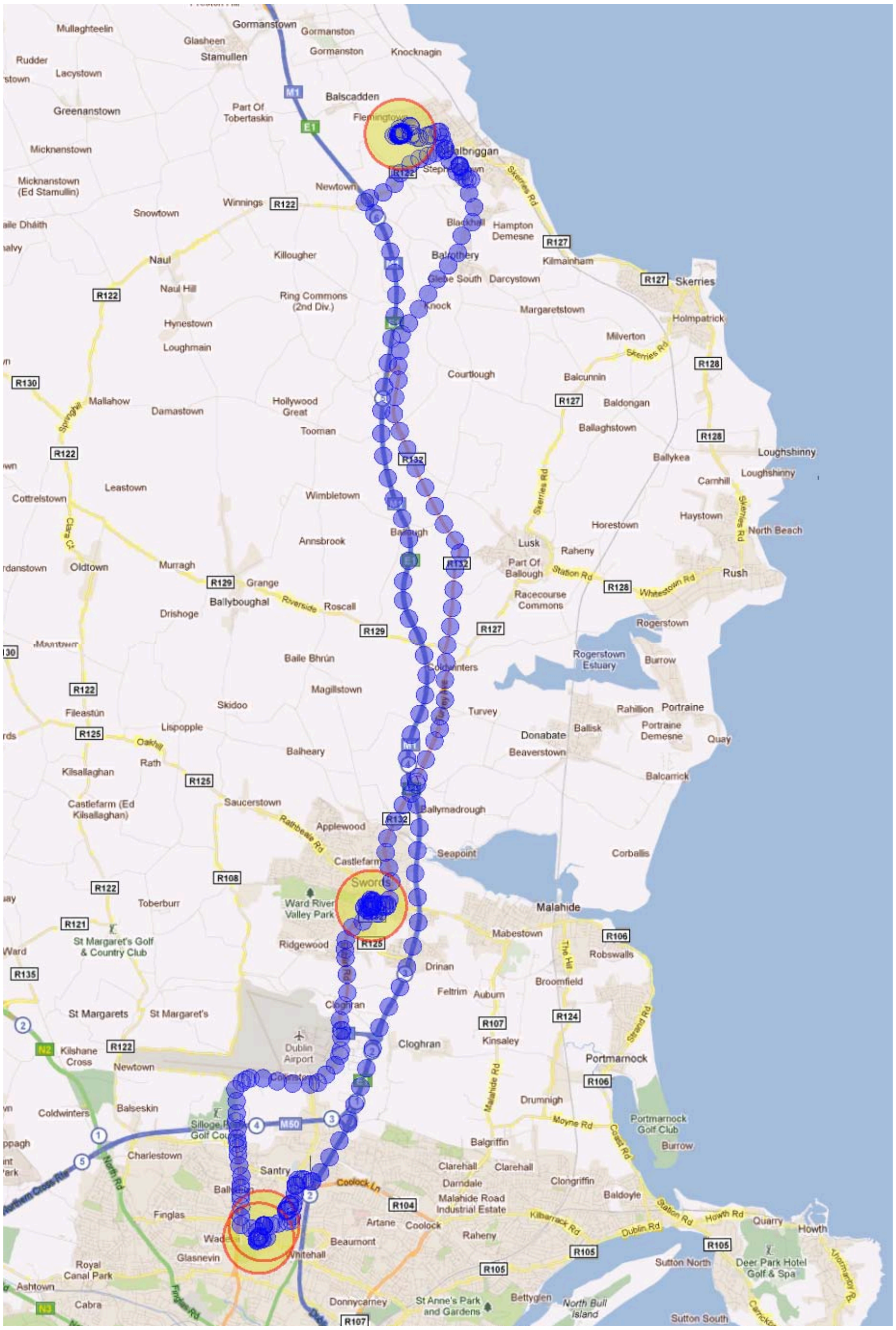
Appendix D

Location Clustering Examples

D.1 Clustering for a Whole Month of One Lifelogger



D.2 Clustering on Another User for a Day


Mullaghteelin
Gormanstown
Gormanston
Glasheen
Gormanston
Knocknagin
Rudder
Stamullen
Lacystown
M1
Balscadden
Part Of Tobertaskin
Greenanstown
Flemingtown
E1
Balbriggan
Micknanstown
Stephenstown
R122
Micknanstown (Ed Stamullin)
Newtown
Skerries Rd
Winnings
R122
Snowtown
Blackhill
Hampton Demesne
R127
Naul
Killougher
Kilmainham
Balrothery
R127
Skerries
R122
Naul Hill
Ring Commons (2nd Div.)
Glebe South
Darcystown
Hynestown
Knock
Margaretstown
Holmpatrick
Loughmain
Milverton
Skerries Rd
R130
Courtlough
Balcunnin
R128
Mallahow
Hollywood Great
R127
Baldongan
Damastown
Ballaghstown
Tooman
R128
R122
R132
Ballykea
Loughshinny
Loughshinny
Leastown
Skerries Rd
Carnhill
Cottrelstown
Wimbletown
Horestown
Haystown
Skerries Rd
North Beach
Annsbrook
Balrough
Lusk
Raheny
Oldtown
Murragh
R132
Part Of Ballough
Station Rd
R128
Whitestown Rd
Rush
R129
Grange
Ballyboughal
Riverside
Roscall
Racecourse Commons
Drishoge
R129
R127
Rogerstown
Balle Bhrún
Rogerstown Estuary
Burrow
Coldwinters
R122
Magillstown
Fieldstown
Skidoo
Turvey
Rahillion
Portraine
R125
Lispopple
Donabate
Ballisk
Portraine Demesne
Quay
Oakhill
Balheary
Beaverstown
Rath
Kilsallaghan
R125
Balcarrick
Saucerstown
Ballymadrough
Castlefarm (Ed Kilsallaghan)
R132
Applewood
Seapoint
Corballis
Castlefarm
R108
Swords
R122
Ward River Valley Park
Malahide
Toberburr
Mabestown
R121
Ridgewood
R125
R106
St Margaret's Golf & Country Club
The Hill
Robswalls
Ward
Drinan
R135
Feltrim
Auburn
Broomfield
Cloghran
R107
R124
St Margarets
St Margaret's
Kinsaley
Cloghran
Dublin Airport
Strand Rd
N2
Kilshane Cross
R122
Newtown
Portmarnock
Malahide Rd
R106
Drumnigh
Coldwinters
Balseskin
Moyne Rd
Portmarnock Golf Club
Silloge Park Golf Course
M50
Coast Rd
Burrow
Balgriffin
Charlestown
Santry
Clarehall
Clarehall
Clongriffin
North Rd
Darndale
Baldoyle
Ballymun
Coolock Ln
Malahide Road Industrial Estate
Finglas
R104
Station Rd
Howth Rd
Artane
Coolock
Kilbarrack Rd
Quarry
Howth
Wadelai
Raheny
Dublin Rd
Beaumont
Whitehall
R105
R105
Glasnevin
Sutton North
Deer Park Hotel Golf & Spa
Royal Canal Park
R105
Ashtown
Donnycarney
St Anne's Park and Gardens
Bettyglen
North Bull Island
Cabra
N3
R107
Sutton South