Ontology for multimedia content descriptions

Liu, Song

2008

Liu, S. (2008). Ontology for multimedia content descriptions. Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/40928>

<https://doi.org/10.32657/10356/40928>

ONTOLOGY FOR MULTIMEDIA CONTENT DESCRIPTIONS

A thesis submitted to the Nanyang Technological University
in fulfilment of the requirement for the degree of Doctor of Philosophy

by

LIU SONG

Supervisor: Dr. Chia Liang-Tien, Clement

Division of Computer Communications
School of Computer Engineering
Nanyang Technological University
Singapore

Aug 2008

Acknowledgement

Before the presentation of the research work, I need to thank many people who helped in this project and the development of my thesis.

First of all, I owe the greatest gratitude to my supervisor, Dr. Chia Liang-Tien, Clement. He is such a nice man who always takes great pains giving me remarkable guidance. No doubt, I have learned many things and benefited a lot from him during these years. Next, I would like to express my gratitude to Dr. Deepu Rajan for his support since the beginning, and for his patience and guidance. Without their continuous encouragement and advice, I could never complete this thesis.

I want to thank Dr. Lee Bu-Sung, Dr. Chan Syin and Dr. Sourav Saha Bhowmick for their help in Pet-Device project. I would like to thank all my friends and colleagues in Center of Multimedia and Network Technology (CeMNet), especially Yi Haoran, Hu Yiqun, Wang Huan, Wang Surong, Chen Ling and Chu yang, with whom I have worked together, for their patience of listening & discussion and friendship. And I need to thank Mr. Patrick Lim for his help in developing of my thesis.

Finally, I need to thank my parents and my wife. In these years, they supported my work very much and gave me great understanding. They encouraged me to work hard no matter if I have not much time to spend with them. Without all of their support, I could not have gone so far in my research work.

The main part of the research work in this dissertation has been completed by year 2007. After that, I accepted a job offer in industry. In this period, I worked on the experiments based on large image test set and the elaboration of Chapter 3. This caused the delay of the thesis submission.

Table of Contents

Table of Contents	ii
List of Figures	v
List of Tables	viii
Abstract	ix
1 Introduction	1
1.1 Background	1
1.1.1 Ontology and the Semantic Web	1
1.1.2 Content-based Media Retrieval	3
1.1.3 Multimedia Description Standard	4
1.2 The Proposed Research	5
1.2.1 Motivations and Objectives	7
1.2.2 Issues in the Proposed Research	9
1.3 Thesis Outline	11
2 Background Work	12
2.1 Knowledge Representation and Ontology	12
2.1.1 Introduction to the Semantic Web	13
2.1.2 RDF Model and RDF Schema	15
2.1.3 The Web Ontology Language: OWL	17
2.2 MPEG-7: Multimedia Content Description Interface	21
2.2.1 Introduction to the MPEG-7 Standard	21

2.2.2	Introduction to the MPEG-7 Descriptors	23
2.3	Overview of CBMR	23
2.3.1	Scope of Content-based Retrieval	24
2.3.2	Descriptions of Content: Features and Processing	26
3	Ontology and Content-based Image Retrieval	35
3.1	Ontology-based CBIR System	36
3.1.1	Overview of framework	36
3.1.2	The First Attempt of Integration	38
3.2	Ontology for Still Image Description	41
3.2.1	Overall Structure	44
3.2.2	Define and Build Ontology	47
3.2.3	Examples and Implementation	53
3.3	CBIR System with Semantic Knowledge Building	56
3.3.1	Related Work	56
3.3.2	System Design	58
3.3.3	Semantic Knowledge Building	62
3.3.4	Image Attributes Description	68
3.3.5	Web Image Retrieval	73
3.3.6	Experimental Results	80
3.4	Discussion	89
4	Feature Extraction for Ontology Enhancement	91
4.1	Visual Description: The Texture Correlogram	93
4.2	High-level Perceptual Information Description: Attentive Region	95
4.2.1	Overview for High-level Perceptual Information Retrieval	95
4.2.2	Attentive Region Selection for Professional Photographs	100
4.3	Experimental Results	113
4.4	Discussion	121
5	Ontology and Content-based Video Analysis	122
5.1	Ontology for Audiovisual Description	124

Ph.D Thesis

CONTENTS

5.1.1	Overview of the Ontology	124
5.1.2	Define and Build the Ontology	127
5.1.3	Example of the Ontology for Audiovisual Content Descriptions . .	132
5.2	Multi-modal Audiovisual Content Analysis	136
5.2.1	Multi-modal Basketball Video Analysis and Annotation	138
5.2.2	Multi-modal Structure Analysis and Event Detection	146
5.2.3	MPEG-7 Compliant Annotation File Generation	150
5.2.4	Experimental Results	152
5.3	Discussion	155
6	Conclusions	156
6.1	Achievements and Contributions	156
6.2	Future Work	160
	List of Publications	164
	Bibliography	166

List of Figures

2.1	Tim Berners Lee's vision of the Semantic Web.	14
2.2	Overview of MPEG-7 Multimedia Description Schemes (DSs) [1].	22
2.3	Four different ways to group image data [2].	31
3.1	Layer structure of Ontologies.	45
3.2	General structure of Domain Ontology.	47
3.3	General structure of Textual Description Ontology.	49
3.4	General structure of Visual Description Ontology.	51
3.5	Example of ontology system: ellipse and rectangle represents class and instance respectively.	54
3.6	Overall system structure for CBIR system with semantic knowledge building.	60
3.7	An example of typical Web page.	63
3.8	Downloading interface of image crawler program. A) project information and download status, B) system status, and C) database summary.	64
3.9	Experimental Result of Web Image Retrieval (1).	86
3.10	Experimental Result of Web Image Retrieval (2).	86
3.11	Experimental Result of Web Image Retrieval (3).	87

4.1	Expand the proposed Visual Description Ontology.	95
4.2	Different focal points layouts, (a) consumer compact camera focal point layout, (b) Canon 1Ds, (c) Canon 20D, (d) Nikon D2x.	101
4.3	Intensity change ratio curve for the edges with different blurs.	104
4.4	Boundary map for in-focus area, (a) original image, (b) boundary map for in-focus region.	106
4.5	Illustration of ‘Edge flow’ to pixel P_i	107
4.6	Example of attentive region detection: (a) original image, (b) edge map for original image and (c) the process of attentive region detection. . . .	111
4.7	Expand the proposed Visual Description Ontology to include attentive region information.	112
4.8	Performance comparison of texture correlogram, co-occurrence matrix and Ga- bor texture feature.	113
4.9	Results of focal point detection. The yellow rectangle region indicates in- focus areas, the red rectangle region indicates out-of-focus areas, and the purple rectangle region indicates homogeneous areas.	116
4.10	Comparison of our approach with the traditional visual attention model, (a) original image, (b) the saliency map generated using Itti’s model, (c) the attentive region generated by our algorithm. (d) the salient re- gion generated from saliency map.	117
4.11	More results for attentive region generation, (a) original image, (b) bound- ary map for in-focus region, (c) salient region map detected using ‘edge density’ information and (d) final attentive region map.	119

4.12	Examples of attentive region size detection. (a) attentive region of the photographs and (b) rectangle representation for attentive region.	120
5.1	Overall Audiovisual Ontology Structure.	125
5.2	Example of an Audiovisual Domain Ontology.	128
5.3	The general structure for Content Analysis Ontology (1).	129
5.4	The general structure for Content Analysis Ontology (2).	129
5.5	The general structure for Concept Analysis Ontology.	131
5.6	Example of the Content Analysis Ontology.	132
5.7	Example of Concept Analysis Ontology.	134
5.8	Example of typical scenes.	142
5.9	The structure of hierarchical SVM.	144
5.10	Example for ODI detection and event detection.	148

List of Tables

3.1	Results of Web Image Retrieval (Google) using Textual Information. . . .	83
3.2	Results of Web Image Retrieval (Google) using Textual & Visual Information.	84
3.3	Results of Web Image Retrieval (Google) using Textual, Global and Local Visual Information.	85
3.4	Results of Web Image Retrieval (Yahoo) using Textual Information. . . .	88
3.5	Results of Web Image Retrieval (Yahoo) using Textual & Visual Information.	88
3.6	Results of Web Image Retrieval (Yahoo) using Textual, Global and Local Visual Information.	89
5.1	Classification rates for level 1 and level 2 classes.	153
5.2	The statistics about the appearance of ODI detection.	154
5.3	The statistics about the appearance of potential event detection.	154
5.4	Results of event detection using multi-modal approach.	154

Abstract

Web ontology provides us a valuable tool for describing and representing domain knowledge on the Semantic Web. In a knowledge sharing scenario, an ontology defines the terms that are used to describe and represent an area of knowledge. Such ontology is used by people, databases and applications that need to share domain information. It is attractive to integrate Web ontology in a multimedia retrieval system, in order to facilitate fast and efficient media retrieval with an acceptable level of content interpretation.

In this dissertation, as we desire to utilize ontology technology in multimedia retrieval, we have been looking at this topic from the perspective of creating high-level semantic descriptions for multimedia content, and also from the point of view of media annotation and retrieval. The novelty of our research lies in designing and integrating Web ontologies into a content-based media retrieval system. A general framework is presented for combining the ontology for still image description which provides domain knowledge descriptions, as well as an image retrieval system. A Web image retrieval system was implemented in which semantic knowledge is extracted from Web content. A video analysis and annotation system was designed and implemented based on the content and concept descriptions that are encapsulated in the ontology for audiovisual descriptions. Moreover, we also investigated how to achieve better media content description by designing and extracting new low-level and high-level feature descriptors.

To achieve the content description for still images, we present an ontology for still image description. In the implemented ontology, the domain concepts, textual descriptions, and visual descriptions are encapsulated in a single ontology system. Therefore, it contains multi-modality information including both high-level and low-level descriptions for image content, together with the relevant domain knowledge. This multi-modality ontology becomes a rich resource for image content descriptions in our Web image retrieval system. In this system, the relevant images can be retrieved from the image database

according to user input query keywords or narrative descriptions. The retrieval efficiency is improved in the average retrieval precision, compared to single modal and non-ontology approaches.

The context aware feature descriptor achieves better performance in both image retrieval and classification. A new texture low-level feature: texture correlogram was developed to capture textures as well as their spatial distribution information. Furthermore, we also present a new approach to automatically detect the attentive region in professional photographs by analyzing high-level out-of-focus blur (OFB) information. Based on the extracted attentive regions and exchangeable image file format (EXIFs), the estimation of geometric size for a focused object can be achieved. This geometry information can greatly facilitate high-level image indexing and retrieval by providing domain context information of query objects.

We also implemented an audiovisual description ontology to provide semantic interpretations for audiovisual data at both content and concept levels. At the level of content description, it provides information on how to produce semantically aware video and audio segments. Beyond content analysis description, it also provides the descriptions for audiovisual domain concepts and captures the information regarding how semantically meaningful concept descriptions can be generated. By integrating the audiovisual description ontology, we introduced a novel semantic analysis and annotation framework by using multi-modal analysis of video and audio information. A basketball video annotation and analysis system was implemented to test the proposed framework.

Chapter 1

Introduction

1.1 Background

1.1.1 Ontology and the Semantic Web

The first area in our research is Web Ontology. It emerged along with the development of the Semantic Web [3]. The Semantic Web is a vision for the future of the Web in which available information is given well-defined meaning. It enables machines to automatically process and integrate information available on the Web. Moreover, it also enables greater synergy between computers and people. Research topics on the Semantic Web are prevalent because currently it is very difficult to perform an accurate search on the Web. For example, if we want to search for a video clip about Shakespeare, most Web search engines return a large quantity of pages referring to the works of Shakespeare that are redundant or irrelevant. If we want computers to perform the searches effectively, we may need to clearly define the meaning of both requests and data in the target database. Apparently, correct identification of related information would lead to a more accurate search. The development of the Semantic Web is driven by this demand for more accurate searches. The basic idea of the Semantic Web involves annotating documents with semantic markup. That means, we annotate a document not simply just for

presentation on a display but also for the expression of content within a document. This is often described as “a Web understood by machines” as opposed to “a Web understood by humans”. Therefore, Resource Description Framework (RDF) [4], the first language constructed for the Semantic Web, was quickly set up for describing the relationships between the resources on the Web. To achieve more powerful semantic markup, the World Wide Web Consortium (W3C) has launched a long-awaited Semantic Web activity, it took over the development of RDF and its extensions.

The Semantic Web is built upon XML’s [5] capability of defining customized tagging schemes, and RDF’s flexible approach to representing data. With the research going on, the next required element of the Semantic Web is Web Ontology. An ontology is a specification of a conceptualization [6]. The word “ontology” has been used in philosophy for a long time, in which it refers to the subject of existence but it has a very different meaning in the context of knowledge sharing [7, 8]. That is, an ontology is a description (like a formal specification of a program) of the concepts and relationships that can exist for an agent or a community of agents. This definition is consistent with the usage of ontologies as sets-of-concept-definitions [6]. In a knowledge sharing scenario, an ontology defines the terms that are used to describe and represent an area of knowledge. Such ontology is used by people, databases and applications that need to share domain information. In other words, computer-usable definitions of basic concepts in a given domain and the relationships among them are encapsulated in ontologies. Moreover, an ontology is able to encode knowledge in a domain and also knowledge that spans domains. A domain indicates a specific area of knowledge or area of subjects from a human perspective, such as “nature”, “art” or “medicine”. The knowledge of this domain contains a set of human conceptual items that are defined by the specialists of that given domain. In brief, ontologies make the domain knowledge reusable [9, 8].

Ontologies can be used to improve the performance of existing Web-based applications

and may provide new uses for the Web. For example, ontologies could be integrated into Web portals to enable intelligent searching of the contents that are requested by users [10]. The inferences handled by ontologies can allow users to obtain some search results by looking at context and relevant information, which is hard to achieve for conventional retrieval systems. Ontologies can also be used in the areas of “corporate web site management”, “design documentation”, “agents and services”, “ubiquitous computing”, and “multimedia collections” [9, 11]. Therefore, ontology building and manipulation have received more interest and attracted more research efforts recently. However, most ontologies are designed for textual content but not for multimedia data [12, 13, 14, 15]. To build ontologies providing the content descriptions for multimedia data, we need to have the knowledge from both ontology creation and content-based media analysis which is the other research area involved in this dissertation.

1.1.2 Content-based Media Retrieval

Content-based image retrieval (CBIR) is the first research topic in content-based media analysis. CBIR systems utilize measures that are based on low-level image attributes, including color, shape, texture and so on. Current research in CBIR focuses on powerful methods that can find regions from an image according to given objects that users wish to retrieve. There have been some success in finding human faces of different selected sizes, known patterns of horses, zebras and other textured animals, and such backgrounds as trees, water, and sky. Similarly, the content-based video retrieval (CBVR) systems utilize measures that are based on low-level video attributes, such as motion, color, texture and so on. Current research in CBVR focuses on video authoring, abstracting, indexing and retrieving based on the content and structure of the video data. Moreover, both CBIR and CBVR are hybrid research areas that require knowledge from both computer vision and database systems. Large image (or video) databases are created, and images (or

videos) from these collections are utilized in many areas where images (or videos) can be used to enhance the product. These images (or videos) are generally organized loosely by categories, such as “animals”, “natural scenes”, “people” in the image domain, and “news”, “movie”, “sports”, “drama” in the video domain. However, most image (or video) indexing and annotations are done manually. That means people list the important objects and other terms in an image or a video clip by which users may wish to access them. However, this method is not suitable for today’s very large image (or video) databases due to the heavy manual annotation work [16].

Content-based media retrieval (CBMR) is not yet a commercial success, because most real users want to search for multimedia data by specifying semantically meaningful descriptions of scenes or objects. This requires the CBMR system to capture and manipulate both low-level image (or video) attributes and domain knowledge that varies with different contexts. The other problem that affects the development of CBMR is the diversity of feature descriptions. Since different people describe these non-textural objects in different ways (using different low-level features or different expressions of the same feature), it is difficult to exchange the feature descriptions between different systems and correspondingly challenging to design a general engine for accurate and high performance image searching and querying. This situation has remained unchanged until the emergence of MPEG-7 [1].

1.1.3 Multimedia Description Standard

MPEG-7 [17, 18, 19], formerly named “Multimedia Content Description Interface” was created by the Moving Pictures Expert Group (MPEG), a working group of the Geneva-based ISO/IEC standards organization. It is a standard for describing multimedia content. The objective of this standard is to develop a rich set of standardized tools to enable both humans and machines to generate and understand audiovisual descriptions. Differ-

ent from other MPEG standards, MPEG-7 describes audiovisual information regardless of storage, coding, display, transmission, medium, or technology. It addresses a wide variety of media types including: still pictures, graphics, 3D models, audio, speech, video, and the combinations of these. By using MPEG-7 descriptions that describe multimedia content, users can index, search, browse and retrieve that content more efficiently than they could by using today's text-based search engines. It is important to note that MPEG-7 addresses many different applications in many different environments, which means that it needs to provide a flexible and extensible framework for describing audio-visual data. Therefore, MPEG-7 defines a multimedia library of methods and tools, which can be extended in the future.

1.2 The Proposed Research

Nowadays, the demand for querying and retrieving audiovisual information efficiently increases from day to day. This requires a machine-processable representation of audiovisual data and their associated semantics. In this case, providing semantic descriptions to multimedia content is a requirement not only of the Semantic Web but also of high performance media retrieval systems that can fulfil the demand of users. However, it is much more complicated and difficult for machines to extract meaningful semantics from multimedia data than from natural language documents. Currently, multimedia resources are typically indexed by captions or metatags, but not semantic descriptions for their contents. Therefore, building semantic descriptions for multimedia content poses a great challenge. Web ontology provides a potential measure to solve this problem through the implementation of the semantic annotations for collections of images, audio, or other non-textual entities. Our research focuses on a unified methodology (Web ontology) for semantic representations of multimedia contents, and implement it in a content-based retrieval system. The fundamentals of a portion of our research is, existing

object-recognition techniques in computer vision, which is based on low-level audiovisual attributes extraction. By combining both low-level audiovisual descriptions and domain knowledge, numbers of middle-level features or descriptions for multimedia content are extracted and stored in an ontology. Facilitated by ontology technology, all encapsulated information becomes reusable and exchangeable when we build ontologies with standardized feature descriptors such as those provided by the MPEG-7 standard.

There are two types of ontologies for multimedia descriptions: one is *information-specific ontology* and the other is *content-specific ontology*. Information-specific ontologies mainly describe the properties or attributes of different media (the information of the media file), such as the length, the author, or the name of a video clip. This information is entity-specific, which can not be reused by other entities. Normally, people use literal descriptions (human language) to describe such information. Content-specific ontologies mainly describe the subjects and the contents of a resource such as the participants or what subjects are included in an image, which can be reused by other documents that have a topic in the same domain. There are also two categories of content-specific ontologies that are introduced in multimedia content descriptions. The first is built with the annotations of contents, which come from a spectator viewpoint, while the second is built with the content's feature descriptions (the metadata [20] that describe the subjects). Currently, most of the annotations are carried out manually. On the other hand the feature descriptions can be extracted automatically by machine. The MPEG-7 standard also provides us a recommended method to extract the descriptor for some feature descriptions, which is a great convenience for building content-specific ontologies. In this thesis, the ontologies have the same meaning as content-specific ontologies. We named our ontology, "Ontology for Multimedia Content Descriptions" because all metadata used in the ontologies are similar to the descriptors provided by the MPEG-7 standard.

1.2.1 Motivations and Objectives

My dissertation title is “Ontology for Multimedia Content Descriptions”. To develop a library of *semantic descriptions for multimedia contents*, we needed to provide the machine-processable representations of their associated semantics, likely to describe very different aspects of contents, which can be integrated into the proposed ontologies.

The novelty of our research comes from integrating Web ontology and multimedia content descriptions into a multimedia retrieval system to achieve efficient and semantic-aware media query and retrieval. The primary objective of the proposed ontologies was to capture additional knowledge about the domain, which helps to improve the efficiency of media retrieval. For example, users can retrieve a video clip by providing only a short description of the contents, which is not included in the information (metadata or metatags) of the video file. In this case, it is very important that the search facility goes beyond just such simple keyword matching. This kind of content-based retrieval could be achieved by incorporating ontology technology into conventional media retrieval systems. We also attempted to narrow the *sensory gap* and *semantic gap*¹ which are dealt with the area of “Content-based Media Retrieval” through building the ontologies for multimedia content descriptions based on MPEG-7 and other descriptors. On the other hand, ontology design and development have become increasingly important in recent years as main components for creating the Semantic Web. The ontologies for multimedia content descriptions are indispensable parts of Web ontology, as multimedia data are indispensable parts of the Semantic Web. Another key objective was to reuse and share knowledge between different domains or specific areas of knowledge. These were the original reasons for our investigations into building ontologies for multimedia content descriptions. There are many applications, such as automatic media annotation, content-based media retrieval, ontology-based human action prediction, etc., [21, 22, 23]

---

¹For the reference to these two gaps see Section 2.3.1 on page 24

which further provided the motivation for our research.

The main objective of our research was to develop ontologies that contain machine-processable, highly structured descriptions for human conceptual items that describe multimedia contents. MPEG-7 Ontology was first proposed in [24], which provides structured description for MPEG-7 descriptors. However, it does not contain an implementation to describe an actual domain. Furthermore it does not include the domain or context information, therefore, it cannot be applied directly to describe high-level human conceptual items that present in multimedia data. Moreover, we also needed to develop a multimedia retrieval system where media data are described by semantic descriptions and can be accessed by using the human conceptual items that describe the media contents. This is a kind of content-based multimedia retrieval system in which knowledge representation techniques (ontology) for creating semantic descriptions in a specific domain are adapted. With the reasoning power provided by ontologies, such a retrieval system could deal with *fuzzy* information or *partial* information to achieve high-performance and accurate multimedia retrieval in a specific domain. For example, in image retrieval, a system needs to query and retrieve all basketball scenes out from an image database. Image matching should not depend on only the shape or the color of the basketball but also other related information (e.g. the context of scene and camera position changes). Moreover, the proposed ontologies should be extendable. We should be able to combine new knowledge when needed and also incorporate information from other domains. In order to achieve an intelligent retrieval instead of simple “key-word matching” that is currently used in most search engines, the retrieval system needs to have appropriate strategies to handle the knowledge that is provided by these ontologies. This requires the system to choose and manipulate the knowledge in the best way possible. Therefore, to develop an efficient and semantically aware multimedia retrieval system, a number of research issues need to be addressed. They include multimedia content descriptions, knowledge sharing and rep-

resentation, multi-domain matching (and retrieval) algorithm, and inference repository for multimedia contents. A key problem is knowledge representation management. A successful representation enables the human conceptual items for describing the multimedia content and the relationships between them to be described in a very clear and logical structure. This allows both humans and machines to easily understand and access the domain knowledge and also multimedia entity. In recent reports, EU IST AceMedia integrated project [25] utilized same technological approach to make multimedia content more relevant to the user. In this dissertation, we present the different methods for creating ontology and tackle semantic gap in different domain by introducing a framework to integrate ontology into multimedia analysis and retrieval system.

1.2.2 Issues in the Proposed Research

The first issue we tackled was how to build ontologies to provide multimedia content descriptions. To develop such ontologies, we needed to manipulate two categories of semantics extracted from audiovisual data, which are content and concept semantics. The content semantics indicate the feature descriptions for the media data while concept semantics comprise of content descriptions from the human conceptual aspect. For content semantics, we extracted feature descriptors from media data using feature extraction tools and stored the feature descriptors in a formal structure which enabled it to be accessed and manipulated through a standardized interface. For concept semantics, we needed to define the classes to represent the high-level human conceptual terms and also the properties to describe the relationships between these terms. Generally, there are two kinds of properties which describe the subsumption relationships between the terms and properties. They are the subsumption properties provided by the ontology language, such as *rdfs:subClassOf* and *rdfs:subPropertyOf*, and user defined properties that describe the semantic relationships between the terms in a certain domain knowledge. Normally, the

domain experts are responsible for defining the classes and properties according to the domain knowledge needed to be encapsulated into the ontology. However, in our scenario, we had to handle one more issue, that is, how to associate defined terms and relationships in an ontology with multimedia contents.

Therefore, the second issue was how to analyze the multimedia data and extract the necessary information. Subsequently, we needed to apply the information into the ontology definitions. This issue can be construed as a semantic multimedia analysis task in which the team attempted to detect or extract the terms and relationships from multimedia data through analyzing the low-level attributes of the data and domain knowledge. Another potential solution attempted was the use of low-level attributes of the data to provide the semantic interpretations for the terms and relationships that are defined by domain experts. For image analysis, limited by the semantic object segmentation and recognition, only numbers of objects can be detected in an image, and normally requires a lot of human interaction. At the same time, conventional image analysis systems do not combine high-level information (textual information and domain knowledge) with low-level image attributes to analyze the content of an image. This complicates the accessing and understanding of image contents. For video analysis, we needed to solve video segmentation and classification (or clustering) problems. Furthermore, to extract the concept semantics from videos, we had to combine multi-modal features and domain knowledge in some cases. Since semantic video analysis is a process that follows a user-defined workflow, and different domain knowledge is involved at different stages of analysis; how to manage the workflow and the corresponding domain knowledge, and making them reusable, was a great challenge. After defining the ontology for multimedia content descriptions, and extracting the corresponding semantics from the multimedia data, the next step was to apply the ontology in a multimedia retrieval system. To adapt the ontology into a multimedia system, the third issue was the definition of a system

structure which handles both content and concept semantics.

1.3 Thesis Outline

The rest of the thesis is organized as follows. Chapter 2 reviews related research areas concerned with this dissertation, involving knowledge representation and content-based media retrieval. Chapter 3, first describes an image retrieval system structure that implements ontology technology. The rest part of the chapter deals with building an image content description ontology including model-based and discrimination-based approaches. The experiment's results of the proposed system demonstrate that the performance of the content-based image retrieval system can be improved by combining the high-level information encapsulated in ontologies. Chapter 4 introduces a new texture visual descriptor and also a new method for extracting human perceptual information — the attentive region in professional photographs which can be further integrated into the proposed ontology system to enhance its descriptive power. Chapter 5 firstly presents our work on building video content description ontology and then describes a multi-modality video annotation system based on the proposed ontology. Lastly, in Chapter 6, we summarize the research work that was done over the last four years and issues that will be addressed in the future.

Chapter 2

Background Work

In this chapter, we review related research areas concerned with this dissertation. The first area we want to introduce is knowledge representation and ontology which is directly associated with our research topic. Secondly, we discuss the multimedia content description interface: the MPEG-7 standard which is a widely accepted standard for multimedia content description. Lastly, we describe a state-of-the-art content-based media retrieval system.

2.1 Knowledge Representation and Ontology

The use of ontology to represent knowledge is gaining popularity with the development of the Semantic Web. As a very important component of the Semantic Web, ontology provides the possibility of managing human conceptual items and relationships between them (known as domain knowledge) in a highly structured hierarchy. In [11], ontology was utilized to solve the problems with static, restricted, and inflexible traditional Web linking. At the same time, it enables the knowledge to be assessed, retrieved and reused easily. In this section, we first introduce the emergence of the Semantic Web. After that, we describe the Resource Description Framework (RDF) model which is the fundamental model for ontology building. Lastly, we describe the current ontology building language:

Web Ontology Language (OWL).

2.1.1 Introduction to the Semantic Web

Most Web content existing today is designed to be read by humans, but it is difficult for computer programs to understand their meaning. It is very difficult for a machine to handle information regarding the semantic meanings and relationships between Web resources. Currently the Web lacks the feature of integrating machine-processable data. Therefore, it is hard for machine to achieve fast and accurate information search and retrieval, especially, for non-textual content. The Semantic Web was conceptualized to solve this problem. “The Semantic Web is an extension of the current Web in which information is given well-defined meaning, better enabling computers and people to work in cooperation.” — Tim Berners-Lee, James Hendler, Ora Lassila — The Semantic Web, Scientific American, May 2001. The definition of the Semantic Web given by W3C is: *The **Semantic Web** is the abstract representation of **data** on the World Wide Web, based on the RDF standards and other standards to be defined.* The Semantic Web brings structure to the meaningful content of Web pages, creates an environment where software agents roaming from page to page can readily carry out sophisticated tasks for users [3]. It is aimed at two main kinds of information on the Web. One is human readable material, and the other is machine-processable data. The basic idea about the Semantic Web also extends from the information about information, normally known as metadata. Metadata is the data that enables people to get new information from some existing information. In this case, we can describe the Semantic Web as: the Web with rich metadata. Figure 2.1 shows Tim Berners-Lee’s vision of the Semantic Web. The basis of the Semantic Web is the Uniform Resource Identifier (URI) [26]. A URI is a Web identifier that uses a string to represent a source on the Web. It makes the resource available under a variety of naming schemes and access methods such as HTTP, FTP,

Figure 2.1: Tim Berners Lee's vision of the Semantic Web.

and Internet mail addressable. In fact, every thing *on the Web* has a URI and the Web is built on top of URIs. The layer above URI is XML and the XML Schema layer. This provides a way to mark up and structure the data with element and attribute tags and thus makes information exchangeable between machines. The next layer is RDF and the RDF Schema layer. Basic resource description models are created in this layer, in which the models provide a possible formulation of the universal Web of semantic assertions. On top of the RDF and RDF Schema layer is the Ontology layer. This layer provides more expressive descriptions to Web semantics than the RDF model, thereby enabling structural domain knowledge descriptions and knowledge sharing. The Logic layer is created based on a logical language for making inferences. This layer will be useful when the Semantic Web becomes expressive enough to help us in a wide range of situations. The last two layers are Proof and Trust. The motivation for creating these two layers is to deal with the states of concept-description conflictions. This means the Semantic Web can proof-check for situations when different users give conflicting semantic descriptions for the same resource. Very little is written about these layers now, but applications in

the future will generally contain proof checking mechanisms, such as digital signatures, to support further demands. In the following subsections, we will introduce the RDF layer (RDF and RDF schema) and the Ontology layer in more detail. These two parts are important for mapping semantics of domain knowledge (human conceptual items) to computer-processable models, which is essential for the purpose of the Semantic Web.

2.1.2 RDF Model and RDF Schema

The first language adopted for the Semantic Web is RDF. It is a framework for describing and interchanging metadata. RDF is designed and developed based on the following rules:

- **A *Resource*:** any thing having a URI is a resource.
- **A *Property*:** a resource that has a name and can be used as a property.
- **A *Statement*:** a statement is a combination of a resource, a property and a value. These parts are known respectively as the “subject,” “predicate,” and “object” of a statement.
- **A *Graph Serialization Syntax*:** XML was chosen as the language, in order to store instances of this model into files or to communicate these instances from one agent to another.

RDF can also be defined as a language that utilizes three URIs in a triplet, where every URI represents a *Resource*. From the resource description aspect, we can say “the Semantic Web is generally built on the syntaxes (RDF syntaxes) where data is represented by URIs.” RDF is a carefully designed language. The RDF model can be represented by XML syntax directly, so it is easy for us to interchange the model around. RDF statements have a very simple expression that is: three-part records (Resource, Property, Value). Therefore, it is easy for everyone to manipulate it. All the properties, values and

statements are resources in the RDF model. In this case, a property can also have its own property, which enables the creation of metadata for describing the relationships between properties. This means RDF has the feature of self-description. It is a very important feature, especially, when we have a large amount of properties, values and statements.

Since it is easy to create, modify and extend an RDF model, the RDF framework provides good scalability and makes “every one is free to say anything” a possibility. However, there is the perpetual question regarding the implementation of RDF: Why would someone want to use RDF, a new language, instead of the existing XML, with newly invented tags? The answer to this question is that the RDF framework provides a rich feature set that cannot be achieved by using XML. Firstly, RDF provides a methodology to directly and unambiguously map information into models. An RDF model is decentralized for which there are many generic parsers already available [27]. Secondly, XML does not fulfill the design requirement of *Scalability*. This issue can be explained from two different aspects. 1) The order in which the elements appear in an XML document is significant and often very meaningful, which seems highly unnatural in the metadata world. 2) Some constructions allowed by XML are hard to handle, even in moderate amounts, let alone by the billion [20].

RDF provides a basic model for metadata, and a syntax that enables independent parties to exchange and use it. It also means RDF only provides a skeleton but not the details. People are required to set up a series of properties to describe the resources that are utilized in an RDF model. We call such a package of properties, *Vocabularies*, which means these properties are correlative or have some internal relationships between them. Since any one can invent and also share the *Vocabularies*, RDF becomes a very important part of the Semantic Web. In [28], S. Cranfield proposed a method to convert Meta-Object Facility (MOF) based modelling language into an equivalent RDF representation. This proves that Object Management Group’s ontology definition model can be achieved

by using RDF. RDF Schema [29] was designed to be a simple datatyping model for RDF. Its role is to provide a mechanism for declaring the properties used in RDF, and for defining the relationships between properties and other resources. A schema defines not only the properties of resources (e.g., title, author, subject, size, color, etc.) but also the kinds of resources being described (e.g., books, Web pages, people, companies, etc.) [29]. The three most important concepts derived from RDF and RDF Schema are: “Resource” (*rdfs:Resource*), the “Class” (*rdfs:Class*), and the “Property” (*rdf:Property*). These concepts are defined as “Classes,” which means we can define terms that belong to these classes. RDF Schema also has a few more properties that we can make use of: *rdfs:subClassOf* and *rdfs:subPropertyOf*. These two properties allow us to indicate that one class or property is a sub-class or sub-property of another. We can create a clear hierarchy for classes and properties by utilizing these properties. Two other concepts provided by RDF Schema which must be mentioned, are ranges (*rdfs:range*) and domains (*rdfs:domain*). Ranges and domains let us define what classes the subject and object of each property must belong to. RDF Schema also contains a set of properties that provides annotation for vocabularies. Properties *rdfs:label* and *rdfs:comment* are used to add comments and labels for an RDF resource. RDF Schema is very simple and easy to use. Therefore, it allows rapid construction of an RDF description model for Web resources.

2.1.3 The Web Ontology Language: OWL

An ontology is an explicit specification of a conceptualization. This word was borrowed from philosophy, in which it refers to the subject of existence. As we have mentioned in Chapter 1, an ontology formally defines a common set of terms that are used to describe and represent a domain [9]. The “domain” indicates a knowledge domain that people are interested in or are communicating about. The motivation of designing ontologies

comes from enabling knowledge sharing and reuse. The knowledge representation is the foundational function, which is hidden under that purpose. Ontologies can represent knowledge in different structural complexities, from simple classifications to metadata schemes and up to the level of complicated logical models. Ontologies can then be used by automated tools to power advanced services such as more accurate Web searches, intelligent software agents and knowledge management. Since an ontology provides the means for representing the semantics of resources and enables the semantics to be used by Web applications and intelligent agents, it has become a crucial part of the Semantic Web. An ontology also provides a helpful way to structure and define the meanings of metadata for efficient and accurate knowledge searching or navigation. The application that implements ontologies becomes more intelligent, in the sense that it can more accurately “understand” the meaning of human conceptual items. Some tools also provide automatic reasoning by integrating ontologies as a knowledge base. Therefore, we can create intelligent applications by utilizing these tools. Such applications can be applied in various areas to facilitate: conceptual and semantic searching and retrieving, decision or prediction support, speech or natural language processing understanding, and knowledge management.

As ontology plays a high priority role in the applications, hence selecting an appropriate language to create ontologies is the first issue we needed to investigate during ontology designing and building. Though XML Schema is efficient for exchanging data between parties, it cannot be an ontology language because it lacks the mechanisms or syntaxes for performing the job of inventing new XML vocabularies reliably. RDF and RDF Schema approach this issue by allowing simple semantics to be associated with identifiers. In RDF Schema, we can define one class with several attached sub-classes and super-classes, likewise for the properties. By utilizing RDF Schema, we can describe the relationships between resources, such as: one term may have different meanings in

different contexts, and different terms may share the same meaning. In this sense, RDF Schema is a simple ontology language. However, RDF and RDF Schema are too simple to represent the complicated terms and the relationships between them, such as: disjoint and aggregate relationships. To close the gap between RDF Schema and the requirements of ontology creation, Web Ontology Language (OWL), a language designed for ontology building was first released in February 2004.

OWL is a revision of the DAML+OIL [30, 31] web ontology language that was first released in December 2000. It is an enhanced Web ontology language developed by incorporating the lessons learned from the design and application of DAML+OIL. Similar to the previous Web ontology language, OWL can be used to explicitly represent the meanings of terms in vocabularies and the relationships between those terms. However, OWL has more descriptive power for expressing meanings and semantics than XML, RDF, RDF Schema and DAML+OIL, and therefore machine interpretable contents on the Web can be represented more accurately by using OWL. OWL provides three increasingly expressive sub-languages designed to fulfill the different requirements of specific communities and users. These three sub-languages are “OWL Lite,” “OWL DL,” and “OWL Full.” OWL Lite supports those users primarily needing a classification hierarchy and simple constraints, while OWL DL supports those users who want maximum expressiveness whilst retaining computational completeness and decidability. OWL Full is meant for users who want maximum expressiveness and the syntactic freedom of RDF with no computational guarantees. OWL is a web ontology language based on RDF, however, it is more expressive and powerful than RDF Schema. Naturally, it inherits the language features of RDF and RDF Schema. In brief, the new language features provided by OWL are: “Equality,” “Property Characteristics,” “Restrictions,” and “Class Axioms and Expressions”. The first three features listed above are supported by OWL Lite while the fourth feature is supported by OWL DL and OWL Full only.

Equality : In OWL, equality or inequality relationships between vocabularies are expressed by using a set of equality description properties. They are: “equivalentClass,” “equivalentProperty,” “sameAs,” “differentFrom,” “AllDifferent,” and “distinctMembers.”

Property Characteristics : Property Characteristics features were introduced to provide information concerning properties and their values. They are “ObjectProperty,” “DatatypeProperty,” “inverseOf TransitiveProperty,” “SymmetricProperty,” “FunctionalProperty,” and “InverseFunctionalProperty.”

Restrictions : Restrictions contain two components: “Property Restrictions” and “Restricted Cardinality.” Property Restrictions allow users to define how properties can be used by instances of a class, which include “Restriction,” “onProperty,” “allValuesFrom,” and “someValuesFrom.” Restricted Cardinality constrains the cardinality of that property on instances of that class including “minCardinality,” “maxCardinality,” and “cardinality.”

Class Axioms and Expressions : OWL DL and OWL Full support Class Axioms by extending the constructions of OWL Lite. Class Axioms contain “oneOf,” “dataRange,” and “disjointWith.” Boolean Combination features of Class Expressions contain “unionOf,” “complementOf,” and “intersectionOf.”

OWL has been proved that can handle large-scale real world data in [32]. In our research, OWL DL was selected as the language to create the ontology for multimedia content descriptions.

2.2 MPEG-7: Multimedia Content Description Interface

This section expands on the brief introduction to Multimedia Content Description Interface (MPEG-7) in Chapter 1. It firstly provides a brief introduction to the MPEG-7 standard, and then some descriptors are described in detail.

2.2.1 Introduction to the MPEG-7 Standard

The main elements [1] of the MPEG-7 standard are:

- ***A set of descriptors***: A descriptor (D) is a representation of a feature that defines the syntax and semantics of the feature representation.
- ***A set of description schemes***: A description scheme (DS) specifies the structure and semantics of the relationships between its components, which may be both descriptors and description schemes.
- ***A language that specifies description schemes, the Description Definition Language (DDL)***: It allows for the creation of new Description Schemes and, possibly, Descriptors, and also for the extension and modification of existing Description Schemes.
- ***System tools***: System tools are designed to support multiplexing of descriptions, synchronization of descriptions with content, transmission mechanisms and coded representations (both textual and binary formats) for efficient storage and transmission, management and protection of intellectual property in MPEG-7 descriptions.

The bases of MPEG-7 hierarchy are Ds and DSs, which are combined together to describe the content of multimedia data. Ds are designed for describing the following

attributes of multimedia content: low-level feature descriptions such as color, texture, motion, audio energy and so on; high-level features of semantic objects, events and abstract concepts; content management processes. Using Ds, the representations of the content's features can be created in varying degrees from simplicity to complexity. DSs expand the Ds by combining Ds and other DSs into more complicated structures and specifying the relationships between the components of Ds and DSs.

Figure 2.2: Overview of MPEG-7 Multimedia Description Schemes (DSs) [1].

The relationships between MPEG-7 Description Tools are shown in Figure 2.2. The basic elements are used as elementary components to define DSs, which include basic data types, mathematical structures, schema tools, linking, and media localization tools, as well as the basic DSs. Content management and content description describe the content from several different aspects. *Creation and Production* contain the information describing the creation and production of the content. *Usage* contains the information describing the usage of the content. *Media* contains the description for the storage of media.

Structure aspects contain the description about the structure of audiovisual content and *Conceptual aspects* contain the description about the conceptual notions of audiovisual content. *Navigation and Access* provide the user a convenient way to browse and retrieve by defining the summaries, views, and variations of multimedia contents. *Content organization* helps to organize and model multimedia contents. *User interaction* describes the user preferences and the usage history for consumption of multimedia material.

2.2.2 Introduction to the MPEG-7 Descriptors

One objective of the MPEG-7 standard is to provide standardized descriptions for describing the low-level features of media contents. The MPEG-7 low-level feature descriptors consist of Visual Descriptors and Audio Descriptors. The MPEG-7 Visual Descriptors describe the basic audiovisual media content on the basis of visual information. For video and images, the general classes of visual descriptors are *color descriptors*, *texture descriptors*, *shape descriptors*, and *motion descriptors*. The MPEG-7 Audio Descriptors include a series of low-level tools that apply to any sound in a general way. Based on the foundation of audio descriptors, the MPEG-7 Standard also defines a set of high-level tools that deal with specific sets of applications, such as query and retrieval of speech contents.

2.3 Overview of CBMR

The motivation of our research is to enable fast and efficient multimedia query and retrieval. The idea is to build a content-based media retrieval system based on Web ontology and low-level features, where the Web ontology enables structural knowledge representation and low-level features are used to describe the contents of media. Low-level feature descriptors are used to describe the attributes of elementary concepts and ontologies are used to create semantic descriptions for complicated concepts using elementary con-

cepts in a specific domain. Since our approach is designed based on content-based media retrieval, we present a review of existing content-based media retrieval systems in this section. This section begins with the scope and the computational steps for content-based retrieval systems, subsequently, the implementations and examples are described.

2.3.1 Scope of Content-based Retrieval

This discussion begins with the pattern in users' aims and applications. A wide variety of content-based retrieval methods and systems have been developed over the years. There are three general user aims [2] when using content-based retrieval system:

1. Users search for media data beginning with no specific aim other than trying to find interesting media data. The systems designed for those users are typically highly interactive, and have different methods to guide users' search & retrieval of media data. For example, in such retrieval systems, the search may start with some sample images or sketches and search results can be manipulated interactively with relevant feedback.
2. Users target specific media data. In this case, the search may begin with a media entity provided by users, and the search results will likely be the media data with similar objects, color, texture, or topic. This is a target search by samples. The system designed for this group of users needs to deal with a user sample and find all the matching results from the media database.
3. Users search for media data that represents a specific class, or category. A system designed for these users should return the results that belong to the same class as the arbitrary samples given by the users. Therefore, the system needs to deal with the classification information and is usually interactive with a domain specific definition of similarity.

From the target domain aspect, the media data can be divided into two different domains [2]. One is *Narrow domain* which means a media entity in this domain has a limited and predictable variability in all relevant aspects of its appearance. Since the variability of the content of media entities is limited, the semantic descriptions of the media entity are well-defined and mostly unique to other domains. The other is *Broad domain* which means a media entity in this domain has unlimited and unpredictable variability in its appearance even in the same semantic meaning. In a *broad domain*, the gap between the feature descriptions of content and the semantic interpretations is generally wider than that in a *narrow domain*. Accordingly, to correctly identify the semantic interpretation, a system needs to provide a full scale description for content features. In this scenario, a gap appears:

The sensory gap is the gap between the object in the world and the information in a (computational) description derived from a recording of that scene [2].

This gap describes a state that the system may trigger a false detect or alarm regarding objects when there is insufficient information to describe the features of the content from all relevant aspects. To close this gap, the content-based retrieval system should provide a mechanism to eliminate the influence of ambiguous information and to understand what potential explanations for the content are.

Another gap was found when implementing early retrieval systems. This gap is the *semantic gap*:

The semantic gap is the lack of coincidence between the information that one can extract from the visual data and the interpreting that the same data have for a user in a given situation [2].

The difference between human language and computer-processable data is the fundamental reason for this gap. Firstly, it is hard to use linguistic terms to describe an image

clearly, probably impossible, since every linguistic description is contextual but images may not be so. Secondly, it is difficult to create a bridge from human conceptual items to computer-processable data.

We try to minimize the two gaps that prevent the advancement of high performance content-based retrieval systems in this dissertation. For the sensory gap, MPEG-7 descriptors provide a rich set of standard methods for describing the multimedia contents, as for the semantic gap, a well-defined semantic system based on ontologies can provide a potential solution.

2.3.2 Descriptions of Content: Features and Processing

For a CBMR system, the descriptions of content are divided into two parts: some descriptions come from the image and video processing aspect; the others come from the feature summarization aspect. The image and video processing operations transpose the image data into another data that are relevant to the query, while feature summarization deals with mapping from pictorial information to feature values.

Features and Image Processing

Three kinds of attributes are widely investigated in image and video processing, including *Color*, *Texture*, and *Motion*. Image and video processing are introduced according to these three aspects in this subsection.

A. Color Processing

Color is used in image and video retrieval more actively than other low-level features in computer vision. An important characteristic of color is that it can be described by different color spaces that are mathematical representations of the human eye's response to color. RGB-representation is used widely in color description. However the RGB

space is not quite perceptually uniform. It is a good choice for objects that always record in standard conditions and the color of objects has little variation in recording, such as art paintings [33], trademarks [34, 35], or color composition photographs [36]. A significant improvement over RGB space has been achieved by utilizing of the opponent color representation in [37], which has opponent color axes $(R-G, 2B-R-G, R+G+B)$. The advantage of this representation is that the brightness information, the most sensitive part for humans, is isolated on the third axis. The HSV-representation is more suitable for object retrieval, because of its invariant properties. The element Hue in HSV color space is invariant when the illumination or camera direction has changed. The weakness of HSV space is that the scalar color quantization is not good enough to draw proper outcomes — the more complicated vector color quantization might be needed instead. Therefore, a new color model, HMMD-representation has been developed. HMMD space is a scheme of the scalar quantization that is suitable for image retrieval, and the proposed scalar color quantization of HMMD is equivalent to the appropriate vector color quantization of HSV. In our research, we utilized HSV and HMMD color spaces to provide color descriptions for image and video objects.

The *color invariant feature* was derived in [38] that came from an analysis of the Schafer model of object reflection. The authors derived that coarse patches under white light invariant color space

$$\left(\frac{R-G}{R+G}, -\frac{B-R}{B+R}, \frac{G-B}{G+B} \right),$$

only depend on sensor and surface reflection rate. Especially, under conditions of a glossy surface with white illumination, the invariant representation can be derived as:

$$\frac{|R-G|}{|R-G|+|B-R|+|G-B|},$$

$$\frac{|R-B|}{|R-G|+|B-R|+|G-B|},$$

$$\frac{|G - B|}{|R - G| + |B - R| + |G - B|}.$$

These color models are robust enough to eliminate the side effect of major viewpoint distortion.

Another approach about *color constancy* was derived in [39]. Color constancy refers to the fact that humans perceive the same apparent color even though the physical spectrum has changed because of the variations of illumination. Color constancy was first introduced in [39], and in [40], the method was improved by adding specular reflection, shape, and varying illumination. By computing the blue-ratio vector $\left(\frac{r}{b}, -\frac{g}{b}, 1\right)$, only the color feature is used and the intensity is ignored. In 2D-space, a color constancy map can be generated from a normative scope of colors and surfaces. The ratios defined in [41]

$$\left(\frac{R_{x_1} G_{x_2}}{R_{x_2} G_{x_1}}, \frac{G_{x_1} B_{x_2}}{G_{x_2} B_{x_1}}, \frac{B_{x_1} R_{x_2}}{B_{x_2} R_{x_1}}\right),$$

provide more stability under the condition that geometry is a variable. Color constancy can also be implemented in retrieval by using an illumination invariant color representation. In [42], the authors indexed the ratio of neighboring colors, which makes illumination an independent retrieval condition. However this method brings on some loss of useful processing. In [43], the authors presented an evaluation of three color constancy techniques applied to a landmark detection system. The first technique was the well-known image conversion to a chromaticity space, and the second technique was based on successive lighting intensity and illuminant color normalizations. The third technique was proposed, based on color ratios, which unified the processes of color constancy and landmark detection.

There are two factors that argue against the use of color in content description: (1) the content's color may vary considerably with extrinsic conditions, such as the illumination position, the light's color, the camera's viewpoint and so on; (2) the human perception of color is difficult to simulate.

B. Texture Processing

Generally, textures are composed of a large amount of similar looking patterns. In the context of retrieval, the research concentrates on statistical or generative methods for the characterization of those patterns. The Grey-Level Co-occurrence Matrix (GLCM) texture was firstly introduced by Haralick in the 1970s. The GLCM is a tabulation of how often different combinations of pixel brightness values (grey levels) occur in an image. In image and video processing, the GLCM is utilized to provide the description for global texture for an image or an image patch. The Gabor texture descriptor was derived in [44]. It is calculated by extracting the mean and the variance of each channel as feature values of the texture of an image, from the image obtained after Gabor filtering in the frequency domain. A Gabor filter is a linear filter. It can be viewed as a sinusoidal plane of particular frequency and orientation, modulated by a Gaussian envelope. Therefore, for two images that contain a certain texture, we should receive similar impulse responses from a particular Gabor filter. The Markovian analysis was introduced in [45, 46], in which the property was generated by sliding a mask over the different localizations on an image [47]. Wavelet transform [48] has received more research efforts because of the features of locality and their compression efficiency. Most of wavelet transforms are created by grouping dilations or dilations and rotations that have semantic relationships. The lowest level of wavelet transforms [48] has been implemented in texture representation [49] which sometimes may be combined with Markovian analysis [50].

The Scale-invariant feature transform (SIFT) was introduced by David Lowe in [51]. It was designed to detect and describe local features in images. The SIFT algorithm can detect the particular interest points of an object, which are invariant to image scale and rotation. The interest points detected by using SIFT algorithm are highly distinctive and robust to changes in illumination, noise, occlusion and minor changes in viewpoint. Thus, it is a widely utilized local image feature in object recognition. Another important

technique uses multi-scale autoregressive MRSAR-models, in which textures are considered as the result of a deterministic dynamic system that is used to detect and describe noise [52]. The interest region detection has received considerable attention because its output can be easily utilized in object description and recognition. Harris-Affine and Hessian Affine [53] were two well proven and popular interest region detectors based on the affine invariant theory. The Harris-Affine detector locates interest points by detecting corners on second-moment matrix at multiple scales by using the Harris corner measure. The Hessian-Affine uses a multiple scale iterative algorithm to spatially localize and select affine invariant points. The interest regions detected by these algorithms are robust to image geometric transformation: scaling, rotation and shearing.

The texture feature is useful in satellite image analysis [54], but the texture properties only offer very little semantic references at the current stage.

C. Motion Processing

Motion features play a very important role in video processing. Such features can be applied in video segmentation, classification and event detection along with color and texture features to achieve better performance. In [55], global camera and object motion features were extracted and utilized in video searching and retrieval. The camera motion description tries to capture 3D camera motion parameter information including fixed, panning, tracking, tilting, booming, zooming, dollying, and rolling. The patterns of camera and object motion information were utilized to index and analyze the content of video sequences. The approaches for detecting shot and scene boundaries in video sequences by using motion vectors were reported in [56, 57]. The motion trajectory of an object which is defined as the localization, in time and space, of one representative point of this object. It is a simple, high level feature. A video retrieval approach was proposed in [58] based on the indexing of the object's motion trajectory information.

Feature Summarization in Image Processing

The extraction of feature values from pictorial information is discussed here. The first technique discussed is named “grouping data,” which is widely used in image retrieval. The image is first divided into parts before feature extractions are applied. This partitioning of the image is to obtain more features by extracting the feature in a relevant subdivision. There are four types of partitioning [2] shown in Figure 2.3.

Figure 2.3: Four different ways to group image data [2].

- The first is Strong segmentation [2], which means a complete object segmentation: “strong segmentation is a division of the image data into regions in such a way that region T contains the pixels of the silhouette of objects O in the real world and nothing else, specified by: $T=O$.” But, it is very difficult to implement the object segmentation for the broad domains of generic images.
- Weak segmentation [2] is easier to implement than strong segmentation. “Weak segmentation is a grouping of the image data in conspicuous regions T internally

homogenous according to some criterion, hopefully with $T \subset O$.” Weak segmentation algorithm may fail when two identical objects are close to each other. However, this kind of error rarely affects the results of retrieval. In [38], the homogeneity criterion is based on color and texture, and color is the criterion [59]. The limitation of the weak segmentation is a set of isolated points [38].

- “Localizing sign [2] is finding an object with a fixed shape and semantic meaning, with $T = x_{center}$.” Signs are helpful for content-based retrieval for their direct or unique semantic interpretation.
- Partitioning [2] is the weakest form of grouping: “a partitioning is a division of the data array regardless of data, symbolized by: $T \neq O$.” The feasibility of fixed partitioning comes from the fact that images are created according to normative rules which are accepted by people.

After the computational process, the next step is to calculate the features. A simple but very effective way is using histograms to accumulate features. The original idea of utilizing histograms in image retrieval was presented by Swain and Ballard [37], who realized that using color to identify the object is better than using only gray-values. Since a color histogram may lose all the information about the location of features, [37] aimed to restore the histogram back into the image through searching for best matches. Histograms are an efficient way for image retrieval especially when retrieving an object with a special color pattern. It is a robust measure for the translation of objects and rotation of viewing axes. The histogram can not only be applied in color patterns, but also combines local textures or local shapes [60], edges [61], and higher order structures [36] into it. An alternative is adding one more dimension describing distance into it, which is called a correlogram [62]. A different way to accumulate features is to store all the information of the image into feature values. This method requires appropriate compression transforms

suited for retrieval features. Visual Word extraction is another efficient way to construct descriptions from existing visual features. Basically, a Visual Word is defined as a cluster of visual features. Through mapping each visual feature to the visual vocabulary, we can describe the image as a feature vector according to the presence or count of each Visual Word.

Since strong segmentation is hard to achieve, sometimes weak segmentation can be used as a substitute. It indicates a group of image data in conspicuous regions that are homogeneous. Therefore, a selection is carried out based on their salience in the merged regions. In this case, only a limited number of feature values are used to describe the information of images. So, choosing appropriate information, which is robust and remarkable for description, is important. That is the why the lifetime [63] is an important criterion for selecting saliency in addition to other attributes including wagginess, spatial width, and phase congruency. In [64], firstly, a transformation from the raw pixel data to a small set of homogeneous regions in color and texture space was made. The Expectation-Maximization algorithm, which models the joint distribution of color and texture with a mixture of Gaussians, was used to perform automatic segmentation based on image features. The authors also demonstrated a query mechanism that uses that measure to retrieve images and help guide user queries. We can say an image holds a sign if one interpretation is strong enough to express the meaning of the image. Typical signs include a character, trademark, or icon. In [65], map symbols and the spatial relationships between them provided a way to understand the contents. There were many systems developed based on signs-detection, such as face detection [66], medical image [67] and art image [68] retrieval. Many comparisons of shape for retrieval are shown in [69], in which the features of a 500-element trademark data set were evaluated.

Another topic of feature calculations concerns the structure and layout of features. It reveals the fact that features and the relationships between them should be stored

together, when the same feature is available for different entities in the image. Generally, the descriptions can be stored in a graph, hierarchy, or any other ordered set of feature values and corresponding relationships. To achieve this, in [59], layout descriptions were captured in the form of a graph. In [70], a similar graph was used to store topological relationships of homogenous RGB-regions. This method has the advantage that the descriptions are invariant when selected features and the relationships of topology are invariant, but the RGB-representation used in this method cannot fully support this purpose. In [71], the system created models to express the relationships between regions that are characterized by locations, size and features. Matching based on the 2D-string representation was investigated by Chang and Hau in a later system [72].

Chapter 3

Ontology and Content-based Image Retrieval

This chapter presents a general framework for integrating domain knowledge into content-based image retrieval. Firstly, we provide an overview to the proposed framework. An initial experimental implementation is described after the overview to highlight the practical issues we faced in the integration. Secondly, we introduce an ontology system that encapsulates the domain knowledge for still image content descriptions. This ontology system is able to capture and organize the domain knowledge we extracted from Web page content in a highly structured format. Therefore, it enables the domain knowledge to be accessed efficiently by the image retrieval system. Next, we describe a content-based image retrieval system based on semantic knowledge building. This system implements the proposed framework and the ontology system. The experimental results on Canine image retrieval show that the proposed system can achieve higher performance by incorporating domain knowledge in image retrieval. Lastly, the open issues for seamless integration of ontology with content-based image retrieval system are discussed.

This chapter is organized as follows: Section 3.1 introduces a new framework for ontology-based Web image retrieval system, Section 3.2 describes the ontology design for Web image retrieval, Section 3.3 presents a Web image retrieval system based on semantic

knowledge building, finally, a discussion is provided in Section 3.4.

3.1 Ontology-based CBIR System

3.1.1 Overview of framework

In recent years, with the remarkable increase of images generated and distributed throughout networks, there is an evident need to develop an effective and efficient Web image retrieval system. The recent survey on Web search engine [73] shows 22 out of 65 general Web search engines support multimedia retrieval and there are 37 multimedia specialized Web search engines. However, all Web search engines still rely on file metadata and characteristics of the Website. Only 5 out of 102 examined Web search engines support content-based media search which is still very limited on low-level or mid-level features. Moreover, all search engines can handle semantic concepts which are limited as keywords instead of structured vocabulary. Similar to Web search engine, existing image retrieval systems use same cues to perform image matching and retrieval. Since the file property or metadata is not a reliable source for image content description, most systems utilize cross references and relevant feedback to improve their performance. This requires a lot of human interactions. On the other hand, the approaches based on visual features are limited by the descriptive power of visual features. It is difficult for such systems to handle high-level human conceptual items as input. To improve the overall performance of content-based Web image retrieval systems, we propose a method to combine domain knowledge with visual features and utilize both of them in image matching and retrieval.

Currently, there is no mature method of incorporating domain knowledge with image retrieval systems to achieve high-level image retrieval based on a user's single query. For example, one of the critical problems in CBIR is the mismatch between what such systems can provide and the needs of users. In this case, how to generate and manipulate machine-

processable representations for the semantics associated with images, are challenges faced by CBIR systems. The key issue here is, how to manipulate the two kinds of semantics that describe different aspects of multimedia data. They are feature descriptions for the image itself (content semantics) and content descriptions from the human conceptual aspect (concept semantics). In this case, we propose a framework to handle both content and concept semantics at the same time. This framework has four main components:

- A visual feature extraction and analysis module. This module takes Web images as input and it is able to extract visual image features from Web images. The visual image features are analyzed and summarized as standardized feature descriptors. According to domain definitions, a set of visual vocabulary is further extracted, which provides basic content descriptions for Web images. This module is designed to handle content semantics in the proposed framework.
- A domain knowledge extraction and analysis module. This module takes Web documents or other Web resources as input to generate domain concept descriptions. The descriptions include both concept definition and relationship descriptions. The visual vocabularies extracted from visual features are applied in concept descriptions to enhance descriptive power and thus associate content semantic with concept semantic.
- An ontology system. This ontology system encapsulates the content and concept semantics for a specific domain, which are extracted by the two previous modules. Such an ontology system facilitates knowledge sharing and reuse. Moreover, it enables the domain knowledge to be accessed easily.
- An image matching and retrieval engine. This engine is able to evaluate and rank Web images according to the content and concept semantics associated with the user's query.

By implementing such a framework in the Web image retrieval system, we receive benefits from both content and concept semantics. Firstly, the user's query can be a high-level human conceptual term. The image retrieval system can "understand" the query term by searching for the associated semantics in the proposed ontology system. Secondly, the content semantics enhance the image retrieval performance by describing high-level human conceptual items using visual features.

3.1.2 The First Attempt of Integration

To evaluate the proposed framework and to investigate the potential research issues, we implemented a simple natural scene image retrieval prototype system based on the proposed framework. In the system, we defined an ontology system to provide a multi-layer tree structure hierarchy description for image contents. This ontology system was built using Resource Description Framework (RDF) Schema and the DAML+OIL ontology language.

We defined three sub-ontologies in this prototype ontology: 1) The first sub-ontology system was the Ontology for MPEG-7. We specified the relationships between the features and descriptors in the ontology according to the MPEG-7 standard. For example, MPEG-7 visual descriptors were associated with their corresponding classes: Color, Texture, Motion, and Shape features. Since the semantic hierarchy is given in the MPEG-7 standard, the RDF statements used in the ontology strictly adhere to this standard. 2) The second sub-ontology was defined for the semantic description of multimedia content. The goal of this ontology was to describe the hierarchy representing the concepts of image contents, and define a schema that provides semantic interpretations of the concepts utilizing MPEG-7 visual descriptors. To map the domain knowledge of nature scenes into the ontology, human conceptual items were abstracted into several concept groups and relationships based on different abstract levels, and represented by DAML classes

and properties. The relationships between the elements defined in this ontology can be simply understood as: one *domain* consists of several *concepts*, one *concept* consists of several *objects* and *object* is described by some MPEG-7 descriptors. Based on the above definitions, a hierarchy description for domain: nature scenes was created. Facilitated by the language features of DAML+OIL, the above definitions and relationships were specified in a legible hierarchy in the ontology 3) The third layer was an instance for multimedia content description. This layer was an implementation of the previous two ontologies. That means, the instances describing the image contents are created in this layer based on the schemas predefined in the ontology. Such instances include not only the entities of abstract concepts but also the data of low-level feature descriptors. Through searching this layer, the retrieval system can obtain the associated low-level descriptors from the requested high-level concepts.

To manipulate content semantics, three MPEG-7 visual descriptors were implemented in the ontologies to describe the low-level features of image contents including Dominant Color Descriptor (DCD), Color Layout Descriptor (CLD), and Edge Histogram Descriptor (EHD). Furthermore, we trained a multidimensional gaussian distribution model from several examples for each defined object in the ontology. The trained models were utilized to provide content semantic descriptions for high-level human conceptual items. For domain knowledge extraction, all concepts and relationships between concepts were defined manually. In the ontology, we selected five types of nature scene images to make up the concept. They are “beach,” “winter scene,” “forest,” “grassland,” and “sunset.” Next, we defined the “objects” which are utilized to describe the above “concepts”. Currently, twelve objects including “clear-day-sky,” “cloud-day-sky,” “night-sky,” “grass,” “blasted grass,” “tree or forest,” “sand,” “clay,” “shadow,” “mountain,” “sea,” and “snow and ice” are defined in the ontology. For each object, 80 to 120 sample image blocks corresponding to the object were manually generated from randomly selected training images. After

that, the multidimensional gaussian distribution models were trained from these samples. The process of ontology-based image retrieval (matching and retrieval engine) can be described as follows: (1) the user keys in a word indicating the content of the images; (2) the search engine invokes a query expansion module that performs query expansion for the input keyword using WordNet; (3) the search engine invokes an ontology query engine. The concepts are passed to the ontology search engine after query expansion. Corresponding object description models are returned if relevant domain knowledge is defined in the ontology; (4) the search engine invokes an image retrieval module to match and retrieve Web images based on similarity matching.

To test the performance of this prototype system, we selected a diverse collection of 40 classes (5300 images) of outdoor images from Corel photo gallery as a test set. The images in Corel photo gallery were already classified as antelope, balloon, beach and so on. We also labeled the images manually for generating ground truth of specific concepts. The experimental results on image retrieval for three different types of sky showed that the overall 64.7%/63.4% recall/precision was achieved. Furthermore, for five different types of natural scene image retrieval, overall recall/precision of 69.2%/50.2% was achieved. The feasibility of applying the proposed framework into the CBIR system is demonstrated from the above implementation and experimental results. The main advantage of this system is that high-level human conceptual items can be associated with low-level visual features. This enables the machine to deal with such high-level query keywords directly. At the same time, the proposed ontology provides a methodology to manipulate and organize the domain knowledge. To further apply the proposed framework in a more practical Web image retrieval system, we focused on the following research issues:

- We needed to introduce more visual features into the proposed framework to enhance the descriptive power of Web image content description. By incorporating more effective and efficient visual features, the image retrieval performance can be

improved directly.

- We also needed to design a more general ontology system to handle more complicated content and concept semantics and the relationships between them. The prototype ontology system is not flexible enough to handle different types of descriptions and also it does not have sufficient capability to handle terms from different domains.
- We needed to have a method to extract or build semantic knowledge automatically. With an entirely manually-defined ontology, we cannot handle a large amount of query keywords in Web image retrieval. This limits the usability of the proposed framework.
- To achieve better performance of Web image retrieval, we needed to implement an image retrieval engine that receives benefits from both content and concept semantics. Specifically, the engine needs to leverage on the ontology system to have the capacity of reasoning.

In the following several sections, we discuss how to improve the implementation of the proposed system by trying to address the above issues. To evaluate the performance of the new implementation, we selected “Canine Animal” as our target domain.

3.2 Ontology for Still Image Description

From the prototype system implementation, we learned that the design of multimedia content description ontology is a crucial part for our proposed system. To manipulate the content and concept semantics for still image descriptions in a more general and formal structure, we improved the prototype ontology to capture and store such semantic information. Before we describe our ontology into details, we want to highlight several ex-

isting multimedia ontologies base on the MPEG-7 standard. The emergence of MPEG-7 ontology is trying to resolve the limitations of MPEG-7 that have been reported recently [74, 74] and further narrow the semantic gap that is mentioned in Section 1.2.1. In literature, the first attempt to build MPEG-7 ontology was made by Hunter [24]. This ontology has usually been utilized to describe the decomposition of multimedia entities and their visual descriptors for use in other frameworks. Tsinaraki et al. [75] extended the MPEG-7 ontology to cover the full Multimedia Description Scheme (MDS) part of MPEG-7, in which concentrate more on linking domain specific knowledge with multimedia ontology for video indexing. Garcia and Celma [76] have presented the Rhizomik approach for the first complete MPEG-7 ontology by mapping XML Schema constructs to OWL constructs. This facilitated the data conversion from the great amount of metadata produced by the XML community to ontology. Arndt et al. [77] have proposed the Core Ontology for Multimedia (COMM) for annotation. It was composed of multimedia patterns specializing the Descriptive Ontology for Linguistic and Cognitive Engineering (DOLCE) [78] design patterns for Descriptions & Situations and Information Objects. Moreover, it covered the most important part of MPEG-7 that is commonly used for describing the structure and the content of multimedia documents. In this section, we present an ontology which provides semantic interpretations extracted from low-level image feature descriptors of the image. This ontology is capable of structuring and defining the meanings of metadata for efficient and accurate search or navigation. Moreover, it is also capable of representing domain knowledge in different structural complexities. The above capabilities of this ontology allow the development of an ontology system that provides the semantic interpretations for high-level human conceptual terms. At the same time, in combination with standardized low-level feature descriptors e.g. MPEG-7 visual descriptors, we attempted to narrow the semantic gap by providing the semantic interpretation for image contents, utilizing low-level image feature descriptors. Compared with

existing MPEG-7 ontologies, our ontology focuses on the conceptual content description instead of decomposition of the image like Hunter's approach or describing the individual media entity like Tsinaraki's, Rhizomik, and COMM approaches. In this case, we do not cover as many parts of the MPEG-7 standard as other approaches. However, we provide a simple approach for using low-level and mid-level visual feature to describe a conceptual item. Different from COMM approach, in which the semantic information is generated from manually annotation for a media entity, the information in our ontology is generated from machine learning methods for an abstract concept. For Hunter's, Tsinaraki's and COMM approaches, they are able to be integrated with other domain semantics that are very important components in our proposed ontology system. For such domain or context description ontology, we aim to provide good conceptual clarity and rich axiomatization for media content. Meanwhile, we should be able to adapt the existing online resources and convert them into ontology to reduce ontology building workload.

Our proposed ontology system facilitates the semantic interpretations for still image contents in a top-to-bottom manner. Firstly, a domain was defined, which is the aggregation of abstract concepts. Each abstract concept consists of one or more visual middle-level concepts which correspond to the homogeneous region(s) on image feature spaces. These middle-level visual concepts are further described by a set of low-level image feature descriptors. Apart from the visual middle-level concepts derived from low-level image features, a set of descriptive properties was also defined, which provided narrative descriptions for domain abstract concepts. Using the proposed ontology, the concept semantics in different complexities and the content semantics were associated and structured as a tree-like hierarchy. Through searching and querying the ontology, users can retrieve the formal descriptions for high-level human conceptual items and also the relationships between them. Moreover, the relevant domain knowledge defined in the proposed ontology provides a wealth of information, and thus enables the application to

have a better understanding for query terms.

3.2.1 Overall Structure

This is an overview for the improved ontology system. In recent reports [14, 79], concept hierarchies and ontology have been proved as a efficiency for data summarization of textual information. Several approaches [80, 81] have been proposed for representing the media contents in a hierarchy structure. One shortcoming of these approaches is the lack of knowledge sharing and reusing. The concepts defined in these systems are not easily accessible by other programs or applications. On the other hand, ontology encodes knowledge within a domain as well as knowledge that spans domains, which makes such domain knowledge reusable [82]. The concepts in ontology, once defined, can then be utilized in other definitions or ontologies. Similarly, we can also import the definitions or information from other ontologies. Facilitated by the descriptive power of Web ontology language (OWL), a formal ontology for still image content descriptions can be built, which provides semantic interpretations (multi-layer hierarchical descriptions) for image contents. The semantic interpretations provided by the proposed ontology system include the low-level descriptions for the appearance of high-level human conceptual items and the relevant domain information. This enables machines to identify human conceptual items, which are user input keywords, by analyzing low-level feature descriptors. At the same time, the machines can achieve better interpretation of the semantic meanings of the query terms by incorporating the relevant domain information. Here, the relevant domain information contains the descriptions which comes from: 1) the context information relevant to the term itself and 2) the correlations describing the relationships between the terms in different abstract levels. By utilizing the proposed ontology in an image indexing or retrieval system, the system is able to “understand” the meanings of human conceptual items and therefore narrow the semantic gap.

The ontology in our current system, was built using the OWL ontology language. Figure 3.1 shows the layer structure of the ontology. Currently, the ontology has three

Figure 3.1: Layer structure of Ontologies.

components. Different components of the ontology system are connected using import statements provided by OWL. That is, an import statement enables different ontologies to exchange information and definitions. Each import statement references another ontology through a URI, which specifies where the ontology is to be imported from. Hence, definitions included in the references can be applied to current OWL resources.

In Figure 3.1, there are three blocks indicating three sub-ontologies that constitute our multi-modality ontology. The first sub-ontology is the Domain Ontology, which contains a hierarchical tree structure description for a specific domain based on the formal definitions of domain knowledge. The relationships between two concepts are represented by their relative positions in the hierarchical tree structure. Moreover, the different layers in this hierarchical tree structure also reflect the different abstract levels of concepts. In this case, Domain Ontology enables retrieval of not only a single concept category but also

related concept categories.

The second sub-ontology is the Textual Description Ontology which encapsulates narrative domain descriptions. Normally, narrative domain descriptions can be any document written in a human natural language. However, as the natural language process (NLP) still faces the obstacle of precisely mapping human high-level domain descriptions directly to an ontology, for this part, concept semantics are collected from the domain knowledge (documents) prepared by domain experts. Moreover, the narrative domain descriptions here indicate a special category of the concept semantics which cannot be described by any low-level or middle-level visual features. Therefore, the concept semantics (both resources and properties) defined in this ontology are purely on a textual level.

The last sub-ontology is the Visual Description Ontology. This ontology is constructed from high-level concepts derived from low-level visual features. The goal of this ontology is to manipulate the correlations between the concept semantics that can be described by corresponding content semantics. The concept semantics defined in this sub-ontology can be described by content semantics such as any low-level or middle-level feature. At the same time, the concept semantics can also be understood by humans because they have prior knowledge which is generated from the human understanding of the objective world. To manipulate the concept and content semantics in this sub-ontology, the proposed ontology provides tree-like hierarchical descriptions and semantic interpretations for high-level human concept terms in a top-to-bottom manner. Utilizing this hierarchical structure, we can get the domain relevant middle-level description for a human high-level term. At the same time, the low-level features associated with middle-level features allow us to automatically extract or generate middle-level features. This helps to narrow the semantic gap in the scenario of still image retrieval.

3.2.2 Define and Build Ontology

In [12], authors present a ontology construction mechanism to extract domain ontology from unstructured text documents. In this subsection, we provide the detailed information for creating the three sub-ontologies in multimedia domain. The information includes the formal definitions for the main classes and properties utilized in the ontology system, and also how to derive the terms and properties in the ontology.

Domain Ontology

The definitions of a specific domain can be extracted or imported from other resources such as taxonomies provided by online dictionaries. Since this ontology is a domain specific ontology, terms, properties, and the structure of the ontology totally depend on their relevant domain knowledge. The tree structure describes the concept semantics using classes, and the relationships between them are described using subclass properties which is shown in Figure 3.2. The *class:Domain* is the parent node for all types of

Figure 3.2: General structure of Domain Ontology.

domains, and *class:Concept* indicates all types of high-level human conceptual items under a certain domain.

According to the definition of this Domain Ontology, it is easy to derive from the

taxonomy of a specific domain. The taxonomy of a specific domain provides us a basic classification of organisms into groups based on similarities of structure or origin. This classification can be directly mapped into Domain Ontology because the subclass relationship is also a basic description in the classification. Other than the formally defined taxonomies like zoological and botanical taxonomies, Domain Ontology can also be built from other resources such as an online dictionary/thesaurus or encyclopedia. WordNet [83] is a lexical reference system which organizes English nouns, verbs, adjectives and adverbs into synonym sets, each representing one underlying lexical concept. Our Domain Ontology was easily built by utilizing WordNet. The hyponyms relationship defined in WordNet was refined as a sub-class relationship in our ontology. Therefore, the terms defined in WordNet can also be imported into an ontology as classes. The Domain Ontology can be extended or modified easily. Since *rdfs:subClassOf* has the feature of transitivity, any new class can be added into the hierarchical structure at any time. We only needed to disconnect two classes and then set original parent class as the parent class of the new added class and set original child class as the child class of the new added class. Moreover, we didn't need to import the whole taxonomy from other resources, the unnecessary classes were removed from taxonomy structure first and then imported back later, if necessary. This made the ontology relatively light-weight, yet sufficiently informative.

Textual Description Ontology

As mentioned in Section 3.2.1, the concepts and relationships defined in Textual Description Ontology are extracted from the narrative domain descriptions prepared by domain experts. The general definition for Textual Description Ontology is presented in Figure 3.3. By using the resources shown in above figure, the Textual Description Ontology can provide the most basic narrative description for a domain concept. The mapping

Figure 3.3: General structure of Textual Description Ontology.

from narrative domain descriptions prepared by domain experts to this ontology is quite straightforward:

- Step 1.** Find the resource which describes the domain concept defined Domain ontology.
- Step 2.** Import this concept into Textual Description Ontology by using the URI of the domain concept as a subclass of *class:DomainSubject*.
- Step 3.** Extract the verb or predicate construction in the narrative description, a corresponding property is created as a sub-property of *property:hasDescription*.
- Step 4.** Extract the object which is normally a noun or noun phrase in the narrative description, a corresponding class is created as a subclass of *class:SubjectDescription*.
- Step 5.** A statement which contains the newly defined classes and properties is created in the ontology.

The building of Textual Description Ontology requires good understanding of the targeted domain. Therefore, simply using the normal NLP process is not enough for automatic ontology building of Textual Description Ontology. Moreover, it is not easy to extract domain relevant descriptions directly from general Web documents if they contain too many adjunct words. In this case, most likely, the Textual Description Ontology is

built by domain experts directly or reconstructed from well-formatted domain description documents. To reduce the workload for ontology building, a Textual Description Ontology builder was implemented to automatically extract the concept semantics from well-formatted domain description documents. There are many domain description documents available online, such as domain specific dictionary ¹, thesaurus, or encyclopedia ². Firstly, a documents parser for a specific document format was created to extract the concept semantics out of these documents. The parser extracted semantic concepts based on a user defined scheme, designed according to the format of the domain description document. Then, the extracted semantic concepts were utilized in the proposed ontology. To further enhance the descriptive power of the proposed ontology, we also defined a *property:hasConfidence*, This property reflects the confidence level of a domain description. The *property:hasConfidence* was defined as having a float value in the range of 0 to 1. Value 0 indicates the minimum confidence level while value 1 indicates the maximum confidence level. The different *property:hasConfidence* values were set according to where the domain descriptions were extracted. This enables the ontology to deal not only with exact but also fuzzy domain descriptions. During the creation of this ontology, some other OWL language features were also be utilized to provide more accurate domain descriptions.

Visual Description Ontology

The Visual Description Ontology provides semantic interpretations for high-level human conceptual items by using low-level visual features. The general structure of the ontology is shown in Figure 3.4. In this ontology, we first introduce *class:Visualsubject* and *class:Visualconcept*. *class:Visualconcept* was defined as the parent class for all semantically meaningful concepts which were described by using low-level features. A visual

---

¹BBC Science & Nature Animal Category: http://www.bbc.co.uk/nature/wildfacts/animals_a_z.shtml

²Wikipedia: http://en.wikipedia.org/wiki/Main_Page

concept is semantically meaningful to humans and is homogenous in a certain feature space and thus, can be described and distinguished from other visual concepts by using low-level visual feature descriptors. In this case, any middle-level feature in the domain of image processing can be defined as a visual concept. To maintain the semantics across different low-level features, we first introduce another important ontology that is a part of Visual Description Ontology, that is, the MPEG-7 Ontology.

The motivation for creating this ontology was to provide a semantic description for

Figure 3.4: General structure of Visual Description Ontology.

MPEG-7 visual descriptors [24]. This ontology is created with OWL and the relationships

between the features and descriptors are specified in the ontology. For example, MPEG-7 visual descriptors are associated with their corresponding subclasses: *Color*, *Texture*, *Motion*, and *Shape* features. If a user wants to retrieve images using color features as the retrieval conditions, the ontology can tell a retrieval system how many color feature descriptors are defined and can be used. Accordingly, the retrieval system gets the hierarchy of visual descriptors from the ontology to make MPEG-7 accessible, reusable and inter-operable with other domains. With this ontology, the MPEG-7 descriptors become meaningful, which can then interact with humans or other programs instead of being used only as computational data. Since the semantic hierarchy is given in the MPEG-7 standard, the RDF statements used in the ontology strictly adhere to this standard. Therefore, in building the MPEG-7 Ontology, the definitions of the standard were followed and the necessary classes and properties for image content description were created. However, there are many other visual feature descriptors that are not included in the MPEG-7 standard. To allow the *class:Visualconcept* to access those visual feature descriptors, additional feature descriptors are also defined in the MPEG-7 Ontology according to what type of feature they are described for. For example, *class:ColorCorrelogram* is defined as a subclass of *class:ColorDescriptor*. Moreover, some additional properties were created to provide more detailed description for low-level feature descriptors. For example, *property:hasAreatype* is defined to describe a feature that is a local (*class:Localfeature*), global (*class:Globalfeature*), or point (*class:Pointfeature*). If a feature descriptor is defined as a local feature, an additional property *property:Localwindowsize* will also be defined to provide the information of preferred local window size for a specific feature.

To build Visual Description Ontology, the domain experts firstly define a set of visual concepts which are domain relevant middle-level features. After that, they decide which visual concept (middle-level feature) should be used to describe a specific visual subject. The definition of visual concepts and relationships between visual concepts and visual

subjects can also be derived from the standardized image categories or image databases. The classification information of these image categories/databases provide a good guideline to define the proposed ontology. The next step in ontology building is linking a set of low-level feature descriptors defined in the MPEG-7 Ontology with the visual concept. At this stage, the supervised learning manners are followed. Firstly, a set of images is manually annotated according to the definitions of visual concepts. After that, the low-level feature descriptors are extracted from these images. Lastly, the low-level feature descriptors extracted from these images containing the same visual concepts are labeled. These are then linked to corresponding visual subjects.

3.2.3 Examples and Implementation

In this section, we use a series of examples demonstrate how to use the proposed ontology system to define a high-level conceptual concept by utilize both textual and visual information. In these examples, we defined the domain of ontology as, animal description. By utilizing the ontology system, an image retrieval/indexing system can derive both narrative and content descriptions for a query term, or a high-level concept level item. This improves system performance by narrowing the semantic gap. Figure 3.5 shows an overall example for the prototype ontology system. At the top part of the figure, each segment indicates a sub-ontology structure. On the lower part of the figure, two individuals are created to demonstrate how the domain relevant knowledge can be retrieved from the prototype ontology.

The top-left segment of Figure 3.5 provides an example for Domain Ontology under the domain of animal description. It describes the relationships between canine animals according to animal taxonomy provided by WordNet. The top-middle segment of Figure 3.5 shows an example of Textual Description Ontology. As mentioned in Section 3.2.1, the concept semantics encapsulated in this ontology were extracted from well-formatted

Figure 3.5: Example of ontology system: ellipse and rectangle represents class and instance respectively.

domain descriptions. Here, we selected the online encyclopedia: BBC Science & Nature Animal Category ³ as the domain description knowledge base. This Science Category provides standard descriptions for around 620 animals. For each animal, it provides a list that contains a set of properties and corresponding descriptions. Therefore, we were able to extract the concept semantics such as distribution, habitat, scientific name and so on out of this encyclopedia and defined them as semantic relationships, such as: *property:hasDistribution*, *property:hasHabitat*, and *property:hasScientificName*. At the same time, as these properties and descriptions were extracted from authoritative resources, we gave them the maximum confidence level. For other properties that were extracted from empirical documentation or indicated partial authentic domain relevant information, a lower confidence level was assigned to these properties. For example, *class:Chimp* was defined to have *property:hasRelevantPeople* with the value of *class:PeopleJaneGoodall*. For an image which is annotated as relevant to Jane Goodall, it is highly possible that it also be annotated as containing animal chimp. However, this kind of inference is not always correct. Therefore, a lower confidence level such as 0.6 was given to this type of properties. The top-right segment of Figure 3.5 shows an example for Visual Description Ontology. In the figure, server classes were defined as the subclasses of general *class:Visualconcept* to provide semantic interpretations for *class:VisualSubject*. For each *class:Visualconcept*, a set of low-level features extracted from images or image segments were linked to it to describe its visual features.

By matching the information, we can extract from both image and textual content, even if the exact matched query keyword can not be found, the best matching semantic interpretation, according to the knowledge encapsulated in the ontology can be found. Take for example, an anonymous image concept *X*. It is known that a canine *hasDistribution* of Asia and *hasFur* of brown color. From this information, it can be inferred that

---

³http://www.bbc.co.uk/nature/wildfacts/animals_a_z.shtml

this is a Redwolf and not a Redfox, because Redfoxes only appear in America. Therefore, the domain knowledge captured by the ontology can facilitate the image retrieval by providing more accurate semantic interpretations for high-level human conceptual items.

3.3 CBIR System with Semantic Knowledge Building

Section 3.2.3 introduced our prototype ontology system which provides semantic interpretations for image contents by using both low-level visual feature descriptors and high-level narrative descriptions. This section further describes how to adapt this kind of domain knowledge encapsulated inside an ontology in an image retrieval system. As mentioned in the last section, the standardized MPEG-7 [1] and other feature descriptors provide us a methodology to manipulate *content semantics*, and the prototype ontology is used to manipulate the *concept semantics* of image content. This helps to narrow down the *semantic gap* in an image description scenario. To apply the domain knowledge in image retrieval, unlike the traditional semantic annotation method which is carried out manually, we propose a new method to build up semantic knowledge for image database by automatically analyzing Web page contents (textual information). Combining the semantic knowledge and useful low-level feature descriptions, our Web image retrieval system benefits from both content and concept semantics, which facilitates direct Web image retrieval using high-level descriptions.

3.3.1 Related Work

In this subsection, we want to highlight several existing image retrieval systems that have domain knowledge integrated. Nah et al. [84] have presented a visual media retrieval framework which is built on basis of Web service. An ontology based metadata of image annotation was created and integrated into media retrieval framework. The query

term was automatically broken into more detailed semantic meaningful items according to ontology definition, and therefore, helped retrieval system to produce more accurate retrieval results. Dasiopoulou et al. [85] have introduced two general approaches for improving multimedia retrieval and composition by using ontology technology. However, for both systems, they highly rely on accurate image annotation information. Hai et al. [86] have proposed a medical image retrieval system based on both content and semantic information. Several visual features including shape, color, and texture features were utilized in this system to achieve higher retrieval precision. The system also utilized metadata to enhance the retrieval results by providing semantic information on both query object and search process. Hai's system utilized visual description and semantic information in media retrieval, but it did not provide axioms for concept definition. It is hard to apply this framework on general image retrieval because of the non-standardized visual features and limited semantic descriptions. Feng et al. [87] introduced a framework for annotating and retrieving Web image by defining concepts using machine learning approaches based on both textual and visual descriptions. This framework did not provide any method to manage semantic information and concept definitions. We need to re-run the classifier if we need to add more concepts into this framework which make the semantic information not reusable. Antonio et al. [88] have presented an ontology-based image retrieval system, in which image segmentation and classification were utilized to associate visual descriptor and conceptual items. This system provided the axioms for concepts, however, the contextual information was very limited and need to be populated manually. EU IST AceMedia integrated project [25] focused on a framework to make content more relevant to the user; to automate annotation at all levels; and to add functionality to ease content creation, transmission, search, access, consumption and re-use. In this project, ontology was utilized to support content processing, knowledge assisted analysis and inference. Comparing with these image retrieval systems, our proposed system integrate

ontology to provide both concept and content descriptions, which is different from Hai's and Feng's approaches. Moreover, we define conceptual items by using basic visual concepts that is reusable and can be generated from automatically machine learning process, which is different from Nah's, Dasiopoulou's, and Feng's approaches. Apart from that, our contextual information can be extracted from online resources, which provides more complicated and restricted axioms than Antonio's approach. Compared with AceMedia project, we investigate into the different method for summarizing Web textual information and integrate them into ontology system. We also experiment different approaches for matching image in context-aware environment to produce final retrieval results. We describe the details of our system design and implementation in following subsections.

3.3.2 System Design

From the point of view of information retrieval, each document has its own topic or meaning, and these can be analyzed and extracted automatically. As a Web page describes or concentrates on a certain topic, it can be a semantically meaningful document. Therefore, the same technologies applied in the information retrieval can be directly adapted on analyzing the Web text content. At the same time, Web page content includes not only the text content but also the multimedia content. Through investigating the relationships between the text content and multimedia content, we found a large amount of Web page contents that provided us rich semantic information describing Web images. Such information provides not only content but also context description for semantic meanings of images. Therefore, semantic Web image retrieval can be achieved through searching the semantic knowledge built from Web page contents. Since the descriptions for the images in our semantic knowledge are semantically meaningful terms, users can directly retrieve the images using high-level query terms and overcome the semantic gap encountered in traditional methods. To extract that information accurately, we proposed

and implemented various methods to extract the semantic meaning of Web page content, and further built up semantic knowledge for the image database.

The semantic knowledge extracted from Web pages provides a feasible method to generate high-level descriptions for image content. However, the high-level descriptions generated through Web page content are not always reliable due to the loosely-coupled relationships between the images and Web page contents. In this case, the high-level image content descriptions derived directly from the images were a helpful reinforcement for improving the precision of high-level image content descriptions. To extract the high-level content descriptions from images, a lot of effort was made in approaching the high-level image contents through the analysis of low-level image features. Various systems have been developed for retrieving images based on low-level attributes, such as Photobook [89], Visualseek [90] and MARS [91]. These systems perform image retrieval by matching the features of a query image with that of the database documents. However, they do not provide enough semantic interpretation for high-level human conceptual terms. Users cannot retrieve images directly using high-level query terms. In this case, extracting the semantic interpretations from image content becomes a challenge. During the semantic interpretation extraction from low-level features, we faced a problem, which was how to manipulate the correlations between the high-level concepts and corresponding low-level features. It is known that the meanings of high-level concepts are domain specific. A term may have different meanings under different scenarios or domains. Moreover, the high-level relationships between the different terms that are natively defined by human knowledge are important components of semantic interpretations of image contents. In this case, we needed an appropriate mechanism to capture and store such information in a highly structured format, and also required the information to be modified and retrieved easily. Therefore, we introduced the prototype ontology system into the image retrieval system. Through the prototype ontology system, the high-level

Figure 3.6: Overall system structure for CBIR system with semantic knowledge building.

conceptual terms were associated with the corresponding low-level features. At the same time, the relationships between the high-level conceptual terms were also described in a tree-like structure which enables the ontology to provide not only the description for the term itself but also the description for the correlations between them. In addition, the ontology has the capability to capture the domain knowledge which are the high-level descriptions to the context of a conceptual term. Combined with such reliable concept semantics, we could potentially improve high-level image retrieval, and further reduce the semantic gap.

Figure 3.6 shows the overall structure of the CBIR system with semantic knowledge building. At the top of the figure, the first module in the system is the Web Information Collector. This module is designed to automatically collect Web textual information along with the Web multimedia content. That was achieved by implementing a Web image crawler. The crawler program travels every page under a certain URL recursively and downloads Web images/pages according to predefined criteria. After that, there are three modules in parallel. They are Domain Knowledge Building, Textual Context Analysis and Semantic Interpretations for Image Contents. Basically, the Domain Knowledge Building is the same process as ontology building described in Section 3.2.3. The next module is the Textual Context Analysis module, which handles the semantic information extraction out of Web pages or other textual documents. The last module is the Semantic Interpretations for Image Contents module, which generates middle-level image descriptions by utilizing low-level visual feature descriptors. At the lower part of the figure, there are two modules: the Semantic Handling Module and the Image Retrieval Module. The Semantic Handling Module is used to merge the multi-modality information collected and then matched with the information from user queries at the semantic level. The Image Retrieval Module is used to deliver retrieved images to end users.

3.3.3 Semantic Knowledge Building

This subsection discusses the semantic knowledge building system, based on detailed analysis of Web page contents from the following aspects: we first describe the Web image crawler and the database design. After that we explain how the dictionary is generated, discuss the construction of feature vectors using TF/IDF, and also describe Web content analysis based on the similarity matrix and path distance generated by using WordNet [83].

Before starting on the description of this component, two assumptions are first listed; they are the basis for the design of the semantic knowledge building system.

- The Web images are described by the Web text contents in the same page.
- The semantic meanings of Web image contents are closely associated with the semantic meanings of the Web text content.

Based on these two assumptions, we attempted to solve the problem of image content analysis by analyzing the text context, which are also the Web textual contents. Web text contents are written using a human natural language. In our opinion, if we could extract the semantic meaning of a Web page directly from Web text contents, and further infer the semantic meaning of the Web image, we could also narrow the semantic gap in the area of CBIR.

Image Crawler and Database design

Let us examine a typical Web page first. Figure 3.7 shows an example of a typical Web page.

At the top of the figure, is the page_URL and page_title which provide a summary of the contents of the Web page. Under the page_title, the anchor text provides a description

Other contents: Meta Description, Meta Keywords

Figure 3.7: An example of typical Web page.

for each link. There is an image at the center of the figure, which may be of interest to the user. Around this image is an area labeled with a dashed line. The text inside this area is labeled “Surrounding text of the image,” which provides the description for the image content. The text inside the broken line is labeled “Plain text,” which includes all texts inside one Web page. Other than contents that are visible on the page, a Web page may contain meta-keywords and meta-descriptions that are written inside the head section of the html file. These Web page contents were used as the raw data in the semantic knowledge building of the Web image database. To extract the above page content, we assumed that the Web pages conformed to the World Wide Web Consortium (W3C) standards such as, having meta-contents and using standard HTML codes.

An image crawler program was implemented as a final year project [92] with help from undergraduate students for collecting the Web images and page contents. The crawler program downloads Web images and pages based on the defined project. By

Figure 3.8: Downloading interface of image crawler program. A) project information and download status, B) system status, and C) database summary.

entering single Web links in individual projects, a user initiates the Web image crawler to download the Website from a single link, as well as expand and branch to other Websites. Web pages and images were downloaded and stored locally so that the Web page contents can be analyzed later. Images in JPEG or GIF formats were downloaded only if their sizes are between 200×200 pixels to 4000×4000 pixels, or some other user-defined range. This helps to exclude images of menu bars, icons, buttons and advertisements. In addition, images from advertisements can be filtered out effectively by checking the number of times they are downloaded. Images that occur more than 2 times during downloading are very likely to be repetitive and would be excluded. Figure 3.8 shows the download interface of the image crawler program. In the figure, the top-left frame (Section C in the image) is the database summary which displays information about the local database, including the current database name, the quantity of: projects set, pages and images downloaded, and pages/images remaining in the download list. The top-right frame (Section A in the

image) displays the project information and downloading status, including the current project information (project ID, project name and project URL), quantity of pages and images downloaded in the current session, the runtime of the program, and the status for each downloading thread. This image crawler interface provides great flexibility for controlling the downloading process. The user can define the number of links the crawler program should branch out to, how long the crawler will run, and the number of threads that will be run concurrently. The crawler program also provides a thread checking mechanism which can abort the dead-lock and timeout threads automatically.

Once the base URL is entered by the user, the image crawler program will crawl and store relevant hyperlinks of images and pages from various sites into the database. With the Web pages downloaded, we then stored into our database the relevant data comprising image sources, relations between pages, titles, meta-keywords, meta-descriptions, anchor texts of links, text surrounding image links, as well as plain text, in the Web pages.

Textual Context Analysis

Web page analysis consists of 3 main parts: 1) Dictionary Generation, 2) Feature Vector Representation, and 3) Word Semantic Similarity Measurement Analysis.

Dictionary Generation: While there are no specific standards on how Web images are labeled, most Web pages use anchor links and descriptive text are placed immediately together with the image. By capturing this information, we effectively drew out the keywords associated with the images. Also, an HTML parser was used to filter off irrelevant HTML codes and capture only plain text in the document.

The strings of words extracted were parsed using WordNet, which captured the various forms of words such as nouns, verbs, adverbs and adjectives. As we were concerned with retrieving images of entities such as places, objects and people, only words in the noun and verb forms were retained to construct the dictionary. To limit the dictionary size, words

that were found in nearly every Web page were excluded using a document frequency threshold.

Feature Vector Representation: To effectively represent the text information in each Web page, we made use of feature vectors based on Rocchio algorithm TF/IDF [93]. Given a Web page, firstly, we used the page-title, meta-keywords, meta-descriptions, anchor texts of links, and surrounding text of images in the page to build up the page data. The surrounding text extraction was achieved by looking at the text before and after the image URL, for up to 300 characters. Subsequently we calculated the TF/IDF for each word in the page data using the equation:

$$tfidf(t_k, d_j) = \#(t_k, d_j) \cdot \log \frac{|D_s|}{\#D_s(t_k)} \quad (3.3.1)$$

where $\#(t_k, d_j)$ denotes that the number of times word t_k occurs in the page d_j , $|D_s|$ is the cardinality of the set D_s of all pages under one project, and $\#D_s(t_k)$ denotes the number of pages in which the word t_k occurs. Finally, we converted the page data into a feature vector. The vector length is the size of the dictionary and its components are the TF/IDF values of each word. The feature vectors of all the Web pages were constructed in the same way.

Word Semantic Similarity Analysis: In the CBIR system with semantic knowledge building, the ontology system introduced in Section 3.2 is used to provide domain knowledge and contextual information. To evaluate the performance of the prototype ontology system versus other textual aware methods, and we present two methods that are capable of improving high-level word-context abstraction. By using these two methods, the Web image retrieval system is capable to incorporate textual information without any domain background. The word similarity extracted from dictionary is used to evaluate the semantical similarity between two concepts. In the first method, we took pairs of words from the dictionary and computed their similarity values using the WordNet API. These similarity values were then used to construct a word semantic similarity matrix (WSSM)

M which is a square matrix of the same dimension as the dictionary. In our experiments we named this matrix WSSM1.

To obtain the semantic similarity values between a pair of words, we first implemented the method provided by the JavaSimLib toolbox [94] which is an extension project using WordNet. The words were compared and a similarity value, which is a double-precision value that lies in the interval $[0, 1]$, were returned by the toolbox (which measures the similarity of two words). For example, ‘Car’ and ‘Automobile’ would have the similarity value 1.0 as each clearly defines the same object as the other. Likewise, words ‘Coast’ and ‘Shore’ have a similarity value close to 1.0 as they share very similar meanings. On the other hand, ‘Coast’ and ‘Forest’ are different in meaning and have a similarity value close to 0. Since we wanted to identify words that have similar semantic meanings, we filtered out similarity values that were lower than a threshold α . In this way, the query terms entered by the user were expanded to include other semantically similar words in the dictionary. For example, if a user queries a topic on ‘Monkey,’ the following words like ‘Chimp,’ ‘Chimpanzees,’ ‘Primate,’ ‘Gorilla,’ ‘Ape,’ ‘Gibbon,’ ‘Orangutan,’ or ‘Orang,’ would also be considered relevant.

One shortcoming of the above approach (WSSM1) is that for given words t_k and t_j the similarity $S_v(t_k, t_j) = S_v(t_j, t_k)$, which means the order of influence is not considered in the semantic similarity. For example, $S_v(\text{‘monkey’}, \text{‘animal’}) = S_v(\text{‘animal’}, \text{‘monkey’})$. In this case, for the query keyword ‘animal’, ‘monkey’ is a correct inference, but ‘monkey’ $\rightarrow$ ‘animal’ would be an over-expanded inference and it can lead to many irrelevant returns in the query results. This prompted us to propose an alternative method.

In the second method, we constructed the matrix M in a slightly different way. We named this matrix WSSM2 in our experiments. Using WordNet, we obtained a different type of information that also measures the semantic similarity between words. That information contains a common parent index I and the depths of different nodes or

words D_p , which share the same common parent and have different levels. Based on this information, the semantic relationships between two words can be summarized into these four types:

- *Synonyms*: If words t_k and t_j are synonyms, then $I(t_k, t_j) = I(t_j, t_k) = 0$ and $D_p(t_k, t_j) = D_p(t_j, t_k) = 0$.
- *Child-parent relationship*: If word t_k is the child of t_j , then $[I(t_k, t_j) = D_p(t_k, t_j)] > 0$, and the value of D_p or I shows how different the semantic meanings of (t_k, t_j) are.
- *Parent-child relationship*: If word t_k is the parent of t_j , then $I(t_k, t_j) = 0$, $D_p(t_k, t_j) > 0$ and the value of D_p shows how different the semantic meanings of (t_k, t_j) are.
- *Sibling relationship*: If word t_k is the sibling of t_j , then $I(t_k, t_j) > 0$, $D_p(t_k, t_j) > 0$ and the combination of D_p and I shows how different the semantic meanings of (t_k, t_j) are.

$I(i, j)$ is calculated from the distance between node i to node x which is the common parent for i and j , assuming the distance between children and direct parent node is 1. $D_p(i, j)$ is calculated from the distance between nodes i and j , assuming the distance between children and direct parent node is 1. For each type of relation, the combined values of I and D_p were quantized into a value in the interval $[0, 1]$. This value was obtained for every pair of words in the dictionary and the values were used to construct the matrix WSSM2.

3.3.4 Image Attributes Description

Other than analyzing Web textual information, we also extracted semantics from images by analyzing the low-level feature descriptors which provide content descriptions of

images. In this system, several types of low-level features including color and texture features are extracted from images to describe image content. Firstly, we give an overview of these low-level feature descriptors. There are five MPEG-7 visual feature descriptors utilized in this system. They are Dominate Color Descriptor (DCD), Color Layout Descriptor (CLD), Color Structure Descriptor (CSD), Edge Histogram Descriptor (EHD) and Homogenous Texture Descriptor (HTD).

DCD provides an effective, compact and intuitive description for representing color in an image or image region [1]. DCD is defined as:

$$F_{DCD} = \{(c_i, p_i, v_i), s\}, (i = 1, 2, \dots, N) \quad (3.3.2)$$

In the equation, N represents the number of dominant colors. The dominant color value c_i is a vector of component values corresponding to a specific color space. The percentage p_i is the proportion of pixels that have the associated color c_i in the image or image region, and $\sum_i p_i = 1$. The color variance v_i indicates the variation of color values of the pixels in a cluster around a corresponding dominant color. The spatial coherency s represents the overall spatial homogeneity of the dominant colors within the whole image. In our system, DCD was utilized to capture the global color distributions for image contents.

CLD is a very compact and resolution-invariant representation for the spatial distribution of colors [1]. CLD is defined as:

$$F_{CLD} = \{DY, DC_b, DC_r\} \quad (3.3.3)$$

In the above equation, DY, DC_b, DC_r represent the DCT coefficients corresponding to color components in YC_bC_r colorspace. To generate these coefficients, each image is, at first, divided into 8×8 equal partitions, where each partition is known as an image block. Then, the representative color is extracted from each image block. Lastly, each of the three color components is transformed by 8×8 DCT. We used this descriptor to describe the spatial distribution of color for specific objects. The semantic of position information

is captured by this descriptor. For example, in most cases, the feature descriptors for the higher-part of the image are homogenous if they all contain the same visual concept of *sky*, while the feature descriptors for the lower-part of the image are homogenous if they all contain the visual concept of *sand*

CSD is a color feature descriptor that captures both color content and the structure information about the specific content. CSD is defined as:

$$F_{CSD} = \{h_0, h_1, h_2, \dots, h(i-1), h(i)\}, (i = 32, 64, 128, 256) \quad (3.3.4)$$

In the above equation, CSD is an X dimensional histogram, in which X is decided by color space quantization operating points. CSD is generated by sliding a 8x8 pixels structuring element over the image. The number of times a quantized color value presented in this structuring element is calculated as bin values. The major functionality for CSD is image-to-image matching. Unlike from conventional color histograms, CSD can be used to distinguish two images in which a given color is presented in identical amounts, but the location of the structure of the groups of pixels having that color is different in the two images.

EHD is designed to describe the local-edge distribution in images [1]. EHD is defined as:

$$F_{EHD} = \{h(0), h(1), h(2), \dots, h(77), h(78), h(79)\} \quad (3.3.5)$$

In the above equation, $h(i)$ ($i = 0, 1, \dots, 79$) indicates the relative population of a specific type of edge in one image block. To generate this descriptor, firstly, the image space is divided into 4×4 non-overlapping and equal blocks. Then five types of edge histograms — vertical, horizontal, 45° diagonal, 145° diagonal and nondirectional edges — are computed in each image block. We used EHD to capture the global spatial distribution of edge for image contents.

HTD is designed for searching and browsing through large collections of similar looking

patterns. It is defined as:

$$F_{HTD} = [f_{DC}, f_{SD}, e_1, e_2, e_3, \dots, e_{30}, d_1, d_2, d_3, \dots, d_{30}] \quad (3.3.6)$$

HTD is extracted by filtering target image with a bank of orientation and scale tuned filters using Gabor filters. The number of filters used is $5 \times 6 = 30$ where 5 is the number of “scales” and 6 is the number of “directions” used in the multi-resolution decomposition using Gabor functions. After that, the mean value f_{DC} and standard deviation f_{SD} of the original image, as well as the energies e_i and their standard deviations d_i of the Gabor filtered image are extracted to form the feature vector. In our system, HTD is used to capture similar texture patterns in images.

Besides the MPEG-7 visual descriptor described above, we also utilized some other image feature descriptors in our system. They are: Color Correlogram Descriptor (CCD), Run Length Descriptor (RLD) and Cooccurrence Matrix Descriptor (CMD).

CCD has been proven to be an efficient and effective feature in content-based image retrieval. Compared to color histogram, correlogram contains not only pixel-based color distribution but also spatial color distribution information. A correlogram $C^k(i, j)$ is a matrix whose element (i, j) is the probability of finding a color with index value j , at a distance k , from a color with index value i . Specifically, in our system, a subset of the correlogram $C^k(i, j)$ which is the autocorrelogram containing only the diagonal elements of $C^k(i, j)$ is utilized as a feature descriptor. It gives the probability of finding the same color index at a particular distance.

RLD is a compact image texture descriptor. A run-length RL value is the number of consecutive pixels $A_i, A_{i+1}, \dots, A_j$ along the scan line $\overline{A_0 A_n}$. $\overline{A_i A_j}$ is a subset of $\overline{A_0 A_n}$ where $A_i, A_{i+1}, \dots, A_j$ have an identical color. The color of A_{i-1} is different from the color A_i and the color of A_j is also different from the color of A_{j+1} . Furthermore, we could define the run-length feature as $\alpha - RL$, if the inclination of $\overline{A_0 A_n}$, which is also the

angle drawn from the X-axis to $\overline{A_0A_n}$, is in the counterclockwise direction. To reduce the computation in feature extraction in our system, two types of quantization were applied on the image feature space. Firstly we quantized the HSV colorspace into 64 bins and then extracted the RLD. We also tried to extract the RLD feature from the intensity map of a color image to describe its texture feature.

CMD is used to capture the texture feature of an image. CMD is extracted based on the intensity map of an image. To extract the co-occurrence matrix feature, we calculated the co-occurrence matrix using three different distances: $[0,1]$, $[1,1]$ and $[1,0]$, and then the energy, entropy and contrast of each co-occurrence matrix were concatenated as a feature vector.

Apart from the global feature we described above, we also incorporated the region based local feature into the proposed system. To introduce region based feature into our system, the first step was selecting an interest region detection algorithm. After implementing several different approaches, Harris-Affine and Hessian Affine [53] methods were selected. The Harris-Affine detector locates interest points by detecting corners on a second-moment matrix at multiple scales by using the Harris corner measure. Hessian-Affine uses a multiple scale iterative algorithm to spatially localize and select affine invariant points. The interest regions detected by these algorithm are robust to image geometric transformation: scaling, rotation and shearing. In our implementation, each detected affine invariant salient region was described as a 5-dimensional vector. That is, $(u_i, v_i, a_i, b_i, c_i)$ in which $a(x-u)(x-u) + 2b(x-u)(y-v) + c(y-v)(y-v) = 1$ with $(0,0)$ at image's top-left-hand corner. After that we further applied affine transformation on detected regions and then worked out the normalized region. Lastly, we described this normalized region by using the SIFT point descriptor with default settings and output a 128-dimensional vector. To apply point/region based features on abstract image indexing and matching, statistic work is required. We borrowed the concept from the

“bag of feature” theory. For each visual concept, we first detected interest areas from several sample images. We combined all region descriptions for all visual concepts and thus created a matrix with 20 million records. The K-means clustering algorithm [95] was applied on this matrix to generate 500 interest area clusters. Therefore, we generated a codebook with 500 cluster centers. By assigning each detected interest area to a cluster, we converted an interest area list into a histogram. This histogram is utilized as the final feature descriptor for an image or a visual concept.

3.3.5 Web Image Retrieval

This subsection explains how to combine the ontology system with the semantic knowledge that is extracted from Web page text contents and images to retrieve images based on given high-level conceptual terms. As shown in Figure 3.6, the query term is sent to the Semantic Handling Module. In this module, the implemented system tries to match both textual and visual descriptions for the query term with the information extracted from the Web page/image content. Specifically, the narrative descriptions defined in the prototype ontology is matched against the Web page text content, while the low-level feature descriptors extracted from images are matched against the visual concept descriptions defined in the prototype ontology. In this case, we avoid the sensory gap in the matching process. Furthermore, the results generated by matching both textual and visual descriptions are merged and output as a list containing the matched images. Lastly, the matched image is returned to the user. In our experiment, different semantic matching methods were investigated. To match the visual concept defined in the prototype ontology with low-level visual descriptor extracted from images, model-based and discriminative methods were applied in our experiment. Moreover, different methods for merging the ranking results of textual and visual matching are also described in this section. We first introduce how to match the textual information with a query term followed

by a description of how to match the visual information with the query term. Lastly, we explain how to combine the results of the above two processes.

As described in the previous section, each Web page is represented as a TF/IDF vector. The column page vectors of a collection of Web pages are put together to form a matrix of order $L \times |Ds|$, called the page data matrix. The query, which contains one or more words, is also represented by a query vector that has the same length as the page vector. The query vector consists of one or more non-zero elements corresponding to keywords that are found in the dictionary. Therefore, the matching process searches for the page vectors that are geometrically close to the query vector. In the current implementation, we use the cosine distance between the query vector and page vector to measure their similarity,

$$\cos \theta_k = \frac{a_k^T q}{\|a_k\|_2 \|q\|_2} = \frac{\sum_{j=1}^L a_{kj} q_j}{\sqrt{\sum_{j=1}^L a_{kj}^2} \sqrt{\sum_{j=1}^L q_j^2}} \quad (3.3.7)$$

for $k = 1, \dots, |Ds|$, where a_k is a column vector from the page data matrix, q is the query vector and L is the length of the feature vector. Thus, pages whose page vector returns a value from Equation (3.3.7) that exceeds a certain threshold are deemed relevant. Alternatively, the values of $\cos \theta_k$ can be sorted to return the top n results. In our experiments, we named this method, the Original Matrix (OM).

A well-documented problem of the above method is that only pages containing the exact query terms are returned as results. Meanwhile, the context information that is associated with the high-level query term is not utilized during the retrieval process. To solve this problem in semantic knowledge query, we implemented various word semantic similarity measurements (WSSM1, WSSM2) as described in the previous section. For a given dictionary *Dict*, each element M_{ij} in M represents the semantic similarity of words

$Dict_i$ and $Dict_j$. Therefore, we modified Equation 3.3.7 to:

$$\cos \theta_k = \frac{a_k^T M q}{\|a_k\|_2 \|q\|_2} = \frac{\sum_{i=1}^L \sum_{j=1}^L a_{kj} M_{ij} q_j}{\sqrt{\sum_{j=1}^L a_{kj}^2} \sqrt{\sum_{j=1}^L q_j^2}} \quad (3.3.8)$$

Using Equations 3.3.7 and 3.3.8, we calculated the $\cos \theta_k$ for every record inside the page data database. Web pages that satisfy $\cos \theta_k > 0$ are returned as the results. The higher the $\cos \theta_k$ value, the higher the probability that this page contains images corresponding to the query terms. To prevent too many false-alarmed results brought about by the context information, we set only documents containing the *query key terms* defined as the synonymy and direct children of the input query term, to return as query results.

The above methods solved the problem of searching for a specific query term in Web page documents, with certain descriptive flexibility. To further improve the matching performance and also introduce the pre-defined knowledge into the matching performance, we introduced ontology-based textual content matching:

Step 1. For a query term that is normally defined as the individual of *class:Concept* under certain *class:Domain*, we derived its textual description from our Textual Description Ontology which is a statement containing subject, property, and object.

Step 2. After all statements were extracted from the prototype ontology, an enhanced query vector of q_e was formed by using subjects, properties and objects.

Step 3. The hierarchy descriptions of Domain Ontology facilitated the finding of all child-concepts for a given term. In most domains, it is safe to infer that if a is a subclass of A , then all individuals of a are also the individuals of A .

Step 4. Using this feature, all textual descriptions of the child-concepts of a query term were also extracted and formed a set of query terms $q_{ci}, i = (0, 1, 2, \dots, n)$.

Step 5. For each query vector in q_e and q_{ci} , a confidence level mask cm_e and cm_{ci} was created according to the definition of properties in the Textual Description

Ontology. The confidence level mask cm has the same dimensionality with query vector q .

Step 6. The value of each bin in cm was decided by the confidence level of corresponding statements defined in the ontology.

Step 7. The final query vector is defined as $q_e \times cm_e^T$ and $q_{ci} \times cm_{ci}^T$ respectively. Using Equation 3.3.7 we matched the query vector against each Web page text document and calculated a $\cos \theta_k$ value.

As mentioned in the previous paragraph, the query term and the child-concept terms are used as the query key, which must be presented in a document, if it is a matched document. Next, we needed to measure the semantic similarity reflected by the combination of $\cos \theta_k$ values derived from both exact term match and child-concept matches by using match making algorithm 1:

Algorithm 1 match making.

Input: the combination of $\cos \theta_k$ values derived from both exact term match and child-concept matches

Output: match making decision

- 1: **if** A Web page text document has no zero $\cos \theta_k$ values for both exact term and child-concepts **then**
  - 2:   *enhanced match* and the sum value of all matched concepts were used as the final score
  - 3: **else if** A Web page text document only has no zero $\cos \theta_k$ value when matching with an exact term **then**
  - 4:   *exact match*.
  - 5: **else if** Web page text document only has no zero $\cos \theta_k$ value when matching with child-concepts. **then**
  - 6:   *subsume match*
  - 7: **else if** a single query has no zero $\cos \theta_k$ value for a Web page text document **then**
  - 8:   *not match*
  - 9: **end if**
-

After we obtained the results from textual description matching, the next step was

trying to match Web images with visual concepts defined in the prototype Visual Description Ontology. As we mentioned in the last section, in the proposed Visual Description Ontology, visual concepts are described by low-level features. Therefore, we approached the middle-level descriptions by matching the low-level feature descriptors extracted from Web images with the descriptors described visual concepts. In our experiment, two different methods were implemented to achieve the visual description matching including model-based approach and discriminative approach. We first introduce the model-based approach. We defined the probability of a Web image I matching with a visual concept V as joint probability $p(I, V)$. Since visual concept V is described by a set of low-level features, we adopted the kernel density estimation (KDE) [96] to approximate the joint probability $p(I, V)$ based on limited observations O . This method does not require any assumption on the data and it has proved to be an effective statistical model for sparse or high-dimensional data. Moreover, the observations O can be obtained directly from the prototype Visual Description Ontology. The joint probability $p(I, V)$ between the feature image I and visual concept V is calculated as follows:

$$p(I, V) = p(x, V) = \frac{1}{\alpha(x, V)} \sum_{o_i \in O} K_\sigma(x - o_i) \cdot \bar{p}(V|o_i), \quad (3.3.9)$$

where x is low-level feature descriptors extracted from Web Image I , $\alpha(x, V)$ is a normalization factor to ensure $\sum_{x, V} p(x, V) = 1$ and K_σ is the kernel function over the continuous random feature space. $\bar{p}(V|o_i)$ is an un-smoothed conditional probability of the visual concepts as feature vector o_i defined in the Visual Description Ontology. We assumed that V is binary in our scenario and thus $\bar{p}(V|o_i)$ is either 0 or 1. In our experiment, a Gaussian kernel K_σ for D-dimensional features was utilized as the kernel function in Equation 3.3.9 which is defined as:

$$K_\sigma(x - o_i) = \prod_{j=1}^D \exp \frac{-||x^j - o_i^j||}{\sigma^j}, \quad (3.3.10)$$

where the σ is a parameter to embed prior knowledge about the low-level features. Cur-

rently, we calculate the σ from examples by using the standard deviation (STD) value of a specific feature over the feature space, the values of the σ indicate what feature is more or less important than other features. We can also manually define the value of σ . For example, through changing the corresponding values of σ , we can emphasize more on color features and less on the texture features. By using Equation 3.3.9 and 3.3.10, we can approximate the joint probability $p(I, V)$ by taking into account the pre-defined visual concepts, but weighted and smoothed with the Gaussian kernel. It measures the nonlinear kernel distance from the feature s to each observation o_i . Moreover, nearby features in the kernel space contributes more to Equation 3.3.9, which is a basic assumption in our experiment. For each image we now have two double-precision values to describe the probability that it contains the given query term. One is derived from the textual description matching of Web page data and the other is from visual description matching using low-level feature descriptors. The next step is combining these two probabilities to generate a final decision. The first step of the combination process is to normalize the two double-precision values to the same scale as $[0,1]$ and then calculate the final decision value as:

$$Dv = (w) \cos \theta_k + (1 - w)P(y|x) \quad (3.3.11)$$

where w is a weight to decide which has more information. Normally, we can set this weight as 0.5, or according to prior knowledge.

Now, we describe the discriminative approach of visual description matching. By observing the structure of the prototype Visual Description Ontology, we found that a hierarchy classification scheme could be formed based on the definition of the ontology. For example, look at the values of property *property:hasContentType*. There are two *class:ContentType* individuals created, they are, *outdoor* and *indoor*. In this case, we used the low-level feature descriptors for different *class:ContentType* individuals as the training data and built a classifier. After that, once a low-level feature descriptor extracted from a

Web image was passed through this classifier, a corresponding annotation was assigned to the image. In our experiment, we used Support Vector Machine (SVM) as our classifier. In recent reports [97, 98, 99], pyramid match kernel provides a possibility to improve our current middle-level feature extraction. The multi-resolution histogram is not a totally new concept. However, K. Grauman and T. Darrell [97] proved that the intersection of two multi-resolution histograms is positive-definite and thus can be used as a kernel function. They designed a pyramid match kernel and this kernel function has several advantages:

- This kernel function is not a simple distance measure function. Therefore, it captures certain levels of context (semantic) information.
- Since it is a histogram-based approach, this kernel function deals with any data that could be mapped to a histogram.
- Compared with other kernel functions, this kernel has lower computational complexity.

Basically, pyramid matching is finding an approximate correspondence between two sets of n -dimensional vectors. To use the pyramid matching kernel, we first needed to build a series of histograms for a feature vector which can be achieved by placing a sequence of increasingly coarser grids over the feature space. After that, pyramid matching works by calculating the *histogram intersection* of two histograms at different resolutions, which is defined as [97]:

$$I(H_X^l, H_Y^l) = \sum_{i=1}^D \min(H_X^l(i), H_Y^l(i)) \quad (3.3.12)$$

in which H_X^l indicates the histogram of feature vector X at resolution l . The matches found at level l include all the matches found at level $l + 1$, because level $l + 1$ is finer

than level l . Therefore, a weight $\frac{1}{2^{L-l}}$ is assigned to each level l over whole level of L . Finally, a pyramid match kernel is defined as [97]:

$$k^L(X, Y) = I^L + \sum_{l=0}^{L-1} \frac{1}{2^{L-l}} (I^l - I^{l+1}) \quad (3.3.13)$$

In our implementation, we utilized the pyramid match kernel, in which feature vector X and Y are two elements of n -dimensional feature space. That means, X and Y are also n -dimensional feature vectors, more specifically they could be two instances of a feature descriptor. Various types of low-level feature descriptors are involved in our middle-level feature generation stage. Since a pyramid match kernel can deal with an orderless image representation, it allows for precise matching of two collections of features in a high-dimensional appearance space. Using the approach described above, several middle-level descriptions were assigned to each Web image. These middle-level descriptions were joined into a textual page vector. In this case, the new page vector embedded both textual and visual descriptions for the Web image and the $\cos \theta_k$ value in Equation 3.3.7 was calculated using the new query vector, reflecting both textual and visual matching results.

3.3.6 Experimental Results

This section shows the experimental results for image retrieval using keywords that represent high level concepts. We first introduce our test method. For image retrieval based on both content and domain semantics, it is not very easy to compare our experimental results with other systems. Firstly, such kind image retrieval systems are designed for different domains. Medical image indexing or retrieval is the first domain that ontology is utilized for reducing semantic gap. Facilitated by the well defined medical ontologies, several systems [100, 101, 86] are reported to have good performance. However, we cannot use their experimental results as benchmarks to evaluate the general image retrieval due to domain constraint. Secondly, different retrieval systems utilize different domain

semantics in their experiments. As we know, the domain knowledge building acts as a crucial part for media retrieval system with domain semantic integrated. By incorporating richer domain knowledge, normally, system produces better retrieval performance. Because the types and amount of domain knowledge are different, it is hard to compare the performance across different systems. With these limitations, find a common testing data set is a first obstacle we need to overcome. In recent reports [87, 88], the results from Google Image Search have been utilized as a common test set. In [87], 5,418 images with associated web pages were downloaded from Google Image Search and used to evaluate system performance. In [88], 400 images from Google Image Search were used along with normal corel stock photo as test set. There are several benefits by getting test data from Web search engine: 1) the test images are generally available for a long period and for everyone; 2) we are able to get Web pages along with images, which is compatible with our system design; 3) we are able to directly compare our experimental results with traditional textual-based retrieval results. In this section, we evaluate the system performance by looking at the results that are generated by using different approaches.

We experimented with two data sets of Web pages and their associated Web images. Google Image Search indexed over 39 million images and our first test data set is based on Google Image Search. For different categories of animals in the canine family, we found that Google Image Search returns averages of about 60% of the top 200 images correctly. Since a much higher percentage of incorrect results are returned outside this range, any bigger image set becomes meaningless in our experiment. We thus limited our test set to only the top 200 Web images for total of 20 classes. Our second data set is based on Yahoo Image Search, an even bigger Web image search engine that indexed over 1.5 billion images. We found that Yahoo has a much larger database for certain classes of animals. Therefore we selected 28 classes of animals and crawled a total of 13,856 Web pages and their associated images for our experiment. From these Web pages, 15,226

words were selected to form the dictionary. In our experiment, we measured the image retrieval performance in terms of the simple correct m number of images retrieved over top n images, precision and recall (PR) curve and Average Precision (AP), which is defined as the averages (interpolated) precisions at certain recalls. AP is defined as: $AP = \frac{1}{M+1} \sum_{i=0}^M P(r_i)$ where M is the number of AP, $r_i = i/M$ indicates the designated recall sample, $P(r_i) = \max_{r_i < r} P(r)$ is the interpolated precision, and $\{r, P(r)\}$ are the available recall-precision pairs from the retrieval results. By using AP, the PR curve can be characterized in a scalar. A better retrieval performance, with a PR curve staying at the upper-right corner of the PR plane, will have a higher AP, and vice versa. In the current experiment, we set $M = 30$.

Firstly, Table 3.1 shows the experimental results of retrieving Web images based on textual information for a Google test-set. As mentioned above, there are 20 classes of Web images in total and each class has 200 Web images downloaded from Google Image Search. The table shows the AP values for each class using different methods. The basic high-level query keywords are the names of the classes. From the table, it is observed that Google Image Search achieved very high AP values for some classes. Our method which entails only searching for exact query keywords, is labeled as OM in the table. However, OM could not achieve the same performance when compared with Google Image Search results. The reason being, exact keyword matching did not take into account any other context information, and thus, we concluded that the times a given keyword is presented in a document, does not always infer the topic of the documents, or the topic of the image embedded in the Web document. In the table, our methods for searching query keywords with the query expansion using Word Semantic Similarity Matrix one (WSSM1) and Word Semantic Similarity two (WSSM2), are labeled as, WSSM1 and WSSM2 respectively. Compared with the OM method, the query expansion helped us to improve the AP in the retrieval of some cases. Note that, in our previous paper [102], we reported that WSSM2

Table 3.1: Results of Web Image Retrieval (Google) using Textual Information.

name:	Aadwolf	Cape fox	Bush dog	Arctic fox	Ethiopian wolf
Google	0.5864	0.4954	0.4869	0.7070	0.7561
OM	0.4027	0.3845	0.3977	0.5203	0.5079
WSSM1	0.4734	0.4124	0.4209	0.5907	0.5433
WSSM2	0.4853	0.4456	0.4190	0.5897	0.5633
Onto	0.6445	0.6278	0.5594	0.6995	0.7590
name:	Coyote	Gray fox	Grey wolf	Fennec fox	Spotted hyena
Google	0.4853	0.7142	0.7428	0.8081	0.8263
OM	0.3376	0.5479	0.6125	0.6329	0.5987
WSSM1	0.3743	0.5945	0.6875	0.6820	0.7014
WSSM2	0.3967	0.6024	0.6987	0.6797	0.6930
Onto	0.5795	0.7094	0.7593	0.8030	0.8478
name:	Dhole	Red fox	Maned wolf	Black jackal	Bat-eared fox
Google	0.6377	0.7311	0.7813	0.8673	0.7840
OM	0.4327	0.5024	0.5495	0.5648	0.4571
WSSM1	0.4734	0.5732	0.5689	0.6378	0.4908
WSSM2	0.4853	0.5690	0.5981	0.6593	0.5340
Onto	0.6445	0.7533	0.7845	0.8527	0.7992
name:	Dingo	Kit fox	Red wolf	Golden jackal	African wilddog
Google	0.6617	0.6642	0.7567	0.6635	0.7592
OM	0.3765	0.4399	0.5873	0.3494	0.5837
WSSM1	0.4928	0.5874	0.6024	0.4769	0.6356
WSSM2	0.4892	0.5827	0.6329	0.4872	0.6589
Onto	0.6908	0.7439	0.7869	0.6979	0.7893

achieved higher performance compared with the WSSM1 method. In this test, since our test data was downloaded from the results of Google Image Search, most Web pages containing the extended query keywords had corresponding topics. This made the over-expansion problem not so critical in this experiment. Therefore, in some cases, WSSM1 out-performed the WSSM2 method. However, for most cases, the performances of both WSSM1 and WSSM2 were below that of the Google method. In the table, our method includes relevant context information in the query by using ontology labeled as, Onto. From the table, it is observed that the prototype ontology improves the AP of image retrieval in most cases. After combining the domain specific knowledge encapsulated

in the ontology, the retrieval results out-perform Google’s method by a large margin in some classes. Compared with the auto-generated Word Semantic Similarity Matrix, the ontology created by domain experts provides the better and more accurate semantic interpretations for query keywords. That can be proven by comparing the retrieval results listed in Table 3.1.

Next, the experimental results of Web image retrieval using both textual and global visual information for the Google test set are shown in Table 3.2. The table shows a com-

Table 3.2: Results of Web Image Retrieval (Google) using Textual & Visual Information.

name:	Aadwolf	Cape fox	Bush dog	Arctic fox	Ethiopian wolf
model	0.8035	0.8259	0.8337	0.9033	0.8312
classifier	0.8146	0.8541	0.8449	0.9124	0.8392
name:	Coyote	Gray fox	Grey wolf	Fennec fox	Spotted hyena
model	0.7975	0.9053	0.8086	0.8081	0.8957
classifier	0.8069	0.9238	0.8179	0.8149	0.9034
name:	Dhole	Red fox	Maned wolf	Black jackal	Bat-eared fox
model	0.7745	0.8927	0.8649	0.8894	0.8347
classifier	0.7894	0.8873	0.8752	0.8971	0.8549
name:	Dingo	Kit fox	Red wolf	Golden jackal	African wilddog
model	0.7459	0.8761	0.7958	0.8219	0.8157
classifier	0.7862	0.8895	0.8059	0.8258	0.8361

parison of the experimental results for both model-based and discriminative approaches. The model-based approach is labeled as, “model” and discriminative approach as, “classifier” in the table respectively. Overall both approaches help to improve Web image retrieval performance by capturing visual descriptions of Web images. Since the visual descriptions of Web images are directly extracted from the low-level feature descriptors, it is a reliable resource for Web image retrieval. From the table, we can see that both approaches perform similarly. However, for several classes, the discriminative approach out-performs the model-based approach by a small margin. In the experiment, it is observed that the visual descriptions help to improve both recall rate and precision of image

retrieval by capturing semantically meaningful background information. It is known that accurate foreground or object detection in still images is still under investigation. In this case, we tried to incorporate visual descriptions with textual descriptions to provide the description for an image scenario or environment. For example, the cape-fox is a kind of fox only present in the desert area of Africa. If the sand or stone scenery is detected in the image itself, it is strong support that the image contains a cape-fox subject. Similarly, for the arctic-fox, if snow scenery can be detected from the image, it is also strong support for the arctic-fox topic. Global visual feature descriptors were utilized in the experiments to detect background or environment information. Table 3.3 shows the image retrieval results on the Google test set by using both textual, global and local visual features. Table 3.3 shows that the image retrieval performance improved for almost every class on

Table 3.3: Results of Web Image Retrieval (Google) using Textual, Global and Local Visual Information.

name:	Aadwolf	Cape fox	Bush dog	Arctic fox	Ethiopian wolf
model	0.8047	0.8283	0.8349	0.9035	0.8405
classifier	0.8233	0.8539	0.8548	0.9263	0.8517
name:	Coyote	Gray fox	Grey wolf	Fennec fox	Spotted hyena
model	0.7994	0.9093	0.8130	0.8089	0.8952
classifier	0.8131	0.9297	0.8251	0.8174	0.9096
name:	Dhole	Red fox	Maned wolf	Black jackal	Bat-eared fox
model	0.7746	0.8963	0.8661	0.8895	0.8368
classifier	0.7899	0.8973	0.8797	0.8985	0.8594
name:	Dingo	Kit fox	Red wolf	Golden jackal	African wilddog
model	0.7479	0.8793	0.7968	0.8157	0.8187
classifier	0.7943	0.8902	0.8127	0.8386	0.8392

both model based and classifier approach by incorporating the region based local visual feature. The classifier received more benefits from the new visual feature.

In Figure 3.9, 3.10 and 3.11, we compared the Web Image retrieval results using the prototype ontology approach with Google Image Search results for three classes. The images highlighted in red are falsely detected images.

Figure 3.9: Experimental Result of Web Image Retrieval (1).

Figure 3.10: Experimental Result of Web Image Retrieval (2).

Figure 3.11: Experimental Result of Web Image Retrieval (3).

Next, we look at the experimental results based on the Yahoo test set. Compared with the Google Image Search test set, the Yahoo test data contained more Web pages and images. Therefore, we used it to demonstrate the performance of our proposed method under a large image database. The experimental results of Web image retrieval using textual information based on the Yahoo test data are shown in the Table 3.4. From the table, we can see that the proposed OM method had the worst performance. Since we tested this method on a relatively large test data, our requirement of exact keyword matching could not be fulfilled. Compared with the Google test set, the WSSM1 and WSSM2 methods provided greater improvement on retrieval performance for some classes of Yahoo test data. For those classes, a lot of domain irrelevant information was generated. The domain-based query expansion helped to remove the irrelevant information. Comparing WSSM1 with WSSM2, the side-effect of over-expansion is clearly shown. For some classes, WSSM1 did not perform well, as it could not provide any improvement compared with the OM method. Moreover, with the larger test set, the textual description ontology still offered similar or better performance for Web Image retrieval without any user rating or feedback.

Table 3.4: Results of Web Image Retrieval (Yahoo) using Textual Information.

name:	Aardwolf	Leopard	Spotted hyena	Bat-eared fox
Yahoo	0.8341	0.5334	0.6891	0.8622
OM	0.3769	0.3789	0.4570	0.3689
WSSM1	0.6589	0.3790	0.5534	0.5637
WSSM2	0.6938	0.4327	0.5962	0.5962
Onto	0.8045	0.5681	0.7079	0.8495
name:	Dhole	Jaguar	Arctic fox	African wilddog
Yahoo	0.4147	0.3248	0.7174	0.8151
OM	0.2391	0.2594	0.4784	0.5892
WSSM1	0.2468	0.2528	0.4981	0.6104
WSSM2	0.2973	0.2961	0.5490	0.6490
Onto	0.4059	0.3874	0.7041	0.8267
name:	Lion	Grey wolf	Ethiopian wolf	Blackbacked jackal
Yahoo	0.5431	0.6927	0.8570	0.8405
OM	0.3483	0.3590	0.4695	0.5477
WSSM1	0.3850	0.4762	0.5837	0.5549
WSSM2	0.4207	0.5291	0.5954	0.5982
Onto	0.5668	0.6944	0.8476	0.8509

Table 3.5 shows the experimental results of Web image retrieval using both textual description and global visual description information for the Yahoo test set, while Table 3.6 shows the experimental results by using textual, global and local visual descriptions.

Table 3.5: Results of Web Image Retrieval (Yahoo) using Textual & Visual Information.

name:	Aardwolf	Leopard	Spotted hyena	Bat-eared fox
model	0.8257	0.6316	0.7742	0.8324
classifier	0.8749	0.6549	0.8357	0.8738
name:	Dhole	Jaguar	Arctic fox	African wilddog
model	0.6320	0.3580	0.8059	0.8347
classifier	0.6570	0.4026	0.8572	0.8672
name:	Lion	Grey wolf	Ethiopian wolf	Blackbacked jackal
model	0.5348	0.8370	0.8699	0.8704
classifier	0.5670	0.8568	0.8739	0.9043

From the table, firstly we can see that for most classes, combining visual description provides noticeable improvement on retrieval performance. However, for some classes,

Table 3.6: Results of Web Image Retrieval (Yahoo) using Textual, Global and Local Visual Information.

name:	Aardwolf	Leopard	Spotted hyena	Bat-eared fox
model	0.8266	0.6395	0.7774	0.8387
classifier	0.8812	0.6689	0.8497	0.8873
name:	Dhole	Jaguar	Arctic fox	African wilddog
model	0.6437	0.35940	0.8123	0.8429
classifier	0.6683	0.4219	0.8648	0.8782
name:	Lion	Grey wolf	Ethiopian wolf	Blackbacked jackal
model	0.5375	0.8439	0.8810	0.8734
classifier	0.5785	0.8597	0.8819	0.9097

such as “Jaguar” and “Lion”, the visual information could not provide good support for inferring the content of the Web images. Those two classes returned only very limited numbers of relevant images, and most irrelevant images were images of outdoor scenery. Even more irrelevant images were brought into the results set by combining the visual description information. By using the model-based approach, the retrieval performance was lower than using only textual description provided by the prototype ontology. For the discriminative approach, there was only very limited improvement by combining visual description information, compared with the use of textual description information only. Moreover, with the larger test set, the more complicated back scenarios went into the test data, the model-based approach could not perform equally well as the discriminative approach in this case. The current model is still unable to handle the description of complicated visual concepts as well as the discriminative approach.

3.4 Discussion

In this chapter, we first introduced the overall structure of our proposed Web image retrieval framework. After that, we described in detail about our proposed multi-modality ontology. In this section, we first explained how to extract concept semantic from other resources and build an ontology for domain description and then several examples were

shown to further visualize the proposed ontology for better understanding. Next, we introduced a Web image retrieval system that implements the proposed framework and ontology system. In such a system, Web pages and images are downloaded automatically. Moreover, textual-based descriptions for Web multimedia content are extracted from the Web page content and utilized in Web image retrieval. The visual descriptions derived from the low-level image feature descriptors also provide semantically meaningful descriptions for image content. The proposed multi-modality ontology provides a solution to combine both textual and visual descriptions and apply them in Web image retrieval. The experimental results demonstrate that by combining domain knowledge, it is possible to improve retrieval efficiency by improving the average retrieval precision.

Regarding ontology structure and ontology building, there are still some open issues. One issue is domain knowledge validation. As we know, the concept semantics extracted from different resources can be merged into one ontology. However, how to provide a trustworthy ontology, validating the semantics extracted from different resources, and resolving the conflicts between the semantics are still under investigation. More research resources are required to work on the high-level aspects of the Semantic Web which is basically, “prove and trust.” From the point of view of system implementation, there are still some open issues for further improving retrieval efficiency and performance. Accurate object detection will be a great help for semantic aware image retrieval. This type of information can easily be merged into our ontology structure and further be utilized in image retrieval. In that case, we can achieve higher precision in Web image retrieval. The other open issues include how to provide more accurate content descriptions by using better low-level feature descriptions or apply new machine learning technology in this type of Web image retrieval system. Some of these issues will be further discussed in Chapter 4.

Chapter 4

Feature Extraction for Ontology Enhancement

In previous chapters, we introduced ontologies for both still images and audiovisual contents description. An open issue is, how to provide more accurate descriptions for multimedia content. In our scenario, accurate content description includes both content semantics and concept semantics. For content semantics, more accurate content descriptions or low-level feature descriptors can greatly facilitate content to concept mapping. From the aspect of concept semantics, the richer and more detailed high-level descriptions help to narrow the semantic gap in multimedia indexing and retrieval. This chapter presents some approaches to enhancing the descriptive power of the proposed ontology, by designing a new visual feature descriptor and generating high-level perceptual information of multimedia descriptions.

In Chapter 3, we introduced a Web image retrieval system to retrieve Web images based on user queries, that is, high-level search keywords. In such a system, the low-level feature descriptor is an important component that provides descriptions for visual middle-level features. In this case, by designing new low-level feature descriptors, we can provide more accurate descriptions for middle-level features and further improve the performance of Web image retrieval systems. This chapter first introduces a texture fea-

ture, called the texture correlogram. The texture correlogram is designed to capture both texture and spatial distribution information. We applied this discriminative feature to provide the description for high-level image content. Similar to designing new low-level feature descriptors, the additional high-level content description also helps to improve image retrieval performance. We introduce a high-level feature providing the description for perceptual information of image contents. Attentive region extraction is a challenging issue for semantic interpretation of image and video content. Successful attentive region extraction greatly facilitates content-based image classification and retrieval. Different from traditional visual attention detection models, we proposed a new attentive region extraction method based on blur information which is contained inside professional photographs that are taken by cameras with multiple focus points and contain EXIFs. Firstly, we combined EXIFs with visual features to quickly select professional photographs from the image database. After that, an algorithm was implemented based on graph theory to automatically extract the attentive region from these photographs. This algorithm measures the saliency for individual pixels based on the edge distribution of images. The proposed approach is able to overcome the contrast map selection problem of traditional visual attention methods, and quickly extract the attentive region using blur information. The attentive region generated by our algorithm has a similar shape and size with the subject of photographs, which is useful information for searching and retrieving high-level semantically meaningful objects. Thus, combined with EXIFs, we can estimate the geometry size of the subject. The geometry information extracted from the image can easily be associated with concepts defined in the ontology, and further utilized in image retrieval.

This chapter is organized as follows: Section 4.1 introduces a new visual description, the texture correlogram, Section 4.2 describes high-level perceptual information extraction for image description, Section 4.3 presents the experimental results, which quantify

the performance of the proposed approach, finally, a discussion is provided in Section 4.4.

4.1 Visual Description: The Texture Correlogram

It is a challenge to select appropriate low-level features for high level image classification. The extracted features should be discriminating enough for differentiating semantically meaningful image categories. A number of image features based on color and texture attributes have been investigated for image retrieval and classification tasks. Among them, color correlogram has been proven to be an efficient and effective feature [103]. Compared to color histograms, correlograms contain not only pixel-based color distribution, but also spatial color distribution information. Histogram-based texture features have been studied by Malik and Belongie in [104], the authors used a histogram of texton for image retrieval and classification. We extended their idea by extracting a correlogram from texton as the feature.

The first step in building texture correlograms is texture feature extraction. We extracted Gabor texture features [44] from a $w \times w$ pixel sliding window and shifted this window along one direction by s pixels at a time until the whole image was covered by the window. $((M - w)/s)^2$ set of Gabor texture features is extracted for a $M \times M$ image. The next step is building up the texton from the extracted Gabor texture features. Since Gabor texture features are high-dimensional vectors, it is very difficult to apply normal quantization methods to them. To solve this problem, we used the K-means clustering algorithm to generate the codebook from Gabor texture features. Subsequently, for each feature vector, we selected the index value of the nearest codebook vector measured by city-block distance to represent its texton. Thus, the cluster number defined in the K-means algorithm, also decides the numbers of different textons.

After we obtained the texton matrix for an image, we calculated the texture correlo-

gram as follows. A correlogram $C^k(i, j)$ is a matrix whose element (i, j) , is the probability of finding a texton with index value j , at a distance k , from a texton with index value i . For an $N \times N$ texton matrix, the correlogram element at (i, j) for distance k is given by,

$$C^k(i, j) = P(I(t_2) = j | I(t_1) = i, d(t_1, t_2) = k) \quad (4.1.1)$$

where $I(t_2)$ ($I(t_1)$) is the index value of texton t_2 (t_1). We used the L_∞ norm as the distance measure $d(t_1, t_2)$, that is, $d(t_1, t_2) = \max\{|x_1 - x_2|, |y_1 - y_2|\}$, where texton $t_1 = (x_1, y_1)$ and $t_2 = (x_2, y_2)$. The computation of the correlogram involves the counting of the value pairs (i, j) , which is very similar to the cooccurrence matrix defined in [105] for texture analysis, i.e.,

$$A(i, j) = \#\{I(t_1) = i, I(t_2) = j, d(t_1, t_2) = k\} \quad (4.1.2)$$

and then, the correlogram can be computed by,

$$C^k(i, j) = \frac{A(i, j)}{n_i \cdot 8k} \quad (4.1.3)$$

where n_i is the number of textons with index value i in I , and $8k$ is the number of textons in the neighborhood at distance k from t_1 . A set of correlograms is found for each distance in the set $s \in \{1, \dots, k\}$. The computational complexity of finding a correlogram for an $N \times N$ texton matrix and distance k is $O(N^2k)$. To compute an entire set of correlograms, one for every distance in $s \in \{1, \dots, k\}$, the computational complexity is $O(N^2q^2)$, where q is the cardinality of s . This can be reduced to $O(N^2q)$ using dynamic programming [103]. However, since we do not propose to use every possible distance in s , and moreover, since $q \ll N$, the computational burden is not a major concern.

A subset of the correlogram $C^k(i, j)$ is the autocorrelogram which contains only the diagonal elements of $C^k(i, j)$. It gives the probability of finding the same texton index at a particular distance. Since the computations can be further simplified by considering only autocorrelograms, we chose to use autocorrelograms to describe image contents on

the basis of texture attributes. We could expand the proposed Ontology for Still Image Description by adding this feature into Visual Description Ontology for image content as shown in Figure 4.1. In Section 4.3, we compare in detail, our proposed texture

Figure 4.1: Expand the proposed Visual Description Ontology.

description, with other texture features.

4.2 High-level Perceptual Information Description: Attentive Region

4.2.1 Overview for High-level Perceptual Information Retrieval

With the rapid growth in the number of digital images, there is an increasing demand for effective access and analysis of digital images. The limitation of current image classification and retrieval systems is due mainly to the difficulty in generating high-level semantic descriptions. Even now, human perception and analysis systems are still too complicated to be modeled completely. To extract and analyze the semantic content of images, a lot of effort is necessary from both the image processing and human visual system analysis

aspects. Initial research into the human visual system has shown that people often pay attention to a specific part of an image rather than the whole image. From human eyes, certain parts of the image are more interesting than the rest. Surveys and experimental results from biological vision and psychology, indicate that there is actually a mechanism known as a saliency map, working behind the human vision system to highlight attention foveas. This finding leads to many possible practical applications. The most direct usage is to guide media adaptation or navigation with the salient map [106], which is the goal for part of the MPEG-21 standard [107]. Other computer vision applications are also able to benefit from visual saliency, for example object recognition [108]. In the past few years, much research has been carried out in visual saliency related areas, to try and mimic the saliency map of the human vision system with computer algorithm. Thus, several computational visual attention models have been created. They provide an effective mechanism for extracting human perception knowledge from images. This perception information is very useful for understanding image semantics, as well as, semantic image classification and retrieval.

The computational visual attention model automatically detects perceptually important regions, and alerts classification and retrieval analysis to this perception knowledge. Existing visual attention models are either low-level feature based, or viewer task dependent [109, 110]. The visual attention model proposed by Itti [109] is one of the popular approaches that has been applied in different applications. It builds up pyramids of three image features (intensity, color and orientation) and uses center-surround difference to calculate contrast. In the next step, different combination strategies are proposed to solve the problem of how to combine the contrasts of different features [111]. For the above models based on low-level features, there is a crucial problem, that is, how to derive the saliency map from several feature contrast maps, because the attention region or subject of an image may not be homogeneous in any feature contrast map. This problem due

to the fact that most current visual attention models identify attention fovea points or attention blocks, rather than generating a region to highlight the visual attention. In this thesis, we try to avoid this problem by looking at it from a different perspective. Unlike previous models, which ignored the potential perception knowledge of image content creators that will actually guide the viewers' attention, we propose instead, extracting semantic perception knowledge of content such as the attentive region of a photograph by utilizing metadata and high-level blur information.

In professional photography, photographers highlight the subject by using several different techniques. When a scene has a very complicated background, or when a particular section of the scene needs to be highlighted, a sharp/blur contrast for subject/background is a method very widely used by most photography enthusiasts. In such types of photographs, the subject or foreground is highlighted using an in-focus area with sharp edges, and the background or all other unimportant objects are represented by an out-of-focus area with blur edges. Since high-contrast strong edges stimulate human visual systems more than low-contrast weak edges, viewers concentrate on the in-focus area when first viewing photographs, this is inline with our understanding of the human visual system. Therefore, using this technique, photographers can guide the viewers' attention easily, this is therefore, an important cue for detecting the visual attention in professional photographs. If we can separate the in-focus and out-of-focus regions (sharp/blur part of the photograph) for a professional photograph, then, it is possible to know the subject of the photograph. Compared with other features, blur information has some remarkable benefits for visual attention detection:

- Unlike other features, blur information is a robust feature for visual attention detection. Firstly, this benefit comes from the characteristic that photographers intend to construct sharp/blur contrasts to highlight the subject. Secondly, the low-level features for a human conceptual subject, may vary greatly due to different appear-

ances in different segments, but the blur information is homogeneous in certain areas which are large enough to represent the subject. Therefore, in some cases, we can use blur information as an isolated feature for visual attention detection.

- Unlike traditional saliency map based approaches, we can easily generate a region from a given photograph, rather than just finding several saliency points. This is due to the blur information, which is homogeneous in the entire in-focus region.
- Blur information can easily be extracted and analyzed using a small area, which minimizes computation. We can therefore analyze one photograph quickly even if the photograph size becomes increasingly larger. Whereas for other features, their sizes increase with an increase in the size of the photograph.

Nowadays, similar to the blur information which provides high-level perception information, the EXIFs stored in digital images provide us with a rich source of information and enable us to estimate the environments where and when the photographs are taken and further help us understand the semantic content inside the photographs. That information is generated directly by cameras and in some cases it reflects the idea how the photographers construct the photographs. Based on the current EXIF standard, much useful information can be extracted directly, such as; aperture size, focal length, subject distance, etc. Furthermore, we can calculate DOF and field of view (FOV) from the information provided by EXIFs. DOF indicates the range of distances from the lens in which objects appear sharply focused, while FOV indicates the angle in object space over which objects are recorded on the film or sensor in a camera. This information helps viewers estimate the geometric size of the subject. Moreover, since EXIFs can be extracted from images without any calculation, they facilitate image searching and indexing in very large image databases. Recently, several approaches that use EXIFs in image retrieval and indexing have been reported. In Caliph & Emir [112] EXIFs have been imported

into graph-like annotations and used for image retrieval in MPEG-7 based Java prototypes for digital photograph and image annotation and retrieval. In [113], EXIFs were utilized in content-based image retrieval. However, how to combine high-level perception information and reliable metadata with traditional visual features is still a challenging problem.

In this chapter, we propose a method to quickly select professional photographs with high-level blur information, and further extract the attentive region from those images by utilizing both metadata (EXIFs) and low-level features. Subsequently, the attentive region information can be utilized in other applications such as content-based image retrieval systems. As a component of a Web image analysis system, some other sub-systems were developed to support it. They include an image crawler system established for efficient Web image downloading and storage, an EXIFapp system developed for extracting EXIFs from Web images; which supports searching and indexing of images using EXIFs. In this attentive region generation module, we first used EXIFs to select a set of professional images from a large image database which had high probabilities to contain high-level blur information, then an automatic focal points detection program was used to further verify whether the images contained blur information, through checking the sharpness of edges near each potential focal point. After that, an algorithm was designed to extract the attentive region from the boundary map of in-focus areas for a given photograph, utilizing graph theory. The attentive region generated using our algorithm has a similar shape and size as the subject of the photograph, which provides useful perception information for the semantic interpretation of image content. It is reasonable for a content-based image classification or retrieval algorithm to classify or search for a subject, using size and shape. Integrating such rich information which describes the image from the human perception aspect, the results of image classification and retrieval become more consistent with human vision. This is the ultimate goal for computational

image classification and retrieval.

4.2.2 Attentive Region Selection for Professional Photographs

This section introduces the use of EXIFs to roughly select professional photographs with high-level blur information, and then goes on to describe our focal point detection based approach for verifying whether images contain such blur information.

EXIF is a standard that, “specifies the formats to be used for images, sound and tags in digital still cameras and in other systems handling the image and sound files recorded by digital still cameras” [114]. The EXIF image file specification stipulates the method of recording image data in files, and specifies the following items: structure of image data files, tags used by this standard, and definition and management of format versions. EXIF provides rich information for environments, when and where the photographs have been taken, such as: the date and time, aperture, shutter speed, ISO speed, as well as other camera settings. The metadata within an EXIF file is stored according to their “tags”, and subsequently, specific metadata can also be retrieved according to these tags. One type of useful information that can be derived from EXIFs is depth of field (DOF) information, defined as the range of distances from the lens in which objects appear sharply focused. The DOF can be calculated by,

$$DOF = \frac{\left(\frac{2L^2D(D-L)}{d \cdot f}\right)}{\left(\frac{L^4}{d^2 f^2} - (D-L)^2\right)} \quad (4.2.1)$$

where L is the focal length of the lens, D is the subject distance, d is the maximum acceptable diameter of a circle of confusion, and f is the aperture size of the lens. A small DOF indicates a greater tendency that an image contains blur information, because only small ranges of distances near the focal point(s) appear sharply focused. By using DOF, we can quickly sort images by their probabilities of containing blur information. For example, a photograph is most likely to contain blur information if the DOF ranges

is only several millimeters. On the other hand, if the DOF is from several hundred meters to infinity, the photograph is most likely a wide angle scene without any blur information. Therefore, a simple fixed threshold of DOF can help filter out most images without blur information, such as photographs of scenery with infinite DOF. However, if we want to calculate blur information accurately, which means estimating how blurry the background is, we still need to know the distance between subject and background. For example, though the DOF for a photograph is relatively small, the entire photograph appears sharply focused if the subject and background stay in the same plane. Similarly, a photograph may contain blur information when the DOF of this photograph is relatively large because of the large distance between subject and background. Since that subject to background distance is not provided by EXIF information, to further refine the results of image selection, we designed a method that can locate the potential focal points (areas) in the given photograph as well as verify whether it contains blur information.

Figure 4.2: Different focal points layouts, (a) consumer compact camera focal point layout, (b) Canon 1Ds, (c) Canon 20D, (d) Nikon D2x.

Each type of camera, has a fixed focal points layout. This layout can be obtained from the viewfinder of the camera. Figure 4.2 shows a comparison between the focal point layouts for a typical consumer compact camera and a few professional Single-Lens Reflex (SLR) cameras. In the figure, (a) is an example of the focal points layout for a compact camera. Normally, for this type of camera, only one to three focal points are provided, and the user is not given the option to select the focal point manually. In this case, users need to focus on the subject first (most likely place the subject at the center of the frame),

recompose the picture after focusing, and then take the shot. This process may cause a focusing error known as the *cosine error*¹ which is due to photograph recomposition after the focus is locked. This leads to the subject being not as sharply focused as desired. Layouts (b), (c) and (d) in the figure, show typical layouts of focal points for professional SLR cameras. Such cameras are provided with many focal points, and the user has the option to select any one of them as the current focal point, which may prevent the occurrence of the cosine error. Therefore, sharply focused subjects are easier to obtain with a professional SLR camera, compared to compact cameras. Consequently, we only targeted photographs taken by SLR cameras in the following sections of this chapter.

It is reasonable to assume that the subject or topic of professional photographs is covered by one or several focal points (areas). If a photograph contains certain blur information, the region around the photographer selected focal points (areas) should appear sharply focused, while the region a distance away from the selected focal points or around unselected focal points should appear out-of-focus (blur). Thus, we can infer whether a large image contains blur information or not by verifying the sharpness of the edges inside the small regions near the focal points (areas). That greatly reduces computations, thus, improving efficiency. One of the problems for implementing this method is how to locate the positions of focal points, because different models of cameras have a different layout of focal points. The metadata describing the model and manufacturer inside the EXIFs helped us solve this problem. It enabled us to find the layout of focal points corresponding to a certain type of camera. After the positions of the focal points were located, the next step was to crop small image patches around the focal points from the original photograph, which are called Camera Focus (CF) areas, and then we used them as samples for edge sharpness detection. Since the focal points are normally represented

---

¹This error is caused by the distance between the subject plane and focal plane. The distance between subject plane and camera equals to the distance between focal plane and camera multiplied by cosine θ which is the angle of camera turn when recomposing the picture.

by rectangular areas, shown as the rectangles in Figure 4.2(b), naturally, we cropped the areas covered by the rectangles, and then used them as CF areas. Normally, the size of one CF area is less than 0.2% of the total size of a photograph.

In our current implementation, the CF areas are classified as:

- **In-focus area:** This type of CF area appears sharply in focus. If one CF area has been detected as an in-focus area, then we assume the focal point (area) corresponding to this CF area is possibly a photographer-selected focal point.
- **Homogeneous area:** The pixels in this type of CF area share a similar appearance, which means they are homogeneous in terms of low-level features such as color and texture. For this type of CF area, we can not tell if they are in-focus or out-of-focus because we do not have sufficient evidence to make a judgement.
- **Out-of-focus area:** If one CF area has been detected as an out-of-focus area, then we assume the focal point (area) corresponding to this CF area is likely to be an unselected focal point.

For a given photograph, if some CF areas are detected as in-focus areas and others are detected as out-of-focus areas at the same time, then we conclude this photograph as containing useful blur information. In this case, we do not take into account whether the images contain homogeneous areas. The images in which all focal points are detected as in-focus areas, out-of-focus areas or homogeneous areas are filtered out due to the fact that they do not contain useful blur information, which is needed when using the attentive region generation algorithm. Therefore, we can refine the results of professional photograph selection using this potential focal point detection method. We describe our algorithm in detail as follows:

To classify the CF area into three classes, we estimated the sharpness of edges inside

one CF area. Currently, we define an ideal edge as a step function S , and the out-of-focus blur edge B as

$$B_\sigma = G(\sigma) * S \quad (4.2.2)$$

where $G(\sigma)$ is a Gaussian smooth filter with sigma σ , and $*$ denotes the convolution. To estimate the sharpness of edges, we defined a feature derived from the edge intensity change ratio (ICR), before and after an additional smooth operation is applied on the image using the Gaussian smooth filter $G(\sigma_s)$ with a small sigma which is defined as:

$$ICR = \sum_{i=1}^n \left| \frac{(G(\sigma_s) * B_\sigma(i)) - B_\sigma(i)}{B_\sigma(i)} \right| \quad (4.2.3)$$

where σ_s is a small value such as $0.2 \sim 0.5$, and n denotes the numbers of pixels on this edge. Figure 4.2.2 shows the intensity change ratio curve for the edges with different

Figure 4.3: Intensity change ratio curve for the edges with different blurs.

blurs. In the figure, the x-axis indicates the different blurs of edges which are measured by using σ in Equation 4.2.2, and the y-axis indicates the ICR values when $\sigma_s = 0.2$ in Equation 4.2.3. From the figure, we can see the ICR values are very large if the σ is very small which means the sharper the edge, the larger the ICR value when we apply

the additional Gaussian smooth filter on it. This provides us a simple way to judge whether the edge is a sharp or blur edge. To further improve the detection accuracy, we calculate the average ICR values (AICR) for all edge points inside the CF areas, a threshold $AICR > \tau$ was set to select the CF areas with sharp edges. The numbers of edge points were generated using the Canny edge detector based on intensity maps of the CF areas. The reason for choosing the Canny edge detector is that the Canny algorithm gives the highest performance when parameters are adapted for each image. Finally, the CF areas with sharp edges were classified as in-focus areas.

For the CF areas that were not assigned as in-focus areas, we needed to further judge whether they belonged to homogeneous or out-of-focus areas. To classify these two types of areas, we used color as a feature. We calculate the standard deviation of color (SDc) in HSV colorspace for all pixels inside CF areas. Therefore, if SDc of one CF area was smaller than a threshold ς , we classified it as a homogeneous area, otherwise we classified it as an out-of-focus area.

Graph Theory based on Attentive Region Generation

After selecting the professional photographs with high-level blur information, we now present a new algorithm to detect the attentive region in photographs based on blur information (as a cue), to compute the image saliency. In Subsection 4.2.1, we mentioned how ICR is helpful for sharp edge detection. Since there must be an edge between the in-focus and out-of-focus areas, attentive region generation can be approached by detecting the sharp boundaries of the in-focus area. The same technique described in Subsection 4.2.1 is used to generate the boundary map for an in-focus area. For a given photograph, we can use a Gaussian smooth filter with a small sigma to smooth the intensity map. After that, we superimpose the smoothed image with the original image. Since the intensities of pixels located on a sharp edge will significantly change, we can get

a clear boundary map of the in-focus region after quantizing the values of the subtracted map. Figure 4.4 shows an example of this process. The figure, on the left is the original

Figure 4.4: Boundary map for in-focus area, (a) original image, (b) boundary map for in-focus region.

image, and the one on the right is the boundary map for the in-focus area of the original image. Greater detail on the boundary map indicates the corresponding sharpness of the edge in the original photograph. As described in Subsection 4.2.1, we assume the sharply focus area is of interest for the human visual system. Therefore, we formulated the solution for attentive region generation as a sharply focused area generation problem, and we tried to solve this problem by edge density estimation on the boundary map of an in-focus area, which is also defined as edge saliency detection. Edges are simple and extensively studied in image processing. They have been used in almost all important computer vision applications (object recognition, tracking, etc). Obviously, an edge itself is salient in a local neighborhood, and the global distribution of the edges captures the attentive region information of the whole image. We propose using ‘Edge Density’ as the measure of the saliency for individual pixels.

We have proven that the perceptual saliency of images is closely related to the edge property underlying blur information. We assume that the attentive region in an image has sharp and well structured edges, while the edges in the unsalient region are always blur and unstructured. The remainder of this section, introduces our new approach

Figure 4.5: Illustration of ‘Edge flow’ to pixel P_i .

for measuring image saliency and generating the attentive region using ‘Edge Density’. ‘Edge Density’ is defined for every pixel P_i , denoted as ED_i . ED_i represents the relative “edgeness” of pixel P_i in the whole image. It is computed by the propagation of the edgeness value of all pixels in the image. To give a mathematical representation of ED_i , a time marching parameter t is introduced and the ‘Edge Density’ for pixel P_i at time t as ED_i^t is denoted. Also introduced is the ‘Edge Density Flow’ which defines the propagation of ‘Edge Density’ from pixel P_j to P_i at time t , denoted by $EF_{i,j}^t$. Then, the propagation process of the edge density in the image is given by the following two equations:

$$EF_{i,j}^t = w_{i,j} \cdot ED_j^{t-1} \quad (4.2.4)$$

$$ED_i^t = \sum_{j=1}^n EF_{i,j}^t \quad (4.2.5)$$

Equation 4.2.4 shows the ‘Edge Density Flow’ from pixel P_j to pixel P_i , at time t . It is the product of the edge density of the originating pixel, P_j , and a decaying factor associated with these two pixels. The choice of the decaying factor will be described later. Equation 4.2.5 shows the ‘Edge Density’ of pixel P_i , at time t . It is the summation of all ‘Edge Density Flow’ to P_i . This is illustrated in Figure 4.5. Pixel P_i receives ‘Edge Density Flows’ from all neighboring pixels and itself.

Equation 4.2.4 is then substituted into Equation 4.2.5. After collecting all instances of ED_j^t as a vector $\mathbf{ED}^t = [ED_j^t, j = 1, \dots, n]'$ and representing $w_{i,j}$ as matrix $\mathbf{w} =$

$[w(i, j), i, j = 1, \dots, n]$, we can combine equations 4.2.4, 4.2.5 and then rewrite them as

$$\mathbf{ED}^t = \mathbf{w} \cdot \mathbf{ED}^{t-1} \quad (4.2.6)$$

The final ‘Edge Density’ value for the image is defined when t goes to infinity.

$$\mathbf{ED}^\infty = \lim_{t \rightarrow \infty} \mathbf{ED}^t = \lim_{t \rightarrow \infty} \mathbf{w}^t \cdot \mathbf{ED}^0 \quad (4.2.7)$$

Where $\mathbf{ED}^0$ is the initial edge density vector for the image, the normalized ‘Edge Density’ vector $\overline{\mathbf{ED}}^\infty$ represents the final ‘Edge Density’ over the image, albeit ‘Saliency’.

The problem can also be formulated in graph theory. Where ‘Edge Density’ of pixels in the image are the nodes in the graph, the decay factor values, $w(i, j)$, between a pair of nodes is the edge connecting these two nodes in the graph. ED_i^t is equal to the accumulation of all possible ‘Edge Density Flows’ to pixel P_i via a path of length t . If we attach a self circle to each node and assign the decay factor of the self circle, $w(i, i)$, to 1 (there is no decay on the self circle), then ED_i^∞ is actually the collection of all instances of ‘Edge Density Flow’ coming to pixel P_i , originating from all possible pixels via all possible paths which do not contain self circles.

The initial edge density vector, $\mathbf{ED}^0$ for the image is set according to the following procedure. If the pixel belongs to edges, the corresponding value in $\mathbf{ED}^0$ is set to 1. Otherwise, it is set to 0. The edge pixels are viewed as the sources of the ‘Edge Density Flow’. Now we describe the choice of the decay factor $\mathbf{w}$. Clearly, the decay factor depends on (1) distance and (2) direction. Since our main focus here is on the ‘edge density’ for attentive region detection, we ignore the direction factor. Only the distance factor is taken into account. In our implementation, the decay factor is defined as

$$w_{i,j} = \exp(-\text{dist}(P_i, P_j)/\sigma^2) \quad (4.2.8)$$

where σ determines the size of the neighborhood. One direct implementation of finding the solution of $\overline{\mathbf{ED}}^\infty$ is to keep increasing the marching parameter t until that $\overline{\mathbf{ED}}^t$ does

not change or the change is below a threshold. This implementation is simple and obvious but not efficient. It also suffers from a numerical problem when the value gets very close to zero. In fact, we can prove that a normalized $\mathbf{ED}^t(\overline{\mathbf{ED}}^t)$ turns out to be converged to the eigen vector($\mathbf{v}_1$) with the largest eigen value of matrix $\mathbf{w}$ as shown in equation 4.2.9. Furthermore, we can prove that $\overline{\mathbf{ED}}^t$, will converge to $\mathbf{v}_1$ independent of the initial vector $\mathbf{ED}^0$ as long as it is nonzero.

$$\overline{\mathbf{ED}}^\infty = \mathbf{v}_1 \quad (4.2.9)$$

Applying Equation 4.2.9, we can derive the new attentive region extraction algorithm. We detect the attentive region of a photograph by detecting the attentive areas one at a time. After detecting the first attentive area, we remove the edge sources lying in the already detected attentive area(s) and recalculate the decaying factor matrix w ; then we apply Equation 4.2.9 to find the next attentive area. We keep repeating this procedure until the desired number of attentive areas is reached. The algorithm is summarized as follows:

- Step 1.** Extract the boundary map of the in-focus area, e , from a photograph based on high-level blur information.
- Step 2.** Construct the decaying factor matrix w from the edge map e using Equation 4.2.8.
- Step 3.** Find the ‘edge density’ vector($\overline{\mathbf{ED}}^\infty$)value of the image by solving the eigen vector problem using Equation 4.2.9.
- Step 4.** Threshold the ‘edge density’ vector($\overline{\mathbf{ED}}^\infty$) using any histogram based threshold method to get the attentive area.
- Step 5.** Update the edge map by removing edges lying in the detected attentive area(s). Then go to Step 2 until the desired number of areas is reached.

Step 6. Combine all detected attentive areas together to get the final attentive region.

Having addressed the attentive region detection algorithm, we proceed to show image processing examples. Figure 4.6 shows an example of the result of the proposed approach. How many attentive area(s) should be generated is a problem when applying our method in the attentive region extracted. Currently, we use the region growing ratio (RGR) to select the number of attentive areas which is defined as: $RGR = (Rs_{(i+1)} - Rs_i) / Rs_i$ where Rs_i indicates the region size of all i attentive areas. The program stops if RGR is smaller than a set threshold.

Since the attentive region generated from our algorithm has a similar shape and geometric size to the subject on the photograph, combined with EXIFs, we can calculate the real subject size according to the size of the attentive region. The FOV of a lens is defined as, the angle in object space over which objects are recorded on the film or sensor in a camera, which simply means the amount of coverage the lens is capable of. It is dependant on two factors, the focal length of the lens, and the physical size of the film or sensor. We can retrieve that information from EXIFs directly. Therefore, FOV can be calculated as follows:

Step 1. The magnification M of the lens is calculated, given its focal length and the distance to the subject: $M = \frac{f}{s-f}$ where f is the focal length of the lens, and s is the subject distance. Both values must be in the same unit, such as millimeters.

Step 2. the horizontal and vertical field of view can be calculated as $V_h = \frac{F_w}{M}$ and $V_v = \frac{F_h}{M}$ where F_w is the width of the film frame, F_h is the height of the film frame and M is the subject magnification.

The subject size is very important perception knowledge for image classification and retrieval. Two subjects with different sizes may have a similar size on the film due to the

Figure 4.6: Example of attentive region detection: (a) original image, (b) edge map for original image and (c) the process of attentive region detection.

difference of M which may lead to confusion during retrieval if they share similar low-level visual features. With appropriate high-level perception knowledge, the efficiency of content-based image retrieval can be improved, for example, we should search for a subject with height from 1 to 2.5 meters, and width from 0.5 to 1 meters, if we want to target human beings.

According to the extracted attentive region, several image descriptors can be generated and integrated in the proposed Ontology for Still Image Description, and therefore be utilized in image retrieval. The first descriptor is MPEG-7 contour shape descriptor, which describes the shape of attentive region based on the Curvature Scale-Space (CSS) representation of the contour. Such contour shape descriptor can be used to browse or retrieve the professional photographs that contain same subject and are taken from similar point of view. Moreover, we can also integrate subject size information in to into Visual Description Ontology for image content as shown in Figure 4.7.

Figure 4.7: Expand the proposed Visual Description Ontology to include attentive region information.

4.3 Experimental Results

First, we compare the performance of texture correlogram with the co-occurrence matrix and the Gabor texture feature. As stated earlier, the autocorrelogram was used as the feature index. The distance set to compute the correlogram was $\{1, 3, 5, 7\}$. To reduce the computational complexity, the size of the Gabor filter dictionary was set to 12. The number of different textons were set to 64. For extracting original Gabor texture features, we used the same parameters as set in [44]. For extracting the co-occurrence matrix feature, we calculated the co-occurrence matrix using three different distances: $[0,1]$, $[1,1]$ and $[1,0]$, and then the energy, entropy and contrast of each co-occurrence matrix were concatenated as a feature vector. Figure 4.8 shows the experimental results when utilizing the texture correlogram and Gabor filter were used as features for high-level image classification. Compared to the vector size of the Gabor texture feature, which

Figure 4.8: Performance comparison of texture correlogram, co-occurrence matrix and Gabor texture feature.

is 30×2 component feature vectors, the proposed texture correlogram consumes more

storage space because we defined 64 different textons and thus, the final feature vector has 256 components. However, from Figure 4.8, we can conclude that the proposed texture correlogram outperforms the original Gabor texture feature when we applied these two types of features for image classification. Moreover, the above comparisons prove that spatial distribution information is a very useful cue for high-level texture image classification.

In this experiment, we also evaluated the effect of different choices of sliding-window sizes on the performance of the texture correlogram. We tried three different window sizes, 8×8 , 16×16 and 32×32 . Experimental results show that the smaller the window size the better the performance of the texture correlogram. There are two explanations: 1) since the small image-block contains a simpler texture pattern than the large one, it is easy to quantize those small blocks into textons with limited numbers; 2) the smaller window size also gives a higher resolution to the texture. This is helpful for accurately estimating the spatial distribution for high-level image contents.

Next, we present the experimental results of high-level perceptual information extraction. We first evaluated the accuracy of metadata captured by cameras, then we showed some results for the focal point detection based photograph selection algorithm. After that, we compared our attentive region detection on the basis of high-level blur information, with the traditional visual attention model [109, 111].

As mentioned previously, we first evaluated the accuracy of metadata captured by professional cameras. We were interested in finding out the accuracy of the subject distance field in EXIFs as it is a very important field for calculating DOF and FOV. Other fields like aperture and focal length taken directly from the camera are likely to be accurate. In the experiment, we selected 1814 photographs that contained subject distance field as test samples. Inside these samples, 905 (49.9% of total samples) photographs were classified as containing accurate subject distance information, 733 (40.4% of total

samples) photographs were classified as containing inaccurate subject distance information and 176 (9.7% of total samples) photographs could not be identified as accurate nor inaccurate [115]. The photographs were classified as containing inaccurate subject distance information mainly due to two reasons: 1) subject distance information in EXIF is inaccurate because it is difficult for the camera to measure this information accurately due to the composition of the photograph, such as, a small subject surrounded by a high-contrast background; 2) the camera does not capture subject distance information but fills in a random number for this field. At the same time, we could not identify whether the subject distance information was accurate or inaccurate because the human visual system has a limited ability to identify the sharpness of subjects in the photographs. Since the current EXIFs cannot provide sufficiently accurate subject distance information, our focal point detection algorithm is helpful in refining the results of professional photograph selection.

Here, we show some results of our focal point detection algorithm. In this experiment, we used the focal point layout of the Canon EOS20D as an example to demonstrate how our algorithm detects the potential focal points. Figure 4.9 shows the results of the focal point detection. In the figure, the yellow rectangle region indicates in-focus areas, the red rectangle region indicates out-of-focus areas, and the purple rectangle region indicates homogeneous areas. The photograph located at the upper-left of the figure shows a typical photograph without blur information but a very small DOF. This is because the subject and background are located in the same plane or the distance between subject and background is too small to have a blur background. For this type of image, all rectangles appear in yellow which indicates the photograph appears sharply focused at all focal points. Since these photographs do not contain the blur information which is required for the next process, we filtered them out. Figure 4.9 shows that our algorithm can automatically select sharply focused focal points from the focal point layout

Figure 4.9: Results of focal point detection. The yellow rectangle region indicates in-focus areas, the red rectangle region indicates out-of-focus areas, and the purple rectangle region indicates homogeneous areas.

of photographs with different types of subjects. This demonstrates that our algorithm is robust and can easily adapt to various types of photographs. One problem to be discussed is that different focal point layouts cover different sizes of images located near the center of the photograph. In some cases, the area covered by a focal point layout is just a small part of the whole photograph, which may affect the efficiency of our algorithm. According to the human visual system, the pixels located at the center of photographs are of greater importance to those located near the boundaries. Moreover, the photographer is more likely to put the subject near the center of a photograph than near the boundaries. Based on these facts, our method performs well for most cases, and is more efficient than scanning the whole photograph. On the other hand, this algorithm is designed for photograph selection. If the subject of the photograph is big enough and all potential focal points are detected as sharply focused, we filter these kind photographs out even if they contain blur information at the boundaries, because the attentive region inside this type of photographs will cover a major area of the photograph.

Next, we compare our approach with the traditional visual attention model. Figure 4.10

Figure 4.10: Comparison of our approach with the traditional visual attention model, (a) original image, (b) the saliency map generated using Itti's model, (c) the attentive region generated by our algorithm. (d) the salient region generated from saliency map.

shows a simple comparison of our method with the traditional visual attention model [109, 111]. In the figure, column (a) shows the original photographs, column (b) shows the saliency maps generated by Itti's model in which the brightness of an area indicates its attention value, column (c) shows the attentive region generated by our approach and column (d) shows the salient region generated from the saliency map, which is the area inside the boundary box. Itti's model is able to highlight the correct attention point for photographs on the first row in the figure. However, since his model is designed only for point-based attention extraction which does not provide any region information, the attention value of the head of the antelope is quite different from the attention value of the body. For photographs on the second row in the figure, Itti's saliency map fails to capture the attention of the photograph because the subject is not salient in any contrast map. For these two photographs, our method detects the attentive region effectively by using high-level blur information. Whereas, extracting the region directly from the saliency map or any individual feature contrast map [116] using the image segmentation method is not able to achieve satisfactory results. Figure 4.11 shows more of the experimental results for attentive region detection. In the figure, column (a) shows the original photographs, column (b) shows the boundary map for in-focus areas, column (c) shows the salient region map which is the combination of all salient region(s) detected by the proposed algorithm, and column (d) shows the final attentive region map. From the figure, we can see that the boundary of the in-focus area appears sharp with strong edges while the out-of-focus area appears blur with weak edges. The figure also proves that our attentive region extraction program is successful in extracting the sharply focused area of the photograph when the boundary map contains noise which is coming from the out-of-focus area of the photograph.

Finally, we use two photographs to demonstrate how the attentive region of the photographs provides perception knowledge for the image-content. Since we can get focal

Figure 4.11: More results for attentive region generation, (a) original image, (b) boundary map for in-focus region, (c) salient region map detected using ‘edge density’ information and (d) final attentive region map.

length and subject distance information directly from EXIFs, we can easily calculate the FOV using the equations provided in Section 4.2.2. In Figure 4.12(b), we use a rectangle to represent the attentive region. For the first photograph in the figure, the horizontal and vertical field of view are 2.4 and 1.6 meters respectively. Therefore we can estimate the height and width of the rectangle in the first photograph which are 1.3 and 1.5 meters respectively. Similarly, the height and width of the rectangle in the second photograph are 0.13 and 0.15 meters respectively. These two examples show that we can estimate the real geometric size for the subject of the photograph effectively from the size of the attentive region. We believe, with this perception information, the user can target specific semantically meaningful image-content more easily, especially when two different image-contents share similar low-level visual features.

Figure 4.12: Examples of attentive region size detection. (a) attentive region of the photographs and (b) rectangle representation for attentive region.

4.4 Discussion

In this chapter, we presented two feature descriptors for both low-level visual features and high-level perceptual information. The proposed texture correlogram captures texture feature as well as spatial distribution information. It helps us to improve the high-level concept description by encapsulating more information into single feature descriptors. The spatial distribution information of texture features has been proven as a very important factor for high-level visual concept detection and classification. In this chapter, we also introduced a method to automatically extract high-level perceptual information from digital photographs taken by professional DSLRs. In our experiments, we demonstrated that by combining the metadata provided inside digital photographs and low-level image features, we can achieve more accurate attentive region detection as compared to the conventional saliency map method. Moreover, by further utilizing EXIF metadata and the attentive region generated using the proposed method, we can achieve a good estimation of the geometric size for the subject of the photograph, which greatly facilitates the object description at the concept level. By using this geometric size information, different objects with similar visual appearances can be distinguished directly using this high-level description. Both the new visual description and perceptual information that we designed, can further improve the descriptive power of our proposed ontology.

Chapter 5

Ontology and Content-based Video Analysis

In recent years, with the remarkable increase of video data generated and distributed throughout networks, there is an evident need to develop an intelligent video browsing and indexing system. To build such a system and facilitate content-based video access, automatic semantic extraction is a pre-requisite and a challenge to multimedia analysis systems. Therefore, semantic video analysis and annotation have received much interest and attracted research efforts. This chapter first introduces an ontology system which provides the descriptions for both audiovisual contents and audiovisual content analysis mechanisms. Unlike still images where many images may be classified into the same category, and the semantic meaning of these images appear the same to the users; audiovisual data may have totally different appearances or representations even if they share the same semantic meanings. At the same time, as there is a lot of information enclosed in an audiovisual sequence, the user may need more detailed descriptions for an audiovisual sequence than an image. In this case, it is very difficult to use the same mechanism as still images to provide the semantic interpretations for audiovisual data. Therefore, we need to build an ontology specifically for audiovisual content descriptions. We want to encapsulate two types of information into this ontology; first, hierarchical descriptions for

the content of an audiovisual sequence; second, the mechanisms for analyzing the content of a specific type of video data. According to the characteristics of audiovisual analysis, the high-level audiovisual analysis requires not only low-level feature descriptions such as color, texture, motion and shape, but also specific domain knowledge such as the rules in a soccer or basketball game. Since different types of audiovisual analysis require a different domain knowledge, ontology is a feasible method to manipulate that knowledge in different complexities. Moreover, the semi-automatic audiovisual analysis for a small enough domain can be achieved if enough domain knowledge is accumulated. Therefore, our ontology for video content descriptions can not only be used as a description for a single video sequence but also as a source for finding the domain knowledge of different types of video data. The rest of this chapter describes the development of our system of analyzing and annotating basketball videos using both low-level features and domain knowledge encapsulated in the proposed ontology system. In particular, we show that the multi-modal based approach based on visual, motion and audio information can generate reliable annotations for a basketball video, which can not be successfully achieved by just using a single mode. We addressed the issue of semantic basketball video analysis and annotation for MPEG compressed videos using multi-modal information. The solution has three related aspects: 1) analyze the structure of the basketball video, 2) locate the potential positions where an interesting event may occur, and 3) represent the results in an annotation file utilizing standardized descriptions. Since the semantic understanding of video content is highly dependent on the utilization of contextual information and domain rules, this basketball video analysis and annotation method is proposed based on visual, motion and audio information as well as domain specific knowledge. In this case, the proposed ontology provides information not only of how to process the video to generate content annotation, but also, the domain knowledge of how to interpret a high-level video event to generate the concept level annotation.

This chapter is organized as follows: Section 5.1 introduces the ontology for audiovisual content description, Section 5.2 presents a multi-modal basketball video analysis system based on the proposed ontology, finally, a discussion is provided in Section 5.3.

5.1 Ontology for Audiovisual Description

5.1.1 Overview of the Ontology

To build a system that facilitates content-based audiovisual access, automatic semantic extraction is a pre-requisite and also a challenge to such multimedia analysis systems. Therefore, semantic audiovisual analysis and annotation have received much interest and attracted research efforts. One key problem is how to manipulate the semantics extracted from audiovisual data and enable those semantics to be queried and retrieved easily. At the same time, unlike the normal approach of audiovisual analysis such as key-frame extraction and short detection, semantic audiovisual analysis requires specific domain knowledge to detect high-level information such as the “goal” in a soccer game. In this case, we also need a mechanism to make that knowledge reusable and accessible. Driven by these requirements for semantic audiovisual analysis, we propose an ontology to provide the description for the audiovisual content and the domain knowledge relevant to this content. The ontology for our current system is built using the OWL ontology language.

Figure 5.1 shows the overall structure of the proposed ontology system. There are three sub-ontologies in the figure. Similar to the ontology for image content descriptions, different layers of the ontology system are connected using import statements provided by OWL.

The first sub-ontology is the Domain Ontology, it contains a hierarchical tree structure description for a specific domain and its semantics are based on the classification or

Figure 5.1: Overall Audiovisual Ontology Structure.

taxonomy of audiovisual content. The objective for creating this ontology was to divide a wide domain into several narrow domains, so that the semantics in narrow domains can have consistent interpretations. Moreover, by utilizing this ontology, the relationships between the audiovisual concept semantics could be specified. For example, the top-most layer for this ontology can be a general concept of audiovisual content, having several subclasses such as “news,” “sports,” “movie,” “drama,” “surveillance,” and so on. Furthermore, “basketball,” “soccer,” and “tennis” can be subclasses of “sports (video).”

The second sub-ontology is the Content Analysis Ontology. This sub-ontology imports Domain Ontology to get basic domain classification information. Moreover, similar to Visual Description Ontology for image content descriptions, we imported the MPEG-7 Ontology to provide the basic definitions of audiovisual content descriptions. In the ontology for image content descriptions, we mainly imported the MPEG-7 visual descriptors that were low-level feature descriptors, while in the ontology for video content descriptions, we mainly imported the elements of MPEG-7 Multimedia Description Schemes (MPEG-7 MDS). The MPEG-7 MDS provides definitions for fundamental elements, such as, audiovisual segments in segment descriptions, and even for content semantics, which are vital elements needed to describe the structure and semantics for video sequences. By import-

ing these formally defined MPEG-7 basic elements, the proposed ontology can provide semantic interpretations for video content by incorporating prior domain knowledge, while remaining compliant with the MPEG-7 standard. Based on the MPEG-7 Ontology, the Content Analysis Ontology provides information about how to decompose or analyze the audiovisual entities under a certain domain. For example, an audiovisual entity can be decomposed into an audiovisual segment of “shot.” Different audiovisual types defined in the Domain ontology, may have different shot detection methods. In this case, different shot detection methods can be associated with the audiovisual segment “shot” by using a *property:generateby* property. By utilizing this Content Analysis Ontology, the video analysis system is able to decompose an audiovisual entity and extract the feature description from audiovisual segments.

The third sub-ontology is the Concept Analysis Ontology. The goal of this ontology is to capture the information about how a high-level audiovisual concept is detected by using both audiovisual content description, and audiovisual domain knowledge. The Concept Analysis Ontology also imports the Domain Ontology to provide domain specific concept descriptions. In the Concept Analysis Ontology, a general audiovisual concept is defined first. Next, audiovisual events are defined as subclasses of audiovisual concepts. After that, several detailed audiovisual events are created, for example, “basketball goal” is one of the events under the basketball domain. The ontology also captures the knowledge about how to detect this event from a basketball video. For example, using human natural language, we can describe the event of a basketball goal as: “the basketball hits the rim, and goes inside the basket, all audiences applaud loudly and then the team that is on defence starts playing ball again.” From this description, we can extract the machine processable interpretation for the “basketball goal” event as: basketball goal event happens in a certain video scene, from which we can detect the audio key sound of ball hits rim, and audiences applaud, and the camera motion will change direction be-

fore and after this video event. Another example, many high-level audiovisual processing systems require a shot/scene classification. In an audiovisual shot/scene classification, an MPEG-7 classification-schemed individual can be associated to a shot, by using *property:hasClassificationscheme* to provide the information of how to label an audiovisual shot/scene. In this case, the semantic interpretations for video concepts are captured by the Concept Analysis Ontology, it also enables that information to be reused and shared between different applications.

5.1.2 Define and Build the Ontology

This subsection provides detailed information about the ontology building for audiovisual content descriptions. This information includes the formal definitions for the classes and properties utilized in the ontology system and the main restrictions applied on the classes or properties.

Domain Ontology (audiovisual)

As mentioned in Section 5.1.1, this ontology provides audiovisual content classification information. The content classification information is abstracted from human high-level descriptions of the audiovisual content. The basic content classification for audiovisual entities can be: “news,” “sports,” “movie,” “drama,” “surveillance,” “cartoon,” “music video” and so on. For each category, we created a class and defined it as a subclass of *class:AudioVisualContent*. In addition, detailed classification information was defined as the subclass of these basic classes. For example “soccer,” “basketball,” “tennis,” “baseball” and so on can be defined as subclasses of “sports” while “indoor scenery series” and “wild life” can be defined as the subclasses of “drama”. The guideline to create content classification information is that if two types of audiovisual entities share the same audiovisual events with different semantic interpretations or observations, these two types

of audiovisual entities should be classified into different content classes. Figure 5.2 shows an example of an audiovisual domain ontology.

Figure 5.2: Example of an Audiovisual Domain Ontology.

Content Analysis Ontology

The proposed Content Analysis Ontology provides information on how to decompose the audiovisual sequence into small machine processable segments. We first introduce the classes and properties that were imported from the MPEG-7 Ontology, which are shown in Figure 5.3. Based on the MPEG-7 Ontology, to manipulate the semantics extracted from video data and their inter-relationships between them, video content description terms are classified into several concept and relationship groups, which are represented by OWL classes and properties respectively. The general classes and properties in this ontology are shown in which are Figure 5.4. By utilizing the proposed Content Analysis Ontology, we could describe the audiovisual content processing in a highly structured way. Each audiovisual sequence was first classified as a certain type of audiovisual, because different types of video require different processing methods. Next, the audiovisual sequence was decomposed into audio and video parts, and then audio and video segments were generated from the audio and video sequences. According to different types of video segments, we further classified the video segments as video shot and video scene.

Figure 5.3: The general structure for Content Analysis Ontology (1).

Figure 5.4: The general structure for Content Analysis Ontology (2).

The difference between video shots and video scenes is that video shots represent the physical boundary of the video, and video scenes represent the semantic boundary of the video [117, 118]. Based on different types of videos, the relationships between video shots and scenes are also described on this abstract level. Similar to video shots and scenes, we defined the audio key sound interval as a semantically meaningful segment of the audio segment. For each step of the video/audio decomposition, an video/audio processing method was associated to the video/audio segment to indicate how the segment was to be generated.

Concept Analysis Ontology

Last subsection describes how to generate temporal audio/video segments. The final objective for audiovisual processing is to extract high-level human concept items to facilitate audiovisual indexing, querying and retrieval. This subsection introduces a Concept Analysis Ontology to capture the domain knowledge and also audio/video processing technology to extract high-level semantics from audiovisual content. Figure 5.5 shows the general structure for this ontology. By utilizing the proposed Concept Analysis Ontology, audiovisual high-level conceptual items can be given a machine processable definition. The proposed ontology provides the support for very fundamental audiovisual semantics such as shot/scene classification information. For example, for a soccer video we can use “Long shot,” “In-field medium shot,” “Close-up shot,” and “Out of field Shot” to classify the video shots that may contain different semantics. Moreover, by combining the domain knowledge, we can further infer the events inside the video segment. For example, through analyzing the visual and audio features of a soccer video segment, we can detect the “Goal” or “Foul” events. Since the events correspond to the segment of interest in inside a video, the successful detection of these events will greatly facilitate video indexing and retrieving.

Figure 5.5: The general structure for Concept Analysis Ontology.

5.1.3 Example of the Ontology for Audiovisual Content Descriptions

This subsection shows several examples for the implementation of the ontology for audiovisual content descriptions. In this example, we demonstrate how the proposed ontology system provides highly structured hierarchical descriptions for audiovisual contents and concepts. After that, we demonstrate how the information provided by the proposed ontology system helps audiovisual analysis by encapsulating the domain knowledge. Multimedia analysis systems can not only obtain content information for a specific audiovisual entity, but also the analysis scheme which indicates the methods, tools and domain knowledge used for generating the content information. Certain types of audiovisual data in a specific domain may share similar analysis schemes. Users can decide to follow schemes provided by prior knowledge or create a new scheme to suit their requirements. Using the proposed ontology, the high-level domain knowledge can be stored in a formal structure, and be accessed, manipulated and reused easily.

Figure 5.6: Example of the Content Analysis Ontology.

Figure 5.6 shows an example of the Content Analysis Ontology. In the figure, the ellipse represents the class, and the connector with text indicates the property between two individuals. There is an audiovisual clip at the top-most level of the ontology. To provide rough information about the video content, the property, that is, *property:hasAudiovisualDomain*, is defined here. The domain indicates which category the video belongs to, this is imported from the proposed Domain ontology. The first step of the video analysis involves dividing the audiovisual clip into audiovisual segments. After that, the audiovisual clip is divided into corresponding audio and video clips which provide the audio and video descriptions respectively. First, we introduce the video content description, then the audio content description. The video content description begins with temporal video decomposition. In the MPEG-7 standard, one video/audio segment can be decomposed further into detailed video/audio segments, which are also defined as the video/audio segments. That means we cannot represent the different video/audio semantics that are derived from different lengths of video/audio segments. For example, in the scenario of a basketball game, a basketball match video clip can be decomposed into four segments in which each of them represents a quarter of the game. Furthermore, we can decompose a quarter of the game into more detailed semantically meaningful segments. However, if we only have a video/audio segment defined in the MPEG-7 standard, we cannot manipulate the semantic and make it support the inference engine (the self-decomposition relationship is not accepted by the logic model). So in this case, to maintain the semantics in different types of video/audio segments, we introduced “scene” and “shot” into our ontology system, which are defined as the subclasses of the video segments. To provide a flexible description structure, we did not restrict the decomposition relationships between “scene” and “shot.” That means one shot may consist of several scenes and vice versa. Though we allowed a flexible decomposition relationship between the two concepts, their definitions restrict the individuals of the two classes to

having different semantic meanings. A shot is generated based on the physical boundary

Figure 5.7: Example of Concept Analysis Ontology.

of the video segments, and a scene is generated based on the semantic boundary of the video segments. In the example, we decomposed a video into several shots and then each shot was further decomposed into several scenes. The figure only shows the video scenes that are decomposed from the video shots, but the ontology definitions can support both decomposition schemes either from video scene to video shot or from video shot to video scene. For each step of the temporal decomposition, a corresponding decomposition method, either *class:ShotBoundaryDetection* or *class:SceneBoundaryDetection* individual, was associated to *class:VideoShot* or *class:VideoScene* to provide information regarding the detection of shot or scene from the video segment. For example, for a *class:ShotBoundaryDetection* individual, a video processing method can be associated to it by using the *property:hasProcessingMethod* property. An MPEG-7 visual or other

feature descriptor can also be associated to it by using the *property:hasVisualDescriptor* property. Similar to video temporal decomposition, we can apply temporal decomposition to an audio clip. Here we defined the minimum unit of a semantically meaningful audio segment as an audio key sound. Semantically, it has the same meaning as a scene defined in the video domain. The corresponding *class:AudioKeySoundIntervalDetection* individual was associated with each audio key sound too.

After the temporal decomposition of audiovisual sequences, we now show an example for Concept Analysis Ontology. In Figure 5.7, we demonstrate how an event can be detected from an audiovisual sequence. Suppose we provide a concept description for a basketball video, therefore, we defined a *class:BasketballEvent* that is the subclass of general class *class:Event* and domain *class:BasketBall*. Before defining a high-level event semantic, we must first define some concepts that can be derived from audiovisual segments such as scene classification information. For describing the scene classification information, we imported the *class:ClassificationScheme* and *class:LabelTerm* from the MPEG-7 Ontology. For a scene, we can use one *class:SceneLabel*, which is a subclass of *class:LabelTerm*, individual, to describe it using *property:hasSceneLabel*. At the same time, each *class:LabelTerm* individual can also be associated with an individual from *class:ClassificationScheme* by using *property:belongsToScheme*. Using this mechanism, we can figure out the scenario where the classification is applied. Based on the definitions of *class:SceneLabel* and *class:ClassificationScheme*, we can also know how many different classification schemes use this *class:SceneLabel* or find all video scenes that are assigned a specific *class:SceneLabel*. This term is even used by different classification schemes. Facilitated by the descriptive power of the Web Ontology Language (OWL), the user can search for a specific individual or a class of individuals easily. Similarly, specific labels are associated to video shot and audio key sound intervals. Now, we can use the basic elements defined above to provide the concept description for high-level semanti-

cally meaningful events. As we know, human priori knowledge is an important cue for detecting high-level semantics in multimedia data. For example, the rules in a basketball game are very useful cues for detecting interesting events such as goals or fouls in basketball videos. To manipulate that human priori knowledge we needed to associate the semantics with their corresponding human priori knowledge. In other words, we tried to provide the interpretation for high-level semantics in audiovisual data by combining priori knowledge. In the figure, the basketball event “foul” is a high-level semantic in basketball videos. We can describe this event as: 1) firstly, this event should occur in a court-view scene, which is an individual of *class:SceneLabel* for video scene classification; 2) secondly, the event “foul” will occur along with the occurrence of audio key sound “whistling”; 3) lastly, only the audio key sound “whistling” is not detected at the beginning of this scene, which indicates that the “whistling” associated with event “foul” does not occur at the beginning or the end of the match. To combine the above conditions, we can first define an audio key sound pattern. This pattern was defined by using *property:hasCurrentAudioKeySoundLabel* that associates an audio key sound label to a pattern and also *property:hasLastAudioKeySoundLabel* and *property:hasNextAudioKeySoundLabel* to provide the context information. After that the event can be defined as, it can be detected in a scene which is described as a union of a specific scene label and audio key sound pattern. In this case, we provided a description of how to detect the potential event “foul” in a basketball video and we can encapsulate the above conditions into the ontology using the properties and classes shown in the figure.

5.2 Multi-modal Audiovisual Content Analysis

Generally, the processing of sports videos include the following areas: analysis of the structure of the video, detection of important events or activities, following a specific player’s actions, and generating the summary. The video analysis aims to automatically

extract semantic information from a video. With such semantics represented in terms of high level descriptors, indexing, searching and retrieving the video content can be improved. Previous research work [119, 120, 121] attempted to extract the semantics from visual and motion information. However, investigation into extraction of semantic information from multi-modality data is still very limited. From the point of view of video processing using visual and motion information, several sports video analysis and modelling methods were investigated. In [122], low-level soccer video processing algorithms and high-level event and object detection algorithms were utilized for automatic, real-time soccer video analysis and summarization. In [123], color and motion features were used for dominant scene clustering and event detection. However, the above methods do not take into full consideration motion information, which is an important cue for sports video analysis. In [124], the authors utilized motion information for describing individual video objects, but object segmentation for complex scenes such as sports videos, is still a challenging problem. Moreover, currently there is no system that can import domain knowledge or pre-defined semantics and utilize them in video analysis. Therefore, we propose a video analysis and annotation system to import video content and concept analysis descriptions directly from an ontology, and follow the pre-defined scheme to generate video annotations. To provide better performance compared to conventional systems in terms of processing, we also propose an approach to differentiate camera and object motion from total motion, without object segmentation. In the proposed method, we introduce a modified scene detection algorithm, based on both visual and motion prediction information. New motion features are proposed to capture the total motion, camera motion, and object motion, respectively. Camera motion is estimated from the motion vectors in the compressed video using an iterative algorithm with robust outlier rejection. The reasons for using motion features are two folds: 1) motion information has a strong relationship with semantic events, i.e. different events exhibit

different motion patterns, 2) different events can be identified by motion features within a game and the video model generated by analyzing motion features is flexible enough to be applied to other classes of sports videos. According to the semantic interpretation provided in the proposed ontology, we utilized both visual and audio information to detect high-level sport audiovisual events. For our prior work on audio analysis, please refer to publication [125].

5.2.1 Multi-modal Basketball Video Analysis and Annotation

According to the definition of basketball video content analysis, the proposed multi-modal video analysis consists of two components: 1) video segmentation and classification using visual and motion features, 2) audio key sound extraction based on Hidden Markov Model (HMM). This subsection mainly describes video processing following the descriptions extracted from the proposed ontology. The video/audio temporal decomposition provides the fundamental elements for semantic video analysis. After that, the specific classifier is used to generate labels for each video/audio segment based on the classification scheme defined in the ontology. Please refer to our prior publication [125] to obtain detailed information about audio processing.

Shot and Scene Boundary Detections

The shot and scene boundary detection is the initial step in our video analysis algorithm. Shot is the physical boundary of a video, while scene is the semantic boundary [117, 118]. Although there is rich literature on algorithms for detecting video shot and scene boundaries, it is still a challenging problem for basketball videos. As mentioned above, a scene can be viewed as a semantic unit. Unlike other types of videos, e.g. movies, in which a scene is a group of shots which constitute the semantic unit, the scene in a basketball video may be a segment of a shot. In basketball videos, a single video shot can be a

court-view camera that tracks the players or basketball for a significant amount of time, without cuts or transitions, but with plenty of panning and some zooming. Generally, one or many meaningful semantics, like actions or events, e.g. *shot at the basket*, or *foul*, are contained in such types of shots. Since it is hard to extract the detailed information for these actions or events from a single long camera shot, it is necessary to further partition the shot into scenes. Therefore, in the current implementation, we segment a basketball video into shots first, and then further segment a shot into several scenes. After analyzing the structure of the long camera shot, we found that the semantics in the shot have a strong relationship with the global motion associated with the movement of the camera. For example, actions such as *shot* and *foul* are most likely when the camera motion is slow, while fast camera motion often indicates offensive and defensive exchanges. Thus, we proposed a video temporal segmentation algorithm based on color and motion prediction information, to detect shot and scene boundaries simultaneously.

Use of motion prediction information in MPEG video to detect the shot boundary was proposed in [126]. Motion vectors are divided into four types, and the number of each type of macroblocks (MBs) in a frame is used to indicate the similarity/dissimilarity of that frame with its neighboring frames. In our current algorithm, we extended the method in [126] to combine a color-based shot boundary detection method to detect the shot/scene boundaries in basketball videos simultaneously.

Firstly, we used the difference between the color histograms of the neighboring frames D_h , as the feature to detect shot boundaries, which is defined as:

$$D_h = \frac{\sum_{i=0}^N |H_n(i) - H_{n-1}(i)|}{width \cdot height} > T_g \quad (5.2.1)$$

where N is the number of bins in the color histogram, H_n and H_{n-1} are the color histograms of frame n and $n - 1$ respectively, $width \cdot height$ denotes the pixel numbers in each frame, and T_g is the threshold for detecting an isolated sharp peak in a series of discontinuity values of D_h .

Subsequently, to detect the scene boundary accurately, we modified the original algorithm defined in [126]. Firstly, we modified the definition of the Frame Dissimilarity Ratio (FDR) to provide a precise scene boundary detection. The definition of the new FDR is:

$$FDR_n = \begin{cases} \frac{Fw_{n-1}}{Bi_{n-1}} & \text{for I frame} \\ \frac{In_n}{Fw_n} & \text{for P frame} \\ \frac{|Fw_n - Bk_n|}{Bi_n} & \text{for B frame} \end{cases} \quad (5.2.2)$$

where In , Fw , Bk and Bi represent the number of MBs for intracoded, forward predicted, backward predicted and bidirectionally predicted frames, respectively, and n denotes the frame number. We modified the FDR by, 1) creating an expression of FDR for P-frame to provide more accurate feature descriptions when the boundary is located at P-frames, and 2) modifying the expression of FDR for B-frame to eliminate false detection if $Fw_n \approx Bk_n$ when they are all much larger than Bi_n . Consider the following frame structure in an MPEG bit stream : $\dots I_1 B_2 B_3 B_4 P_5 B_6 B_7 B_8 P_9 \dots$. If the shot change takes place at B_3 , FDRs for B_2 , B_3 and B_4 will be very high. In order to determine the exact location of the shot boundary, we observed that B_2 is mostly forward predicted while B_3 and B_4 are mostly backward predicted. Thus, at the shot boundary there is a change in the dominant MB type of the B-frame. So, we defined a parameter called dominant MB change ($DMBC_n$) for frame n as

$$DMBC_n = \begin{cases} 0 & \text{if } (Bk_{n-1} - Fw_{n-1}) > 0 \\ & \& (Bk_{n+1} - Fw_{n+1}) \leq 0 & \text{for I frame} \\ 1 & \text{otherwise} & \text{for I frame} \\ 1 & & \text{for P frame} \\ 0 & \text{if } (Bk_n - Fw_n) * \\ & (Bk_{n-1} - Fw_{n-1}) > 0 & \text{for B frame} \\ 1 & \text{if } (Bk_n - Fw_n) * \\ & (Bk_{n-1} - Fw_{n-1}) \leq 0 & \text{for B frame} \end{cases} \quad (5.2.3)$$

Thus, the DMBC acts a filter to locate the scene boundary precisely.

Scene Classification

Based on the definition of our basketball scene classification scheme, we classified basketball scenes into six classes: 1) fast-motion court-view scenes, 2) slow-motion court-view scenes, 3) penalty scenes, 4) in-court medium scenes, 5) out of court or close-up scenes, and 6) bird's-eye-view scenes. The definitions and characteristics of each class are given below:

- *fast-motion court-view scene*: This scene displays a global view of the court and has obvious global motion; hence, this type of scene can serve to differentiate the offensive and defensive exchanges between teams.
- *slow-motion court-view scene*: A scene that displays the global view of the court and has insignificant global motion; hence, this type of scene can be used to locate interesting events.
- *penalty scene*: A scene that shows the taking of a penalty under the rim.
- *in-court medium scene*: A scene that focuses on a full-body shot or players in a cluster. Generally, it is a zoomed-in court-view scene. In most cases, action replays are shown as in-court medium scenes.
- *out of court or close-up scene*: Such scenes display the audience, coach and close-ups. These types of scenes usually indicate a break in the match, or highlight the player who has just executed an exciting event.
- *bird's-eye-view scene*: A scene that shows a global view of the whole gymnasium and is usually taken from a stationary camera.

Figure 5.8 shows an example of each of the six typical scenes. A series of texture and motion features were extracted for classifying a scene into one of the above six classes. In

Figure 5.8: Example of typical scenes.

our initial experiment, the texture features were extracted from the key frame of a scene, which is an I-frame located at the centre of the scene. Two texture features, *run length feature* [127] and *co-occurrence feature* [128] are generated from the key frame. The run length feature vector has four dimensions, including viz., long run emphasis, shot run emphasis, run length entropy, and run length energy. The co-occurrence feature vector has three dimensions — contrast, energy and entropy.

The second type of features is motion features. In order to differentiate the camera and object motion from the total motion, we have to estimate the global motion. Model based motion estimation has been reported extensively in literature [129]. In [130], the affine parameter estimation problem is formulated as a nonlinear minimization problem

which is solved using an iterative algorithm. The objective function to be minimized is the sum of square difference between the original image and the warped image, by the affine transform parameters. This method is semi-automatic because the user needs to identify at least 3 corresponding feature points in two frames.

Our global motion estimation algorithm is an iterative algorithm with robust outlier rejection. Affine parameters are chosen so as to fit the block based motion vector between two frames, which are available from the MPEG compressed video stream. We modeled the global motion as

$$\begin{aligned} mvx_i &= p_1x_i + p_2y_i + p_3 \\ mvy_i &= p_4x_i + p_5y_i + p_6 \end{aligned} \quad (5.2.4)$$

where mvx_i and mvy_i are the components of the motion vector for a particular macroblock (MB), x_i and y_i are the co-ordinates of the center of the MB and p_i 's are the affine parameters that we named *motion vector affine parameters*.

We defined a co-ordinate row vector $\mathbf{c}_i$ for block i as $\mathbf{c}_i = (x_i, y_i, 1)$. Next, the co-ordinate matrix C was formed by vertically concatenating the row vectors $\mathbf{c}_i$ for all blocks which were *not marked as outliers*. C is then, a $N \times 3$ matrix, where N is the number of macroblocks not marked as outliers. The vectors $\mathbf{V}_x$ and $\mathbf{V}_y$ were formed by collecting all the mvx_i and mvy_i respectively, for the MBs not marked as outliers. Lastly, the motion vector affine parameters were grouped together as $\mathbf{p}_x = (p_1, p_2, p_3)^T$ and $\mathbf{p}_y = (p_4, p_5, p_6)^T$. From these definitions, we could write $\mathbf{V}_x = C\mathbf{p}_x$ and $\mathbf{V}_y = C\mathbf{p}_y$ which were then solved for $\mathbf{p}_x$ and $\mathbf{p}_y$ using the pseudo-inverse matrix of C .

$$\begin{aligned} \mathbf{p}_x &= (C^T C)^{-1} C^T \mathbf{V}_x \\ \mathbf{p}_y &= (C^T C)^{-1} C^T \mathbf{V}_y \end{aligned} \quad (5.2.5)$$

After each iteration, we calculated the residual motion vector R_{mvi} as the absolute difference between the actual motion vector and the estimated motion vector, i.e., $R_{mvi} = |(mvx_i - mvx'_i) - (mvy_i - mvy'_i)|$ where mvx'_i and mvy'_i are the estimated components of the motion vector for macroblock i . We proposed an adaptive threshold mechanism to

Figure 5.9: The structure of hierarchical SVM.

reject outliers in the residual motion vectors. The threshold T was decided by comparing the mean of the residual motion vectors over all MBs with a small constant α and choosing the maximum of the two, i.e., $T = \max(\text{mean}(R_{mv_i}), \alpha)$. The role of α is to prevent the rejection of a large number of motion vectors if the mean of the residuals is very small. We chose α to be equal to 0.5. The algorithm was initialized by labeling all macroblocks as *inliers*.

Having determined the frame by frame global motion, we now describe our motion features for shot classification. The first class of motion features is *global camera motion description* that includes *camera horizontal motion* (CHD), *camera vertical motion* (CVD) and *camera zoom* (CZD). The definitions of the above features are:

$$\begin{aligned}
 CHD &= \frac{\sum_{i=0}^N p_{3i}}{N}, \\
 CVD &= \frac{\sum_{i=0}^N p_{6i}}{N}, \\
 CZD &= \sum_{i=0}^N (p_{1i} + p_{5i})/2
 \end{aligned} \tag{5.2.6}$$

where N is the number of frames included in a scene.

The second class of motion features are *total motion matrix*, *object motion matrix* and *camera motion matrix*, which describe the amounts of total motion, object motion and camera motion for each macroblock. These features are defined as:

$$tmx_i = |mvx_i|, \quad tmy_i = |mvy_i| \tag{5.2.7}$$

$$\begin{aligned} cmx_i &= \min(|mvx_i|, |gmvx_i|), \\ cmy_i &= \min(|mvy_i|, |gmvy_i|) \end{aligned} \quad (5.2.8)$$

$$\begin{aligned} omx_i &= \max(0, |mvx_i| - |gmvx_i|), \\ omy_i &= \max(0, |mvy_i| - |gmvy_i|) \end{aligned} \quad (5.2.9)$$

where gmv (denotes $gmvx_i, gmvy_i$) is the global motion vector field that is constructed at each macroblock with its centroid coordinates. Since the estimated ' $|gmv|$ ' may be larger than ' $|mv|$ ' (denotes mvx_i, mvy_i), we used cm (denotes cmx_i, cmy_i) as a minimum of $|mv|$ and $|gmv|$. Similarly, if $|gmv|$ is larger than $|mv|$, the om (denotes omx_i, omy_i) may be negative, since $om = tm - cm$. However, the amount of motion should not be negative. Therefore, we chose the maximum of either 0 or the difference. Then we accumulated tm , om and cm across a shot. Total Motion (TM), Camera Motion (OM) and Object Motion (CM) for a scene, k , with n number of frames, and r MVs for each frame are defined as

$$\begin{aligned} TMX_k &= \sum_{l=1}^n \sum_{i=1}^r tmx_{li}, \\ TMY_k &= \sum_{l=1}^n \sum_{i=1}^r tmy_{li} \end{aligned} \quad (5.2.10)$$

$$\begin{aligned} CMX_k &= \sum_{l=1}^n \sum_{i=1}^r cmx_{li}, \\ CMY_k &= \sum_{l=1}^n \sum_{i=1}^r cmy_{li} \end{aligned} \quad (5.2.11)$$

$$\begin{aligned} OMX_k &= \sum_{l=1}^n \sum_{i=1}^r omx_{li}, \\ OMY_k &= \sum_{l=1}^n \sum_{i=1}^r omy_{li} \end{aligned} \quad (5.2.12)$$

where TM (denotes TMX_k, TMY_k), CM (denotes CMX_k, CMY_k) and OM (denotes OMX_k, OMY_k) can be represented as a matrix of r products. Finally, the projected values of TM , CM and OM on the horizontal and vertical directions were used as feature vectors in our experiment.

A hierarchical SVM classifier was built for basketball scene classification. The structure of the classifier is shown in Figure 5.9. The basketball video scenes were classified into three classes including *court-view*, *bird's-eye-view* and *others* utilizing the texture-based features at the first run of the SVM classifier. The class including *courts-view* was further divided into three classes, *fast-motion court-view scene*, *slow-motion court-view scene* and *penalty scene*, based on the global camera motion information and the TM ,

CM , OM matrices. Lastly, the class *others* was divided into two classes, *in-court medium scene* and *out of court or close-up scene* based on the texture information and the TM , CM , OM matrices.

5.2.2 Multi-modal Structure Analysis and Event Detection

Utilizing the video analysis algorithms described earlier, we achieved high-level scene classifications based on information provided by the proposed ontology. In our previous work [125], Min Xu *et al.* detected specific audio keysounds by using the Hidden Markov Model (HMM) classifier in the basketball audio according to an audio classification scheme, defined in the basketball content description ontology. The next step is to combine the visual information and audio keysounds to infer a higher level of semantic understanding, e.g. detecting the positions of interesting events. At this stage, the system takes the results of video/audio segment classification as input, and combines the basketball domain knowledge to detect high-level interesting events, based on the event description encapsulated in the ontology. These interesting events in the basketball game include foul, steal ball and shot at the basket, etc. The goal of the proposed event detection program is to locate the positions of the events and label a scene with an event.

In our prior work [125] the audio keysound detection algorithm detects several audio keysounds that indicate potential events. However, all events could not be located precisely by only using audio information. For example, we could not distinguish whistling due to a time-out or foul, if only audio information provided. At the same time, although the types and orders of changes in the scenes generated by the scene classification algorithm provided us a good understanding of the structure of a basketball video, it was still very hard to detect the events by using only visual features. In order to locate the exact scenes where events occurred, we utilized the event description provided by the proposed ontology which is a multi-modal event detection mechanism to get benefits

from both visual/audio information and domain knowledge. From the domain knowledge of the basketball game, we know that the locations of events have strong relationships with camera movement and position. The global camera motion provides useful information for event detection, because the camera tracks the players or basketball during the game. Most of the events are located at the scenes with little camera motion. Also, the amount of camera motion in the next scene indicates the type of events that may occur in the current scene. To measure the amount of camera motion inside a scene precisely, we defined a feature called modified accumulated camera motion in time (MACM), as the product of accumulated camera motion in time (ACM) and dominant camera motion filter (DCMF) i.e.,

$$MACM = ACM \times DCMF \quad (5.2.13)$$

where,

$$ACM = \begin{cases} (CHD - CVD) \cdot e^{-CZD} \cdot D_s, & \text{If } CHD \cdot CVD > 0 \\ (CHD + CVD) \cdot e^{-CZD} \cdot D_s, & \text{If } CHD \cdot CVD < 0 \end{cases} \quad (5.2.14)$$

where the D_s is the time duration for a single scene and Equation 5.2.15 (DCMF) is used to filter out the conflict when two neighboring scenes have the same camera motion direction within a single shot.

$$DCMF = \begin{cases} 1 & \text{The first large camera motion scene in the long court-view shot} \\ 1 & \text{if } (ACM_{previous}) * (ACM_{current}) < 0 \\ 0 & \text{All out of court view scenes \& others} \end{cases} \quad (5.2.15)$$

where $ACM_{previous}$ indicates the previously detected large camera motion scene. If MACM is above T_α , then this scene has a large amount of camera motion. Based on the above definitions, we classified scenes into two groups. In our ontology, we gave this type of scene a specific label as: Offensive and Defensive exchange Interval (ODI) scene, if it contains a large amount of camera motion ($MACM > T_\alpha$). Otherwise, it is called

Figure 5.10: Example for ODI detection and event detection.

a non-ODI scene. Based on the sign of the MACM value, two types of ODI scenes can be distinguished: an ODI scene with left-to-right camera motion and an ODI scene with right-to-left camera motion. In the basketball video, some ODI scenes may not have been captured by the camera because it may have focused on a single player on the move. To detect these non-captured ODI scenes, the detected ODI scene sequence was further refined. This refinement was based on the observation that left-to-right change and right-to-left change should alternate in the video. We assigned a scene to be an ODI scene if there was a court-view scene between two scenes that have the same camera motion direction, otherwise we assigned the second of the two scenes as a non-ODI scene.

After ODI scene detection, we now describe how to locate the positions of events using ODI information. Consider two categories of events: 1) *shot at the basket*, *steal ball* and *offensive foul*, and 2) *defensive foul*. Events in category 1 are followed by an ODI scene, and events in category 2 are followed by a non-ODI scene. Therefore, we defined these two categories of events as *events followed by an ODI scene* and *events followed by a non-ODI scene*. Since we have described how to detect ODI scenes, locating the position for these two types of events can be achieved. Figure 5.10 shows an example of ODI detection. In the figure, the hashed blocks represent ODI scenes and the black

squares or round dots represent the points where events occur. For example, the event *foul* is an event occurring before the ODI scene, therefore, a round dot is placed on scene boundaries followed by an ODI scene. However, to present the time interval of one event, the scene before the boundary where the event occurs is used to represent that event. Facilitated by the descriptive power of the ontology language, the above high-level domain knowledge can be represented by a specific scene pattern and captured by our ontology as a description for event detection. After locating the potential positions of events using visual information, it is easy to combine the audio information to finally label a scene with an event. An algorithm for the detection of “foul” and “shot at the basket” was implemented according to their event descriptions, based on visual, audio information and some heuristic decision rules derived from the domain knowledge of basketball games. This is shown in Algorithm 2.

Algorithm 2 Event detection.

Input: Shot classification, ODI information and audio key sound

Output: The event label “foul” and “shot at the basket” for a scene

- 1: **if** The current scene is “court-view and non-ODI scene” or (the current scene is ODI scene and it’s neighboring scenes are non-court view scene) **then**
  - 2:   **if** The audio key sound “whistling” has been detected and it does not occur at the beginning of the scene **then**
  - 3:   **if** The next “court view scene” is an ODI scene **then**
  - 4:   Event “offensive foul” detected
  - 5:   **else if** The “whistling” occurs followed by a non-ODI scene or penalty scene **then**
  - 6:   Event “defensive foul” detected
  - 7:   **end if**
  - 8:   **else if** The audio key sound “Excited sound” has been detected in this scene and the next “court view scene is an ODI scene. **then**
  - 9:   Event “shot at the basket” detected
  - 10:   **end if**
  - 11: **end if**
-

5.2.3 MPEG-7 Compliant Annotation File Generation

The proposed audiovisual content description ontology provides the information for content and concept analysis of audiovisual sequences. However, the objective of designing a video analysis and annotation system is to facilitate content-based search and retrieval for video entities. Thus, we needed to store the results of processing and information in a highly structured format, which enables the annotation information to be queried and retrieved easily. MPEG-7 is a new multimedia standard, designed for describing multimedia content by providing a rich set of standardized descriptors and description schemas. The objective of the MPEG-7 standard is to allow interoperable searching, indexing, filtering and access of multimedia content by enabling interoperability among devices that deal with multimedia content description [1]. The Multimedia Description Schemes (MDSs) expand on the MPEG-7 descriptor by combining descriptors and other description schemes to provide descriptions for both the immutable metadata and the content of audio, visual and textual data. MPEG-7 MDSs consist of the following areas: Basic Elements, Content Description, Content Management, Content Organization, Navigation and Access. We utilized the description schemes (DSs) of content management and description provided by MPEG-7 MDSs to represent the results of the proposed semantic basketball video analysis and annotation system. Currently, the results for video analysis, event detection and audio key sounds detection are stored in two XML files based on different temporal decomposition of the video. The first file is utilized to store the video analysis and event detection information. There are two types of descriptors stored in this XML file, one type is manually annotated information and the other is automatically generated information. The manually annotated information includes:

- *CreationInformation DS*: This describes creation and production of the multimedia content. CreationInformation DS is composed of one Creation DS which contains information about the creation and production of the content not perceived in the

content, such as author, director, and characters.

- *TextAnnotation DS*: This DS contains a FreeTextAnnotation DS which provides a free text annotation for the video contents.

The automatically generated information includes:

- *MediaLocator DS*: This contains a MediaUri DS to describe the location of external media data.
- *MediaTime DS*: This DS is utilized to specify the time intervals of a video segment. It contains MediaTimePoint DS and MediaDuration DS which describe a time point using Gregorian date and day time, and duration of a time period according to date and day time respectively. By utilizing these two DSs, the location of one video segment in the entire video can be specified by time.
- *AudioVisual DS*: This DS is utilized to describe the temporal decomposition of a video entity. To provide a highly structured description for the video contents, the AudioVisualSegment DS is used to describe segments of audiovisual content and their attributes and structural decompositions. The structure of the content description can be described as: one AudioVisual DS represents one audiovisual entity and contains one TemporalDecomposition DS, in which it contains several AudioVisualSegment DSs to represent the shots composing an audiovisual entity. Similarly, under each AudioVisualSegment DS describing the shot, there is also a TemporalDecomposition DS, and the AudioVisualSegment DS under this level of TemporalDecomposition DS describes the temporal decomposition of the scenes.

In each TemporalDecomposition DS some attributes are generated automatically to describe a shot or scene, they are:

- *MediaTime DS*: This specifies the time intervals of a video segment.
- *Term DS*: This contains a termID tag to describe the scene classification information. The termID is defined in a ClassificationScheme DS based on the classification rules described in Subsection 5.2.2.
- *Event DS*: This describes an event, which is a semantic activity that takes place at a particular time or in a particular location.
- *FreeTextAnnotation DS*: This is utilized to describe ODI in a scene.

By using the DSs described above, all the results of video analysis and event detection can be represented in a standardized and highly structured format. The second XML file is used to describe the information of audio keysounds detection. The difference between the two XML files is that they have different temporal decompositions. In the first XML file, the temporal decomposition is based on the results of shot and scene detection, and in the second XML file, the temporal decomposition is based on the intervals of audio keysounds. Since audiovisual segments in our current system have fixed time intervals, the time duration of a keysound is calculated by summing up the time for the neighboring segments labeled as having the same audio keysound before generating an annotation file. That audio keysound information is then represented using MPEG-7 DS in a tree structure similar to the XML file described earlier.

5.2.4 Experimental Results

In this subsection, we present the results of our algorithm for shot/scene detection, scene classification and event detection. The test videos are two basketball videos from different matches with a total length of fifty minutes. The frame structure of the MPEG compressed test videos follows the standard GOP.

Table 5.1: Classification rates for level 1 and level 2 classes.

Classes	Correct classification rate (%)
Court-view	95.2
Bird's-eye-view	99.8
Others	95.0
Penalty scene	83.4
Slow-motion court-view scene	87.0
Fast-motion court-view scene	91.3
In-court medium view scene	88.0
Out of court or close-up scene	88.2

Video Shot and Scene Detection

The performance of the algorithm for hard-cut boundaries and gradual transitions is tabulated together. In the test videos, wipes and dissolves were utilized in the replay and close-up shot. Overall, the algorithm achieved 84.3% recall and 97.5% precision rates over 286 shot boundaries. We achieved a low recall rate as our color-based shot detection algorithm could not detect the gradual transitions accurately. However, the scene detection algorithm helped to reduce the non-detection of gradual transitions. Since scene detection is a very important step for generating the data which is utilized in the scene classification and semantic video analysis algorithms, the results of these subsequent algorithms can be indirectly used to measure the performance of scene boundary detection.

Video Scene Classification

Currently, a two-class SVM classifier is implemented to handle scene classification. For the case of multiple-classes classification, the classification rate of target class vs. others is used as the experimental results. Table 5.1 shows the results of scene classifications for level 1 and 2 scenes over a total of 1053 scenes. In the experiment, half of the data set was used as the training set, and the remainder was used as the test set.

Table 5.2: The statistics about the appearance of ODI detection.

Performance	Ground-truth	Recall (%)	Precision (%)
Before refining	93	91.4	93.4
After refining	93	97.8	92.0

Table 5.3: The statistics about the appearance of potential event detection.

Events	Ground-truth	Recall (%)	Precision (%)
Events before the ODI	85	87.0	92.5
Events without the ODI	29	76.6	76.7

Multi-modal Structure Analysis and Event Detection

Firstly, we show the experimental results of video analysis and event detection by using visual information only, followed by the multi-modal approach. Table 5.2 shows the results of ODI detection. The first row of the table shows the results of ODI detection using *MACM* and the second row of the table shows the results after applying the refining algorithm.

The ground-truth in Table 5.2 was defined as the actual number of ODIs that occurred including captured and non-captured. The results of potential event detection are shown in Table 5.3. From the table, we can see that an arbitrary number of events have been detected and classified to correct classes. Table 5.4 shows the results of event detection

Table 5.4: Results of event detection using multi-modal approach.

Performance	Ground-truth	Recall (%)	Precision (%)
Foul	25	96.1	96.1
Shot at the basket	51	94.5	89.5

using the multi-modal approach. In the table, the offensive foul and defensive foul are tabulated together and shown as “Foul.” Compared with Table 5.3, we conclude that the accuracy of event detection has improved significantly by combining the visual and audio information.

5.3 Discussion

This chapter proposes an ontology to provide the description for both audiovisual content and concept information. The MPEG-7 Ontology was imported in the proposed ontology to provide the description for basic audiovisual elements. The Content Analysis Ontology was built on top of the MPEG-7 Ontology and the audiovisual Domain Ontology. The definitions for high-level audiovisual events have been provided in the Concept Analysis Ontology, based on the content description provided. By implementing the proposed ontology, we introduced a basketball video analysis and annotation system. In this system, the content and concept analysis descriptions were imported from the proposed audiovisual content description ontology. Specifically, the basketball video analysis starts from shot and scene shot boundary detection, before going to semantic scene classification and audio key sound detection. After that, according to the event description provided by the ontology, a basketball event description algorithm was implemented. Lastly, the results of the basketball analysis, which are the annotations for a basketball audiovisual, were stored in an MPEG-7 Compliant format. The major open issues remaining in this area include enhancing the current ontology to enable it to provide the semantic description for a more complicated audiovisual semantic, such as, scenario or story, as well as more accurate content descriptions for semantic extraction from audiovisual sequences.

Chapter 6

Conclusions

6.1 Achievements and Contributions

The novelty of our research lies in designing and integrating Web ontologies into a content-based media retrieval system for better understanding of the multimedia data. The demand for indexing and retrieving multimedia data with an acceptable level of content interpretation provides the motivation for this integration. As we desire to utilize ontology technology in multimedia retrieval, many issues are involved in this research area. During the past five years of research, we have been looking at these problems from the perspective of creating high-level semantic descriptions for multimedia content, and also from the point of view of media annotation and retrieval. By utilizing widely applicable standardized languages, the feasibility of creating machine-processable semantic descriptions (ontologies) for the domain knowledge of multimedia content has been proven. A general framework is presented for combining the proposed ontology for still image description which provides domain knowledge descriptions, as well as an image retrieval system. A Web image retrieval system was implemented in which semantic knowledge is extracted from Web content. A video analysis and annotation system was designed and implemented based on the content and concept descriptions that are encapsulated in the proposed ontology for audiovisual descriptions. Moreover, we also investigated how to

achieve better media content description by designing and extracting new low-level and high-level feature descriptors.

To achieve the content description for still images, we presented an ontology for still image description. In the proposed ontology, we first created a Domain Description sub-ontology to provide a hierarchical description for domains and domain concepts. Next, a Textual Description sub-ontology was created to provide narrative domain descriptions for a given domain concept. Finally, a Visual Description sub-ontology was created to capture middle-level image content descriptions derived from low-level visual descriptors. Therefore, the proposed still image content description ontology encapsulates multi-modality information including both high-level and low-level descriptions for image content, together with the relevant domain knowledge. This multi-modality ontology becomes a rich resource for image content descriptions.

We proposed a general framework for the integration of the Web image retrieval system with the proposed ontology. In this system, the relevant images can be retrieved from the image database according to user input query keywords or narrative descriptions. Based on the visual concept descriptions provided by the proposed ontology, the middle-level descriptions can be automatically extracted from Web images by utilizing machine learning methods, including both model-based and discriminative approaches. The query keywords and descriptions are passed through the proposed ontology system to generate a domain knowledge enhanced query set. This query set was utilized to match against both textual-based descriptions extracted from Web page content, and middle-level descriptions derived from low-level image features. The final rank was worked out based on the scores of both textual and visual matches. The experiment's results demonstrated that by combining multi-modality information the retrieval efficiency is improved with the corresponding improvement in the average retrieval precision, compared to single modal and non-ontology approaches.

In the scenario of visual feature description, the context aware feature descriptor achieved better performance in both image retrieval and classification. A new texture low-level feature: texture correlogram was developed to capture textures as well as their spatial distribution information. Our experiment's results demonstrated that the proposed feature is able to achieve higher correct classification rates in semantically aware image classifications. Furthermore, we also presented a new approach to automatically detect the attentive region in professional photographs by analyzing high-level out-of-focus blur (OFB) information. By combining the metadata in EXIFs with low-level visual features, a new attentive region detection algorithm was implemented on the basis of the 'Edge Density' cue, which enables us to find the attentive region from the boundary map of the in-focus area in a photograph. The detected attentive regions provide useful perception knowledge for the images. Based on the extracted attentive regions and EXIFs, the estimation of geometric size for a focused object can be achieved.

Similar to still images, audiovisual data is another important component of multimedia data. In our research, we also proposed an audiovisual description ontology to provide semantic interpretations for audiovisual data at both content and concept levels. At the level of content description, the proposed ontology provides information on how to produce semantically aware video and audio segments. Beyond content analysis description, the proposed ontology also provides the descriptions for audiovisual domain concepts. The video and audio concepts defined in the proposed ontology include both simple description labels and domain audiovisual events. Utilizing this audiovisual description ontology, the content and concept analysis information becomes shareable and reusable.

We introduced a novel semantic analysis and annotation framework by using multi-modal analysis of video and audio information. A basketball video annotation and analysis system was implemented to test the proposed framework. According to the content

analysis ontology, semantically aware video segment detection was achieved. Next, motion features combined with other visual features were utilized in the scene classification to generate semantically meaningful labels for concept descriptions. By matching the event definitions provided by the ontology with the information that we extracted from the basketball sequence, high-level event detection was achieved. The experiment's results have demonstrated the effectiveness of event detection by utilizing a combination of multi-modal information and domain knowledge.

In summary, the contributions of our work include the following:

- Ontology for multimedia content description:
  - A study of the building of ontology for multimedia content description.
  - An ontology for still image content descriptions.
  - An ontology for audiovisual descriptions.
  - Methods for providing semantic interpretations of visual concept.
  - Methods for providing machine processable audiovisual event description.
- Web image retrieval based on a content description ontology:
  - A general framework for ontology-based Web image retrieval.
  - Methods for extracting textual-based image descriptions from Web pages.
  - Methods for matching query terms with textual-based image descriptions.
  - Methods for matching query terms with low-level visual descriptors.
  - Methods for combining both textual and visual matching scores.
- Basketball video analysis and annotation based on audiovisual description ontology:
  - Video segmentation based on visual and motion prediction information.

- Video scene classification based on domain knowledge.
- Basketball event detection by using multi-modal information.
- Basketball video annotation generation.
- Multimedia content description enhancement:
  - A low-level texture feature description for capturing both texture and spatial distribution of texture information.
  - Methods for detecting attentive regions from images using high-level metadata and out-of-focus blur information.
  - Methods for extracting geometry size of the foreground of the focused object.

6.2 Future Work

Possible future work could focus on improving the ontology for the semantic description of multimedia content. We could expand the ontology system based on the current one. New vocabularies for image, video and audio content description should be investigated and more complicated relationships between concepts should be specified.

There are some open issues for ontology building. The first issue we could investigate into is how to achieve higher efficiency for multimedia content description by introducing other ontology languages. The current prototype ontology we defined has low expressivity. This means the proposed ontology emphasizes on defining concepts or atomic types and the relationship between them in a hierarchy instead of taxonomy of relations, reified statements or semantic entailments. With more concepts and relationships are defined as axioms, the semantic expressivity will become higher and higher. This may introduce a performance gap when we query the knowledge base. To balance the expressivity of the ontology and system performance, EL++ [131] and DL-Lite [132] ontology languages

could be a solution. The EL++ logic eliminates the `allValuesFrom` restriction, retaining `someValuesFrom`, in order to obtain tractability and DL-Lite is a fragment of OWL DL especially tailored for handling efficiently large number of facts. For both languages, some restrictions are revised or simplified compared OWL. They help to reduce performance bottle neck but provide efficient query answering on the data at the same time.

As more and more domain specific ontologies are created, there is a requirement to merge these ontologies and use the information from different ontologies to enhance multimedia content description. This type of mechanism improves the scalability of a single ontology. Moreover, there is one issue regarding domain knowledge validation that needs to be looked into if we want to merge the information from different ontologies. As we know, the concept semantics extracted from different resources can be merged into a single ontology. However, providing a trustworthy ontology, validating the semantics extracted from different resources, and resolving the conflicts between the semantics are issues still under investigation. More research resources are required to work on the higher levels of the semantic Web which are “prove and trust”. The design and development of the proposed ontology have provided a good knowledge base for multimedia content description. In this thesis, the multimedia content description is matched against query terms in a very low-level style. The matching mechanisms try to work out a similarity factor or find a specific pattern for each multimedia entity.

To fully utilize the descriptive power of the proposed ontology, some high-level matching mechanisms should be investigated, such as rule-based reasoning engines. Rule-based reasoning is a particular type of reasoning which uses the “if-then-else” rule statements. Rules mentioned here are simple patterns and an inference engine searches for patterns in the rules that match patterns in the data. The most common expert system is rule-based, containing a knowledge base (rules) and an inference engine (routing mechanism) which analyzes fact patterns and matches the applicable rules. Fact patterns are analyzed until

either the goal succeeds or all of the rules are processed and the goal fails. In a multimedia retrieval system, the ontology created with OWL could provide an abundance of rules applicable to reasoning. Combined with the specific inference engine, a retrieval system can answer some complicated questions like, “Find a reference to the most recent paper about SHOE coauthored by James Handler”. The reasoning can also be used to generate new statements by exposing previously unrecognized relationships. Hence, the inference can be implemented as a measure to expand the existing ontology. In order to follow this approach, a set of pre-defined rules should be applied to the metadata associated with the acquired set of media objects to determine the semantic relationships between them. Fuzzy Markup Language (FML) [133, 134, 135] is one approach for implementing the reasoning mechanisms. FML takes fuzzy knowledge base, fuzzy rule base and real input. It generates real output base on the decision made by inference engine, fuzzification subsystem and defuzzification subsystem. To utilize FML in our scenario, we need to convert our ontology as a fuzzy knowledge base, which needs to have more research efforts. Different from the fuzzy ontology that presented in [13], in our prototype ontology, the information is explicitly extracted from online dictionary and the visual description are generated from machine learning method. These limited our ontology is not a fuzzy knowledge base and therefore, hard to apply any fuzzy rule-based inference on that. Without a proper knowledge, FML only introduces more false alarmed results into existing system.

On the other hand, as the basis of high-level semantic descriptions, low-level feature or attribute descriptors are also very essential for retrieval efficiency and system performance. Two general issues should be addressed in this area: one is designing an efficient retrieval algorithm for high-dimensional conditions multimedia retrieval, the other is designing a formalized object detection and feature description mechanism. Low-level feature or attribute extraction is pre-requirement for all multimedia analysis or retrieval

system. Defining more compact and efficient features can reduce the cost of visual concept extraction and description on both space and time. Moreover, the development of object detection technology can greatly improve the effectiveness of our ontology system. With more visual concepts defined in the ontology, we can tackle the multimedia content description for complicated human conceptual items. With the limitation of hard to achieve high-level object recognition and detection, we still be able to look into the topic of fuzzy descriptions for visual concepts. By introducing the different object filters, recognizers or classifiers, we should extract object information on different fuzzy layers with less computational time and effort. Therefore, the information can be further merged into fuzzy knowledge base and make it processable by fuzzy inference engine. This hopefully reduces the gap for both computation and match making.

List of Publications

- [1] H. Wang, S. Liu and L. T. Chia, "Image Retrieval with a Multi-Modality Ontology Multimedia Systems", *Multimedia Syst.*, vol. 13, no. 5-6, pp. 379-390, 2008.
- [2] H. Wang, L. T. Chia and S. Liu, "Image Retrieval ++ – Web Image Retrieval with An Enhanced Multi-Modality Ontology", *Multimedia Tools and Applications*, vol. 39, no. 2, pp. 189-215, 2008.
- [3] H. Wang, L. T. Chia and S. Liu, "Semantic Retrieval with Enhanced Matchmaking and Multi-Modality Ontology", In *Proceeding of International Conference on Multimedia & Expo 2007*.
- [4] H. Wang, S. Liu and L. T. Chia, "Does Ontology help in Image Retrieval?", In *Proceeding of Asia-Pacific Workshop on Visual Information Processing 2006 (Best Paper Award)*.
- [5] H. Wang, S. Liu, and L. T. Chia. "Does Ontology help in Image Retrieval? - A Comparison between Keyword, Text Ontology and Multi-Modality Ontology Approaches". In *Proceeding of ACM Multimedia 2006*.
- [6] S. Liu, M. Xu, H. Yi, L. T. Chia and D. Rajan. "Multi-modal Semantic Analysis and Annotation for Basketball Video". *Special Issue on Information Mining from Multimedia Databases, EURASIP Journal on Applied Signal Processing*. Publication Date: 3rd Quarter, 2005
- [7] S. Liu, L. T. Chia, and D. Rajan. "Attention Region Selection with Information from Professional Digital Camera". In *ACM Multimedia 2005*, Singapore, November 2005.
- [8] S. Liu, H. Yi, L. T. Chia, and D. Rajan. "Adaptive Hierarchical Multi-class SVM Classifier for Texture-based Image Classification". In *IEEE ICME2005*, Amsterdam, The Netherlands, July 2005.
- [9] Y. K. Lai, S. Liu, L. T. Chia and S. Chan. "Semantic Knowledge Building for Image Database by Analyzing Web Page Contents". In *IEEE ICME2005*, Amsterdam, The Netherlands, July 2005.
- [10] S. Liu, L. T. Chia and S. Chan. "Ontology for nature-scene image retrieval". In *Proceedings OTM confederated international conferences*,

pp. 1050-1061, Cyprus, October 2004.

[11] S. Liu, H. Yi, L. T. Chia, D. Rajan and S. Chan. “Semantic Analysis of Basketball Video Using Motion Information”. In Proceedings IEEE PCM2004, pp. 65-72, Tokyo, Japan, November 2004.

[12] S. Liu and L. T. Chia, “Integrating Ontology into Content-based Image Retrieval”. Proceedings of International Workshop On Advanced Image Technology (IWAIT 2004), Singapore, January 2004.

Bibliography

- [1] B. S. Manjunath, P. Salembier, and T. Sikora, *Introduction to MPEG-7*, John Wiley & Sons, Ltd, 2002.
- [2] A. W.M. Semulders, M. Worring, S. Santini, A. Gupta, and R. Jain, “Content-based image retrieval at the end of the early years,” *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 22, no. 12, pp. 1349–1379, DECEMBER 2000.
- [3] T. Berners-Lee, J. Hendler, and O. Lassila, “The semantic web,” *Scientific American*, May 2001, Available: <http://www.scientificamerican.com>.
- [4] W3C, *Resource Description Framework (RDF) Model and Syntax Specification*, February 1999, Available: <http://www.w3.org/TR/1999/REC-rdf-syntax-19990222/>.
- [5] W3C, *XML Schema Part 0: Primer*, May 2001, Available: <http://www.w3.org/TR/xmlschema-0>.
- [6] T. R. Gruber, “A translation approach to portable ontologies,” *Knowledge Acquisition*, vol. 5, no. 2, pp. 199–220, May 1993, Available: <http://ksl-web.stanford.edu>.
- [7] T. R. Gruber, “Toward principles for the design of ontologies used for knowledge sharing,” *Formal Ontology in Conceptual Analysis and Knowledge Representation*, 1993, Available: <http://ksl-web.stanford.edu>.

-
- [8] T. R. Gruber, “The role of common ontology in achieving sharable, reusable knowledge bases,” in *Principles of Knowledge Representation and Reasoning: Proceedings of the Second International Conference*. 1991, pp. 601–602, Morgan Kaufmann.
  - [9] W3C, *Web Ontology Language (OWL) Use Cases and Requirements*, February 2003, Available: <http://www.w3.org/TR/2003/WD-webont-req-20030203/>.
  - [10] D. M. Le and L. Lau, “An open architecture for ontology-enabled content management systems: A case study in managing learning objects,” in *Proceedings of On the Move to Meaningful Internet Systems 2006: CoopIS, DOA, GADA, and ODBASE*, 2006, pp. 772–790.
  - [11] S. Bechhofer, Y. Yesilada, R. Stevens, S. Jupp, , and B. Horan, “Using ontologies and vocabularies for dynamic linking,” *IEEE Internet Computing*, vol. 12, no. 3, pp. 32–39, 2008.
  - [12] C. S. Lee, Y. F. Kao, Y. H. Kuo, , and M. H. Wang, “Automated ontology construction for unstructured text documents,” *Data & Knowledge Engineering*, vol. 60, no. 3, pp. 547–566, 2007.
  - [13] Z. W. Jian C. S. Lee and L. K. Huang, “A fuzzy ontology and its application to news summarization,” *IEEE Transactions on Systems, Man and Cybernetics Part B*, vol. 35, no. 5, pp. 859–880, 2005.
  - [14] R. R. Yager and F. E. Petry, “A multi-criteria approach to data summarization using concept ontologies,” *IEEE Transactions on Fuzzy Systems*, vol. 14, no. 6, pp. 767–780, 2006.
  - [15] C. S. Lee, M. H. Wang, and J. J. Chen, “Ontology-based intelligent decision support agent for cmmi project monitoring and control,” *International Journal of Approximate Reasoning*, vol. 48, no. 1, pp. 62–76, 2008.

-
- [16] W3C, *Image Annotation on the Semantic Web*, March 2006, Available: <http://www.w3.org/TR/2006/WD-swbp-image-annotation-20060322/>.
  - [17] *ISO/IEC 15938-3 FCD Information Technology - Multimedia Content Description Interface - Part 3: Visual*, March 2001.
  - [18] *ISO/IEC 15938-4 FCD Information Technology - Multimedia Content Description Interface - Part 4: Audio*, March 2001.
  - [19] *ISO/IEC 15938-5 FCD Information Technology - Multimedia Content Description Interface - Part 5: Multimedia Description Schemes*, March 2001.
  - [20] T. Bray, “What is rdf?,” January 2001, Available: <http://www.xml.com/pub/a/2001/01/24/rdf.html>.
  - [21] J. Jeon, V. Lavrenko, and R. Manmatha, “Automatic image annotation and retrieval using cross-media relevance models,” in *SIGIR '03: Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, New York, NY, USA, 2003, pp. 119–126, ACM.
  - [22] K. Deschacht and M. F. Moens, “Finding the best picture: Cross-media retrieval of content,” in *Advances in Information Retrieval*, 2008, pp. 539–546.
  - [23] J. Calvo, M. A. Patricio, C. Cuvillo, and L. Usero, “Context information for human behavior analysis and prediction,” in *IWINAC '07: Proceedings of the 2nd international work-conference on Nature Inspired Problem-Solving Methods in Knowledge Engineering*, Berlin, Heidelberg, 2007, pp. 241–250, Springer-Verlag.
  - [24] J. Hunter, “Adding multimedia to the semantic web - building an mpeg-7 ontology,” *International Semantic Web Working Symposium (SWWS)*, August 2001.
  - [25] “acemedia: Final public report,” Jan 2008, Available: <http://www.acemedia.org/aceMedia>.
-

-
- [26] T. Berners-Lee, *RFC 2396 Uniform Resource Identifiers (URI): Generic Syntax*, August 1998, Available: <http://www.ietf.org/rfc/rfc2396.txt>.
- [27] S. B. Palmer, “The semantic web: An introduction,” September 2001, Available: <http://infomesh.net/2001/swintro/#simpleData>.
- [28] S. Cranfield and J. Pan, “Bridging the gap between the model-driven architecture and ontology engineering,” *International Journal of Human-Computer Studies*, vol. 65, no. 7, pp. 565–609, 2007.
- [29] W3C, *RDF Schema Specification 1.0*, March 2000, Available: <http://www.w3.org/TR/rdf-schema/>.
- [30] D. L. McGuinness, R. Fikes, J. Hendler, and L. A. Stein, “Daml+oil: An ontology language for the semanticweb,” *IEEE Intelligent Systems*, pp. 72–80, September/October 2002, Available: www.daml.org/2000/10/damlont.html.
- [31] D. Fensel et al., “Oil: An ontology infrastructure for the semantic web,” *IEEE Intelligent Systems*, vol. 16, no. 2, pp. 38–45, March/April 2001.
- [32] S. S. Weng and H. L. Chang, “Using ontology network analysis for research document recommendation,” *Expert Systems with Applications*, vol. 34, no. 3, pp. 1857–1869, 2008.
- [33] S. Hastings, “Query categories in a study of intellectual access to digitized art images,” in *Proceedings of the 1995 Annual Meeting of the American Society for Information Science*, 1995, pp. 3–8.
- [34] A. K. Jain and A. Vailaya, “Shape-based retrieval: A case study with trademark image databases,” *Pattern Recognition*, vol. 31, no. 9, pp. 1369–1390, January 1998.
- [35] J. P. Eakins, J. M. Boardman, and M. E. Graham, “Similarity retrieval of trademark images,” *IEEE Multimedia*, vol. 5, no. 2, pp. 53–63, Apr.-June 1998.
-

-
- [36] M. Flickner, H. Sawhney, W. Niblack, J. Ashley, Q. Huang, M. Gorkani B. Dom, J. Hafner, D. Lee, D. Petkovic, D. Steele, and P. Yanker, "Query by image and video content: The qbic system," *IEEE Computer*, 1995.
  - [37] M. J. Swain and B. H. Ballard, "Color indexing," *Computer Vision*, vol. 7, no. 1, pp. 11–32, 1991.
  - [38] T. Gevers and A. W.M. Smeulders, "Pictoseek: Combining color and shape invariant features for image retrieval," *IEEE Trans. Image Processing*, vol. 9, no. 1, pp. 102–119, 2000.
  - [39] G. D. Finlayson, "Color in perspective," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 18, no. 10, pp. 1034–1038, October 1996.
  - [40] D. A. Forsyth, "A novel algorithm for color constancy," *Computer Vision*, vol. 5, no. 1, pp. 5–36, 1990.
  - [41] T. Gevers and A. W.M. Smeulders, "Content-based image retrieval by viewpoint-invariant image indexing," *Image and Vision Computing*, vol. 17, no. 7, pp. 475–488, 1999.
  - [42] B. V. Funt and G. D. Finlayson, "Color constant color indexing," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 17, no. 5, pp. 522–529, May 1995.
  - [43] E. Todt and C. Torras, "Color constancy for landmark detection in outdoor environments," in *Proceedings. 4th European Workshop on Advanced Mobile Robots (Eurobot'01)*, September 2001, Available: <http://www-iri.upc.es/people/torras/articles/eurobot01.pdf>.
  - [44] B. S. Manjunath and W. Y. Ma, "Texture features for browsing and retrieval of image data," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 18, no. 8, pp. 837–842, 1996.
  - [45] S. Krishnamachari and R. Chellappa, "Multiresolution gauss-markov random field models for texture segmentation," *IEEE Trans. Image Processing*, vol. 6, no. 2, 1997.
-

-
- [46] G. L. Gimel'farb and A. K. Jain, "On retrieving textured images from an image database," *Pattern Recognition*, vol. 29, no. 9, pp. 1461–1483, 1996.
- [47] H. C. Lin, L. L. Wang, and S. N. Yang, "Pattern analysis and machine color image retrieval based on hidden markov models," *Pattern Recognition*, vol. 6, no. 2, pp. 332–339, 1997.
- [48] I. Daubechies, "Ten lectures on wavelets," *Philadelphia: SIAM*, 1992.
- [49] A. Laine and J. Fan, "Descriptors of spatially filtered color image regions," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 15, no. 11, pp. 1186–1191, November 1993.
- [50] H. Choi and R. Baraniuk, "Multiscale texture segmentation using wavelet-domain hidden markov models," in *Proceedings. 32nd Asilomar Conf. Signals, Systems, and Computers*, 1998, vol. 2, pp. 1692–1697.
- [51] Lowe and G. David, "Object recognition from local scale-invariant features," in *Proceedings of the International Conference on Computer Vision*, pp. 1150–1157.
- [52] P. M. Tardif and A. Zaccarin, "Multiscale autoregressive image representation for texture segmentation," *Image Processing VIII*, vol. 3, no. 026, pp. 327–337, 1997.
- [53] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Van Gool, "A comparison of affine region detectors," *International Journal of Computer Vision*, vol. 65, no. 1/2, pp. 43–72, 2005.
- [54] C. S. Li and V. Castelli, "Deriving texture feature set for content-based retrieval of satellite image database," in *Proceedings. Int'l Conf. Image Processing*, 1997.
- [55] S. Liu, H. Yi, L. T. Chia, D. Rajan, and S. Chan, "Semantic analysis of basketball video using motion information," in *Proceedings of*
-

- the Advances in Multimedia Information Processing - PCM 2004*. Springer Berlin / Heidelberg, October 2004, pp. 65–72.
- [56] H. Yi, D. Rajan, and L. T. Chia, “A unified approach to detection of shot boundaries and subshots in compressed video,” in *Proceedings of IEEE International Conference on Image Processing*, Sep 2003, vol. 3, pp. II–1005–8.
  - [57] H. Yi, D. Rajan, and L. T. Chia, “A motion-based scene tree for compressed video content management,” *Image and Vision Computing*, vol. 24, no. 2, pp. 131–142, 2006.
  - [58] H. Yi, D. Rajan, and L. T. Chia, “A new motion histogram to index motion content in video segments,” *Pattern Recognition Letters*, vol. 26, no. 9, pp. 1221–1231, 2005.
  - [59] D. A. Forsyth and M. M. Fleck, “Automatic detection of human nudes,” *Computer Vision*, vol. 32, no. 1, pp. 63–77, 1999.
  - [60] A. Gupta and R. Jain, “Visual information retrieval,” *Comm. ACM*, vol. 40, no. 5, pp. 71–79, 1997.
  - [61] A. K. Jain and A. Vailaya, “Image retrieval using color and shape,” *Pattern Recognition*, vol. 29, no. 8, pp. 1,233–1,244, 1996.
  - [62] J. Huang, S. R. Kumar, M. Mitra, W. J. Zhu, and R. Zabih, “Image retrieval using color and shape,” *Computer Vision*, vol. 35, no. 3, pp. 245–268, 1999.
  - [63] P. L. Rosin, “Edges: Saliency measures and automatic thresholding,” *Machine Vision Applications*, vol. 9, no. 7, pp. 139–159, 1997.
  - [64] C. Carson, S. Belongie, H. Greenspan, and J. Malik, “Region-based image querying,” *CVPR97Workshop on Content-Based Access of Imageand Video Libraries*, 1997.
  - [65] L. M. Kaplan et al., “Fast texture database retrieval using extended fractal features,” *Storage and Retrieval for Image and Video Databases, VI*, vol. 3, no. 312, pp. 162–173, 1999.

-
- [66] P. C. Yuen, G. C. Feng, , and D. Q. Tai, “Human face image retrieval system for large database,” in *Proceedings. 14th Int’l Conf. Pattern Recognition*, 1998, vol. 2, pp. 1585–1588.
  - [67] F. Korn, N. Sidiropoulos, C. Faloutsos, E. Siegel, and Z. Protopapas, “Efficient and effective nearest neighbor search in a medical image database of tumor shapes,” *Image Description and Retrieval*, pp. 17–54, 1998.
  - [68] H. Hatano, “Image processing and database system in the national museum of western art,” *Int’l J. Special Libraries*, vol. 30, no. 3, pp. 259–267, 1996.
  - [69] B. M. Mehtre, M. S. Kankanhalli, and W. F. Lee, “Shape measures for content based image retrieval: A comparison,” *Information Procesing Management*, vol. 33, no. 3, pp. 319–337, 1997.
  - [70] C. S. Fuh, S. W. Cho, and K. Essig, “Hierarchical color image region segmentation for content-based image retrieval system,” *IEEE Trans. Image Processing*, vol. 9, no. 1, pp. 156–163, 2000.
  - [71] J. R. Smith and S. F. Chang, “Integrated sspatial and feature image query,” *Multimedia Systems*, vol. 7, no. 2, pp. 129–140, 1999.
  - [72] S. K. Chang and A. D. Hsu, “Image-information systems where do we go from here?,” *IEEE Trans. Knowledge and Data Eng.*, vol. 4, no. 5, pp. 413–442, October 1992.
  - [73] D. Tjondronegoro and A. Spink, “Web search engine multimedia functionality,” *Information Processing & Management*, vol. 44, no. 1, pp. 340–357, 2008.
  - [74] R. Troncy, “Integrating structure and semantics into audio-visual documents,” in *Proceedings of 2nd International Semantic Web Conference (ISWC03)*, 2003, p. 566581.
  - [75] C. Tsinaraki, P. Polydoros, N. Moumoutzis, and S. Christodoulakis, “Integration of owl ontologies in mpeg-7 and tv-anytime compliant
-

- semantic indexing,” in *16th International Conference on Advanced Information Systemes Engineering (CAiSE04)*, 2004.
- [76] R. Garcia and O. Celma, “Integration and retrieval of multimedia metadata,” in *5th International Workshop on Knowledge Markup and Semantic Annotation (SemAnnot05)*, 2005.
- [77] R. Arndt, R. Troncy, S. Staab, L. Hardman, and M. Vacura, “Comm: Designing a well-founded multimedia ontology for the web,” in *6th International Semantic Web Conference (ISWC’2007)*, 2007.
- [78] A. Gangemi, N. Guarino, C. Masolo, A. Oltramari, and L. Schneider, “Sweetening ontologies with dolce,” in *Proceedings of 13th International Conference on Knowledge Engineering and Knowledge Management (EKAW)*, 2002, p. 166181.
- [79] F. E. Petry and R. R. Yager, “Evidence resolution using concept hierarchies,” *IEEE Transactions on Fuzzy System*, vol. 16, no. 2, pp. 299–308, 2008.
- [80] A. Jaimes and S. F. Chang, “Learning structured visual detectors from user input at multiple levels,” *International Journal of Image and Graphics*, vol. 1, no. 3, pp. 415–444, 2001.
- [81] P. Stanchev, “Object-oriented image model,” in *Proceedings of Technology of Object-Oriented Languages and Systems Tools Eastern Europe99*, June 1999, pp. 98–109.
- [82] W3C, *Web Ontology Language (OWL) Use Cases and Requirements*, February 2003.
- [83] G. A. Miller, R. Beckwith, C. D. Fellbaum, D. Gross, and K. J. Miller, “Wordnet: An on-line lexical database,” *International Journal of Lexicography* 3, pp. 235–244, 1990.
- [84] Y. Nah, B. Lee, and J. Kim, “Visual media retrieval framework using web service,” in *Web and Communication Technologies and Internet-Related Social Issues - HSI 2005*, 2005, pp. 104–113.

-
- [85] S. Dasiopoulou, I. Kompatsiaris, and L. J. B. Nixon, “A framework for ontology-driven multimedia analysis and composition,” in *Proceedings of Image Analysis for Multimedia Interactive Services*, 2007, pp. 47–51.
  - [86] J. Hai, S. Aobing, Z. Ran, H. Ruhan, Z. Qin, S. Yingjie, and Y. Wen, “Content and semantic context based image retrieval for medical image grid,” in *GRID '07: Proceedings of the 8th IEEE/ACM International Conference on Grid Computing*, Washington, DC, USA, 2007, pp. 105–112.
  - [87] H. Feng, R. Shi, and T. S. Chua, “A bootstrapping framework for annotating and retrieving www images,” in *Proceedings of the ACM International Conference on Multimedia*, 2004, pp. 960–967.
  - [88] P. Antonio, P. Antonio, and T. Letizia, “Multimedia knowledge management using ontologies,” in *MS '08: Proceeding of the 2nd ACM workshop on Multimedia semantics*, New York, NY, USA, 2008, pp. 24–31, ACM.
  - [89] A. Pentland, R. W. Picard, and S. Sclaroff, “Photobook:content-based manipulation of image databases,” *SPIE Vol 2185: Storage and Retrieval for Image and Video Databases II*, pp. 34–47, February 1994.
  - [90] J. R. Smith and S. F. Chang, “Visualeek: A fully automated content-based image query system,” in *Proc. ACM Multimedia, (Boston, MA)*, November 1996, pp. 87–98.
  - [91] S. Mehrotra, Y. Rui, M. Ortega, and T. S. Huang, “Supporting content-based queries over images in mars,” in *Proc. IEEE Int. Conf. on Multimedia Computing and Systems, (Ontario, Canada)*, June 1997, pp. 632–633.
  - [92] L. T. Chia Y. K. Lai, S. Liu and S. Chan., “Semantic knowledge building for image database by analyzing web page contents,” in *IEEE ICME2005*, 2005.
-

-
- [93] T. Joachims, “A probabilistic analysis of the Rocchio algorithm with TFIDF for text categorization,” in *Proceedings of ICML-97, 14th International Conference on Machine Learning*, 1997, pp. 143–151.
  - [94] N. Seco, T. Veale, and J. Hayes, “An intrinsic information content metric for semantic similarity in wordnet,” in *Proceedings of ECAI’2004, the 16th European Conference on Artificial Intelligence*, 2004.
  - [95] J. A. Hartigan, *Clustering Algorithms*, New York: John Wiley & Sons, Inc, 1975.
  - [96] D.W. Scott, *Multivariate Density Estimation: Theory, Practice, and Visualization*, Wiley-Interscience, 1992.
  - [97] K. Grauman and T. Darrell, “The pyramid match kernel: discriminative classification with sets of image features,” in *Tenth IEEE International Conference on Computer Vision*, 2005, pp. 1458–1465.
  - [98] S. Lazebnik, C. Schmid, and J. Ponce, “Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2006.
  - [99] K. Grauman and T. Darrell, “Unsupervised learning of categories from sets of partially matching image features,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2006.
  - [100] R. Teodorescu, D. Racocanu, W. K. Leow, and V. Cretu, “Prospective study for semantic inter-media fusion in content-based medical image retrieval,” *CoRR*, vol. abs/0811.4717, 2008.
  - [101] J. P. Chevallet, J. H. Lim, and D. T. H. Le, “Domain knowledge conceptual inter-media indexing: application to multilingual multimedia medical reports,” in *CIKM ’07: Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*, New York, NY, USA, 2007, pp. 495–504, ACM.
-

-
- [102] Y. K. Lai, S. Liu, L. T. Chia, and S. Chan, "Semantic knowledge building for image database by analyzing web page contents," in *Proceedings of the IEEE ICME2005*, July 2005.
  - [103] J. Huang, S. Kumar, M. Mitra, W. Zhu, and R. Zabih, "Image indexing using color correlograms," in *Proc. IEEE Comp. Soc. Conf. Comp. Vis. and Patt. Rec.*, 1997, pp. 762–768.
  - [104] J. Malik, S. Belongie, J. Shi, and T. K. Leung, "Textons, contours and regions: Cue integration in image segmentation," in *ICCV (2)*, 1999, pp. 918–925.
  - [105] R. M. Haralick, "Statistical and structural approaches to texture," *Proceedings of IEEE*, vol. 67, no. 5, pp. 786–804, 1979.
  - [106] L. Q. Chen, X. Xie, X. Fan, W. Y. Ma, H. J. Zhang, and H. Q. Zhou, "A visual attention model for adapting images on small displays," *ACM Multimedia Systems Journal*, vol. 9, no. 4, pp. 352–364, 2003.
  - [107] J. Bormans and K. Hill, "Mpeg-21 overview v.5," *ISO/IEC JTC1/SC29/WG11/N5231*, Oct. 2002.
  - [108] U. Rutishauser, D. Walther, C. Koch, and P. Perona, "Is bottom-up attention useful for object recognition?," in *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, 2004, vol. 2, pp. 37–44.
  - [109] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 11, pp. 1254–1259, November 1998.
  - [110] Y. F. Ma and H. J. Zhang, "Contrast-based image attention analysis by using fuzzy growing," in *Proc. ACM Multimedia*, Berkeley, CA USA, November 2003.
  - [111] L. Itti and C. Koch, "A comparison of feature combination strategies for saliency-based visual attention systems," in *Proc. SPIE Human*
-

- Vision and Electronic Imaging IV (HVEI'99)*, San Jose, CA, 1999, vol. 3644, pp. 473–482.
- [112] M. Lux, J. Becker, and H. Krottmaier, “Caliph & emir: Semantic annotation and retrieval in personal digital photo libraries,” in *Proceedings of CAiSE '03 Forum at 15th Conference on Advanced Information Systems Engineering*, June 2003, pp. 85–89.
  - [113] U. Gargi, Y. Deng, and D. R. Tretter, “Managing and searching personal photo collections,” *HP Laboratories Technical Report HPL-2002-67*, 2002.
  - [114] Japan Electronics and Information Technology Industries Association, “Exchangeable image file format for digital still cameras: Exif version 2.2 (jeita cp-3451),” April 2002.
  - [115] G. Z. Fa and L. T. Chia, “Mpeg7 annotation tool for images & graphics files - ii,” in *Technical report of final year project*, 2004.
  - [116] U. Rutishauser, D. Walther, C. Koch, and P. Perona, “Is bottom-up attention useful for object recognition?,” in *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE, July 2004, pp. 37–44.
  - [117] J. Nam and A. H. Tewfik, “Combined audio and visual streams analysis for video sequence segmentation,” in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, April 1997, vol. 4, pp. 2665–2668.
  - [118] C. Saraceno and R. Leonardi, “Identification of story units in audio-visual sequences by joint audio and video processing,” in *Proceedings of IEEE International Conference on Image Processing*, 1998, pp. 358–362.
  - [119] Y. Gong, T. S. Lim, H. C. Chua, H. Zhang, , and M. Sakauchi, “Automatic parsing of tv soccer programs,” in *Proc. of International Conference on Multimedia Computing and System*, May 1995, pp. 167–174.

-
- [120] P. Xu, L. Xie, S. F. Chang, A. Divakaran, A. Vetro, and H. Sun, "Algorithms and systems for segmentation and structure analysis in soccer video," in *Proc. of IEEE International Conference on Multimedia and Expo*, August 2001, pp. 22–25.
- [121] Y. P. Tan, D. D. Saur, S. R. Kulkarni, and P. J. Ramadge, "Rapid estimation of camera motion from compressed video with application to video annotation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 10, no. 1, pp. 133–146, February 2000.
- [122] A. Ekin, A. M. Tekalp, and R. Mehrotra, "Automatic soccer video analysis and summarization," *IEEE Trans. on Image Processing*, vol. 12, no. 8, pp. 796–807, July 2003.
- [123] H. Lu and Y. P. Tan, "Content-based sports video analysis and modeling," in *Proceedings of Seventh International Conference on Control, Automation, Robotics And Vision*, December 2002, pp. 1198–1203.
- [124] Y. Fu, A. Ekin, A. Tekalp, and R. Mehrotra, "Temporal segmentation of video objects for hierarchical objectbased motion description," *IEEE Trans. on Image Processing*, vol. 11, no. 2, pp. 135–145, February 2002.
- [125] S. Liu, M. Xu, H. Yi, L. T. Chia, and D. Rajan., "Multi-modal semantic analysis and annotation for basketball video," in *Special Issue on Information Mining from Multimedia Databases, EURASIP Journal on Applied Signal Processing*.
- [126] H. Yi, D. Rajan, and L. T. Chia, "A unified approach to detection of shot boundaries and subshots in compressed video," in *Proceedings of IEEE International Conference on Image Processing*, Sep 2003, vol. 3, pp. II–1005–8.
- [127] L. H. Siew, R. M. Hodgson, and E. J. Wood, "Texture measures for carpet wear assessment," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 10, no. 1, pp. 92–105, January 1988.
-

-
- [128] R.M. Haralick, K. Shanmugam, and I. Dinstein, “Textural features for image classification,” *IEEE Trans. Systems, Man and Cybernetics*, vol. 3, no. 6, pp. 610–612, November 1973.
  - [129] C. Stiller and J. Konrad, “Estimating motion in image sequences,” *IEEE Signal Processing Magazine*, vol. 16, no. 4, pp. 70–91, July 1999.
  - [130] R. Szeliski, “Video mosaics for virtual environments,” *IEEE Computer Graphics and Applications*, vol. 16, no. 2, pp. 22–30, March 1996.
  - [131] F. Baader, S. Brandt, and C. Lutz, “Pushing the envelope,” in *Proceedings of the 19th Joint International Conference on Artificial Intelligence (IJCAI 2005)*, 2005.
  - [132] D. Calvanese, G. D. Giacomo, D. Lembo, M. Lenzerini, and R. Rosati, “Tailoring owl for data intensive ontologies,” in *Proceedings of the 1st OWL: Experiences and Directions Workshop*, 2005.
  - [133] G. Acampora and V. Loia, “Fuzzy control interoperability and scalability for adaptive domotic framework,” *IEEE Transactions on Industrial Informatics*, vol. 1, no. 2, pp. 97–111, 2005.
  - [134] G. Acampora and V. Loia, “Enhancing the fml vision for the design of open ambient intelligence environment,” in *Proceedings of 2005 IEEE International Conference on Systems, Man and Cybernetics*, 2005, vol. 3, pp. 2578–2583.
  - [135] G. Acampore, M. Di Meglio, and V. Loia, “An integrated development environment for transparent fuzzy agents design: an application to automotive electronic stability program,” in *Proceedings of IEEE World Congress on Computational Intelligence (WCCI 2008)*.