

BAYESIAN LEARNING OF CONCEPT ONTOLOGY FOR AUTOMATIC IMAGE ANNOTATION

RUI SHI

*(MSC. Institute of Computing Technology,
Chinese Academy of Sciences, Beijing, China)*

**A THESIS SUBMITTED
FOR THE DEGREE OF DOCTOR OF PHILOSOPHY
IN COMPUTER SCIENCE
SCHOOL OF COMPUTING
NATIONAL UNIVERSITY OF SINGAPORE**

2007

Acknowledgements

I would like to express my heartfelt gratitude to my supervisors, Prof. Tat-Seng Chua and Prof. Chin-Hui Lee, for providing the invaluable advice and constructive criticism, and for giving me freedom to explore the interesting research areas during my PhD study. Without their guidance and inspiration, my work in the past six years would not be so much fruitful. I am really grateful too for their enduring patience and support to me when I got frustrated at times or encountered difficult obstacles in the course of my research work. Their technical and editorial advice contributed a major part to the successful completion of this dissertation. Most importantly, they gave me the opportunity to work on the topic of automatic image annotation and to find my own way as a real researcher. I am extremely grateful for all of this.

I also would like to extend my gratitude to the other members of my thesis advisory committee, Prof. Mohan S Kankanhalli, Prof. Wee-Kheng Leow and Dr. Terence Sim, for their beneficial discussions during my Qualifying and Thesis Proposal examinations.

Moreover, I wish to acknowledge my other fellow Ph.D. students, colleagues and friends who shared my academic life in various occasions in the multimedia group of Prof. Tat-Seng Chua, Dr. Sheng Gao, Hui-Min Feng, Yun-Long Zhao, Shi-Ren Ye, Ji-Hua Wang, Hua-Xin Xu, Hang Cui, Ming Zhao, Gang Wang, Shi-Yong Neo, Long Qiu, Ren-Xu Sun, Jing Xiao, and many others. I have had enjoyable and memorable time with them in the past six years, without them my graduate school experience

would not be as pleasant and colorful.

Last but not least, I would like to express my deepest gratitude and love to my family, especially my parents, for their support, encouragement, understanding and love during many years of my studies.

Life is a journey. It is with all the care and support from my loved ones that has allowed me to scale on to greater heights.

Abstract

Automatic image annotation (AIA) has been a hot research topic in recent years since it can be used to support concept-based image retrieval. In the field of AIA, characterizing image concepts by mixture models is one of the most effective techniques. However, mixture models also pose some potential problems arising from the limited size of (even a small size of) labeled training images, when large-scale models are needed to cover the wide variations in image samples. These potential problems could be the mismatches between training and testing sets, and inaccurate estimations of model parameters.

In this dissertation, we adopted multinomial mixture model as our baseline and proposed a Bayesian learning framework to alleviate these potential problems for effective training from three different perspectives. (a) We proposed a Bayesian hierarchical multinomial mixture model (BHMMM) to enhance the maximum-likelihood estimations of model parameters in our baseline by incorporating prior knowledge of concept ontology. (b) We extended conventional AIA by three modes which are based on visual features, text features, and the combination of visual and text features, to effectively expand the original image annotations and acquire more training samples for each concept class. By utilizing the text and visual features from the training set and ontology information from prior knowledge, we proposed a text-based Bayesian model (TBM) by extending BHMMM to text modality, and a text-visual Bayesian hierarchical multinomial mixture model

(TVBM) to perform the annotation expansions. (c) We extended our proposed TVBM to annotate web images, and filter out low-quality annotations by applying the likelihood measure (LM) as a confidence measure to check the ‘goodness’ of additional web images for a concept class.

From the experimental results based on the 263 concepts of Corel dataset, we could draw the following conclusions. (a) Our proposed BHMMM can achieve a maximum F_1 measure of 0.169, which outperforms our baseline model and the other state-of-the-art AIA models under the same experimental settings. (b) Our proposed extended AIA models can effectively expand the original annotations. In particular, by combining the additional training samples obtained from TVBM and re-estimating the parameters of our proposed BHMMM, the performance of F_1 measure can be significantly improved from 0.169 to 0.230 on the 263 concepts of Corel dataset. (c) The inclusion of web images as additional training samples obtained with LM gives a significant improvement over the results obtained with the fixed top percentage strategy and without using additional web images. In particular, by incorporating the newly acquired image samples from the internal dataset and the external dataset from the web into the existing training set, we achieved the best per-concept precision of 0.248 and per-concept recall of 0.458. This result is far superior to those of state-of-the-arts AIA models.

Contents

1	Introduction	1
1.1	Background.....	1
1.2	Automatic Image Annotation (AIA).....	3
1.3	Motivation.....	5
1.4	Contributions.....	5
1.5	Thesis Overview.....	9
2	Literature Review	11
2.1	A General AIA Framework.....	11
2.2	Image Feature Extraction.....	12
2.2.1	Color.....	12
2.2.2	Texture.....	14
2.2.3	Shape.....	15
2.3	Image Content Decomposition.....	15
2.4	Image Content Representation.....	17
2.5	Association Modeling.....	18
2.5.1	Statistical Learning.....	18
2.5.2	Formulation.....	20
2.5.3	Performance Measurement.....	22
2.6	Overview of Existing AIA Models.....	23
2.6.1	Joint Probability-Based Models.....	24
2.6.2	Classification-Based Models.....	25
2.7.3	Comparison of Performance.....	28
2.7	Challenges.....	29
3	Finite Mixture Models	31
3.1	Introduction.....	31

3.1.1	Gaussian Mixture Model (GMM).....	32
3.1.2	Multinomial Mixture Model (MMM).....	33
3.2	Maximum Likelihood Estimation (MLE).....	35
3.3	EM algorithm.....	36
3.4	Parameter Estimation with the EM algorithm.....	38
3.5	Baseline Model.....	40
3.6	Experiments and Discussions.....	41
3.7	Summary.....	43
4	Bayesian Hierarchical Multinomial Mixture Model	44
4.1	Problem Statement.....	44
4.2	Bayesian Estimation.....	46
4.3	Definition of Prior Density.....	48
4.4	Specifying Hyperparameters Based on Concept Hierarchy.....	49
4.4.1	Two-Level Concept Hierarchy.....	51
4.4.2	WordNet.....	52
4.4.3	Multi-Level Concept Hierarchy.....	53
4.4.4	Specifying Hyperparameters.....	54
4.5	MAP Estimation.....	55
4.6	Exploring Multi-Level Concept Hierarchy.....	59
4.7	Experiments and Discussions.....	60
4.7.1	Baseline vs. BHMMM.....	60
4.7.2	State-of-the-Art AIA models vs. BHMMM.....	62
4.7.3	Performance Evaluation with Small Set of Samples.....	63
4.8	Summary.....	64
5	Extended AIA Based on Multimodal Features	66
5.1	Motivation.....	66
5.2	Extended AIA.....	67
5.3	Visual-AIA Models.....	70

5.3.1 Experiments and Discussions.....	71
5.4 Text-AIA Models.....	72
5.4.1 Text Mixture Model (TMM).....	72
5.4.2 Parameter Estimation for TMM.....	73
5.4.3 Text-based Bayesian Model (TBM).....	75
5.4.4 Parameter Estimation for TBM.....	78
5.4.5 Experiments and Discussions.....	79
5.5 Text-Visual-AIA Models.....	83
5.5.1 Linear Fusion Model (LFM).....	83
5.5.2 Text and Visual-based Bayesian Model (TVBM).....	85
5.5.3 Parameter Estimation for TVBM.....	87
5.5.4 Experiments and Discussions.....	89
5.6 Summary.....	91
6 Annotating and Filtering Web Images	92
6.1 Introduction.....	92
6.2 Extracting Text Descriptions.....	93
6.3 Fusion Models.....	94
6.4 Annotation Filtering Strategy.....	95
6.4.1 Top N_P	96
6.4.2 Likelihood Measure (LM).....	97
6.5 Experiments and Discussions.....	100
6.5.1 Crawling Web Images.....	100
6.5.2 Pipeline.....	101
6.5.3 Experimental Results Using Top N_P	102
6.5.4 Experimental Results Using LM.....	103
6.5.5 Refinement of Web Image Search Results.....	104
6.5.6 Top N_P vs. LM.....	105
6.5.7 Overall Performance.....	108
6.6 Summary.....	108

7	Conclusions and Future Work	110
7.1	Conclusions.....	110
7.1.1	Bayesian Hierarchical Multinomial Mixture Model.....	111
7.1.2	Extended AIA Based on Multimodal Features.....	111
7.1.3	Likelihood Measure for Web Image Annotation.....	112
7.2	Future Work.....	113
	Bibliography	117

List of Tables

2.1	Published results of state-of-the-art AIA models.....	29
2.2	The average number of training images for each class of CMRM.....	30
3.1	Performance comparison of a few representative state-of-the-art AIA models and our baseline.....	41
4.1	Performance summary of baseline and BHMMM.....	61
4.2	Performance comparison of state-of-the-art AIA models and BHMMM.....	62
4.3	Performance summary of baseline and BHMMM on the concept classes with small number of training samples.....	63
5.1	Performance of BHMMM and visual-AIA.....	71
5.2	Performance comparison of TMM and TBM for text-AIA.....	80
5.3	Performance summary of TMM and TBM on the concept classes with small number of training samples.....	83
5.4	Performance comparison of LFM and TVBM for text-visual-AIA.....	90
5.5	Performance summary of LFM and TVBM on the concept classes with small number of training samples.....	90
6.1	Performance of TVBM and Top N_P Strategy.....	102
6.2	Performance of LM with different thresholds.....	103
6.3	Performance comparison of top N_P and LM for refining the retrieved web images.....	104
6.4	Performance comparison of top N_P and LM in Group I.....	105

6.5	Performance comparison of top N_P and LM in Group II.....	107
6.6	Overall performance.....	108

List of Figures

2.1	A general system framework for AIA.....	11
2.2	Three kinds of image components.....	16
2.3	An illustration of region tokens.....	17
2.4	The paradigm of supervised learning.....	19
3.1	An example of image representation in this dissertation.....	34
4.1	An example of potential difficulty for ML estimation.....	45
4.2	The principles of MLE and Bayesian estimation.....	46
4.3	The examples of concept hierarchy.....	50
4.4	Training image samples for the concept class of ‘grizzly’.....	51
4.5	Two level concept hierarchy.....	52
4.6	An illustration of specifying hyperparameters.....	54
5.1	Two image examples with incomplete annotations.....	67
5.2	The proposed framework of extended AIA.....	69
5.3	Four training images and their annotations for the class of ‘dock’.....	75
5.4	An illustration of TBM.....	78
5.5	Examples of top additional training samples obtained from both TMM and TBM.....	81
5.6	Examples of top additional training samples obtained from TBM.....	82
5.7	An illustration of the dependency between visual and text modalities.....	85
5.8	An illustration of structure of the proposed text-visual Bayesian model.....	86

6.1	Likelihood measure.....	99
6.2	Some negative additional samples obtained from top N_P	106
6.3	Some positive additional samples obtained from LM.....	107

Chapter 1

Introduction

Recent advances in digital signal processing, consumer electronics technologies and storage devices have facilitated the creation of very large image/video databases, and made available a huge amount of image/video information to a rapidly increasing population of internet users. For example, it is now easy for us to store 120GB of an entire year of ABC news at 2.4GB per show or 5GB of a five-year personal album (e.g. at an estimated 2,000 photos per year for 5 years at the size of about 0.5M for each photo) in our computer. Meanwhile, with the wide spread use of internet, many users are putting a large amount of images/videos online, and more and more media content providers are delivering live or on-demand image/videos over the internet. This explosion of rich information also poses challenging problems of browsing, indexing or searching multimedia contents because of the data size and complexity. Thus there is a growing demand for new techniques that are able to efficiently process, model and manage image/video contents.

1.1 Background

Since the early 1970's, lots of research studies have been done to tackle the abovementioned problems, with the main thrust coming from the information retrieval (IR) and computer vision communities. These two groups of researchers approach these problems from two different perspectives (Smith et al. 2003). One is query-by-keyword

(QBK), which essentially retrieves and indexes images/videos based on their corresponding text annotations. The other paradigm is query-by-example (QBE), in which an image or a video is used to present a query.

One popular framework of QBK is to annotate and index the images by keywords and then employ the text-based information retrieval techniques to search or retrieve the images (Chang and Fu 1980; Chang and Hsu 1992). Some advantages of QBK approaches are their ease of use and are readily accepted by ordinary users because human thinks in terms of semantics. Yet there exist two major difficulties, especially when the size of image collection is large (in tens or hundreds of thousands). One such difficulty in QBK is the rich contents in images and subjectivity of human perception. It often leads to mismatches in the process of later retrieval due to the different semantic interpretations for the same image between the users and the annotators. The other difficulty is due to the vast amount of laboring efforts required in manually annotating images for effective QBK. As the size of the image/video collection is large, in the order of 10^4 - 10^7 or higher, manually annotating or labeling such a large collection is tedious, time consuming and error prone. Thus in the early 1990's, because of the emergence of large-scale image collections, the two difficulties faced by manual annotation approaches become more and more acute.

To overcome these difficulties, QBE approaches were proposed to support content-based image retrieval (CBIR) (Rui et al. 1999). QBIC (Flickner et al. 1995) and Photobook (Pentland et al. 1996) are two of the representative CBIR systems. Instead of using manually annotated keywords as the basis of indexing and retrieving images, almost all QBE systems use visual features such as color, texture and shape to retrieve

and index the images. However, these low-level visual features are inadequate to model the semantic contents of images. Moreover, it is difficult to formulate precise queries using visual features or image examples. As a result, QBE is not well-accepted by ordinary users.

1.2 Automatic Image Annotation (AIA)

In recent years, automatic image annotation (AIA) has become an emerging research topic aiming at reducing human labeling efforts for large-scale image collections. AIA refers to the process of automatically labeling the images with a predefined set of keywords or concepts representing image semantics. The aim of AIA is to build associations between image visual contents and concepts.

As pointed out in (Chang 2002), content-based media analysis and automatic annotation are important research areas that have captured much interest in recognizing the need to provide semantic-level interaction between users and contents. However, AIA is challenging for two key reasons:

1. There exists a “semantic gap” between the visual features and the richness of human information perception. This means that lower level features are easily measured and computed, but they are far away from a direct human interpretation of image contents. So a paramount challenge in image and video retrieval is to bridge the semantic gap (Sebe et al. 2003). Furthermore, as mentioned in (Eakins and Graham 2002), human semantics also involve understanding the intellectual, subjective, emotional and religious sides of the human, which could be described

only by the abstract concepts. Thus it is very difficult to make the link between image visual contents and the abstract concepts required to describe the image. Enser and Sandom (2003) presented a comprehensive survey of the semantic gap issues in visual information retrieval and provided a better-informed view on the nature of semantic information need from their study.

2. There is always a limited set of (even a small set of) labeled training images. To bridge the gap between low-level visual features and high-level semantics, statistical learning approaches have recently been adopted to associate the visual image representations and semantic concepts. They have been demonstrated to effectively perform the AIA task (Duygulu et al. 2002; Jeon et al. 2003; Srikanth et al. 2005; Feng et al. 2004; Carneiro et al. 2007). Compared with the other reputed AIA models, mixture model is the most effective and has been shown to achieve the best AIA performance on the Corel dataset (Carneiro et al. 2007). However, the performance of such statistical learning approaches is still low, since they often need large amounts of labeled samples for effective training. For example, the approaches of mixture model often need many mixtures to cover the large variations in image samples, and we need to collect a large amount of labeled samples to estimate the mixture parameters. But it is not a practical way to manually label a sufficiently large number of images for training. Thus this problem has motivated our research to explore the mixture models to perform effective AIA based on a limited set of (even a small set of) labeled training images.

Throughout this thesis, we loosely use the term *keyword* and *concept* interchangeably to denote text annotations of images.

1.3 Motivation

The potential difficulties resulting from a limited set of (even a small set of) training samples could be the mismatches between training and testing sets or inaccurate estimation of model parameters. These difficulties are even more serious for a large-scale mixture model. It is therefore important to develop novel AIA models which can achieve effective training with the limited set of labeled training images, especially with the small set of labeled training images. As far as we know, few research work in the AIA field have been conducted for tackling these potential difficulties, and we will discuss this topic in detail in the followed chapters.

1.4 Contributions

In this dissertation, we propose a Bayesian learning framework to automatically annotate images based on a predefined list of concepts. In our proposed framework, we circumvent abovementioned problems from three different perspectives: 1) incorporating prior knowledge of concept ontology to improve the commonly used maximum-likelihood (ML) estimation of mixture model parameters; 2) effectively expanding the original annotations of training images based on multimodal features to acquire more training samples without collecting new images; and 3) resorting to open image sources

on the web for acquiring new additional training images. In our framework, we use multinomial mixture model (MMM) with maximum-likelihood (ML) estimation as our baseline, and our proposed approaches are as follows:

- *Bayesian Hierarchical Multinomial Mixture Model (BHMMM)*. In this approach, we enhance the ML estimation of the baseline model parameters by imposing a maximum a posterior (MAP) estimation criterion, which facilitates a statistical combination of the likelihood functions of available training data and the prior density with a set of parameters (often referred to as *hyperparameters*). Based on such a formulation, we need to address some key issues, namely: (a) the definition of the prior density; (b) the specification of the hyperparameters; and (c) the MAP estimation of the mixture model parameters. To tackle the first issue, we define the Dirichlet density as a prior density, which is conjugate to multinomial distribution and makes it easy to estimate the mixture parameters. To address the second issue, we first derive a multi-level concept hierarchy from WordNet to capture the concept dependencies. Then we assume that all the mixture parameters from the sibling concept classes share a common prior density with the same set of hyperparameters. This assumption is reasonable since given a concept, say, ‘oahu’, the images from its sibling concepts (say, ‘kauai’ and ‘maui’) often share the similar context (the natural scene on tropical island). We call such similar context information among sibling concepts as the ‘shared knowledge’. Thus the hyperparameters are used to simulate the shared knowledge, and estimated by empirical Bayesian approaches with an MLE criterion. Given the

defined prior density and the estimated hyperparameters, we tackle the third issue by employing an EM algorithm to estimate the parameters of multinomial mixture model.

- *Extended AIA Based on Multimodal Features.* Here we alleviate the potential difficulties by effectively expanding the original annotations of training images, since most image collections often come with only a few and incomplete annotations. An advantage of such an approach is that we can augment the training set of each concept class without the need of extra human labeling efforts or collecting additional training images from other data sources. Obviously two groups of information (text and visual features) are available for a given training image. Thus we extend the conventional AIA to three modes, namely associating concepts to images represented by visual features, briefly called as visual-AIA, by text features as text-AIA, and by both text and visual features as text-visual-AIA. There are two key issues related to fusing text and visual features to effectively expand the annotations and acquire more training samples: (a) accurate parameter estimation especially when the number of training samples is small; and (b) dependency between visual and text features. To tackle the first issue, we simply extend our proposed BHMMM to visual and text modalities as visual-AIA and text-AIA, respectively. To tackle the second issue, we propose a text-visual Bayesian hierarchical multinomial mixture model (TVBM) as text-visual-AIA to capture the dependency between text and visual mixtures in order to perform effective expansion of annotations.

- *Likelihood Measure for Web Image Annotation.* Nowadays, images have become widely available on the World Wide Web (WWW). Different from the traditional image collections where very little information is provided, the web images tend to contain a lot of contextual information like surrounding text and links. Thus we want to annotate web images to collect additional samples for training. However, due to large variations among web images, we need to find an effective strategy to measure the ‘goodness’ of additional annotations for web images. Hence we first apply our proposed TVBM to annotate web images by fusing the text and visual features derived from the web pages. Then, given the likelihoods of web images from TVBM, we investigate two different strategies to examine the ‘goodness’ of additional annotations for web images, i.e. top N_P strategy and likelihood measure (LM). Compared with setting a fixed *percentage* by the top N_P strategy for all the concept classes, LM can set an adaptive threshold for each concept class as a confidence measure to select the additional web images in terms of the likelihood distributions of the training samples.

Based on our proposed Bayesian learning framework which aims to alleviate the potential difficulties resulting from the limited set of training samples, we summarize our contributions as follows:

1. *Bayesian Hierarchical Multinomial Mixture Model (BHMMM)*

We incorporate prior knowledge into the hierarchical concept ontology, and propose a Bayesian learning model called BHMMM (Bayesian Hierarchical Multinomial Mixture Model) to characterize the concept ontology structure and estimate the parameters of concept mixture models with the EM algorithm. By

using concept ontology, our proposed BHMMM performs better than our baseline mixture model (MMM) by 44% in term of F_1 measure.

2. Extended AIA Based on Multimodal Features

We extend conventional AIA by three modes (visual-AIA, text-AIA and text-visual-AIA) to effectively expand the annotations and acquire more training samples for each concept class. By utilizing the text and visual features from training set and ontology information from prior knowledge, we propose a text-based Bayesian model (TBM) as text-AIA by extending BHMMM to text modality, and a text-visual Bayesian hierarchical multinomial mixture model (TVBM) as text-visual-AIA. Compared with BHMMM, TVBM achieves the 36% improvement in terms of F_1 measure.

3. Likelihood Measure for Web Image Annotation

We extend our proposed TVBM to annotate the web images and filter out the low-quality annotations by applying the likelihood measure (LM) as a confidence measure to examine the ‘goodness’ of additional web images. By incorporating the newly acquired web image samples into the expanded training set by TVBM, we perform best in terms of per-concept precision of 0.248 and per-concept recall of 0.458 as compared to other state-of-the-art AIA models.

1.5 Thesis Overview

The rest of this thesis is organized as follows:

Chapter 2 discusses the basic questions and reviews the-state-of-art research on automatic image annotation. We also discuss the challenges for the current research work on AIA.

Chapter 3 reviews the fundamentals on finite mixture model, including Gaussian mixture model, multinomial mixture model and estimation of model parameters with EM algorithm based on an MLE criterion. Meanwhile, we discuss the details of our baseline model (Multinomial Mixture Model) for AIA.

Chapter 4 presents the fundamentals of Bayesian learning of multinomial mixture model, including the formulation of posterior probability, the definition of the prior density, the specification of the hyperparameters and an MAP criterion for estimating model parameters. We propose a Bayesian hierarchical multinomial mixture model (BHMMM), and discuss how to apply Bayesian learning approaches to estimate the model parameters by incorporating hierarchical prior knowledge of concepts.

In Chapter 5, without collecting new additional training images, we discuss the problem of effectively increasing the training set for concept classes by utilizing visual and text information of the training set. We then present three extended AIA models, i.e. visual-AIA, text-AIA and text-visual-AIA models, which are based on the visual features, text features and the combination of text and visual features, respectively.

In Chapter 6, we apply our proposed TVBM which is one of text-visual-AIA models to annotate new images collected from the web, and investigate two strategies of Top N_P and LM (Likelihood Measure) to filter out the low-quality additional images for a concept class by checking the ‘goodness’ of concept annotations for web images.

In Chapter 7, we present our concluding remarks, summarize our contributions and discuss future research directions.

Chapter 2

Literature Review

This Chapter introduces a general AIA framework, and then discusses each module in this framework, including image visual feature extraction, image content decomposition and representation, and the association modeling between image contents and concepts. In particular, we categorize the existing AIA models into two groups, namely the joint probability-based and classification-based models, and discuss and compare the models in both groups. Finally we present the challenges for the current AIA work.

2.1 A General AIA Framework

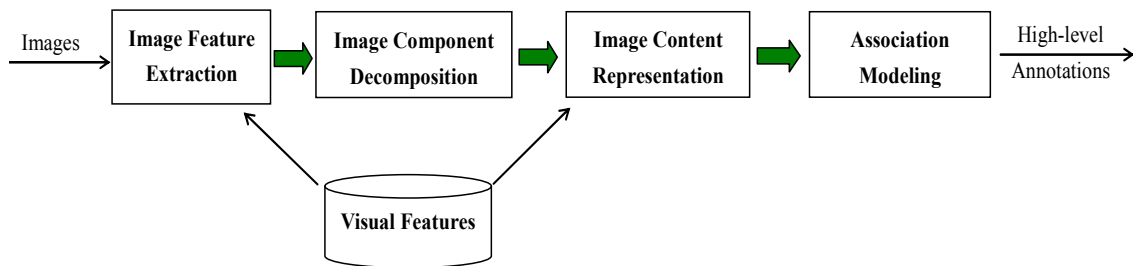


Figure 2.1: A general system framework for AIA

Most current AIA systems are composed of four key modules: image feature extraction, image component decomposition, image content representation, and association learning. A general framework of AIA is shown in Figure 2.1. The feature extraction module analyzes images to obtain low-level features, such as color and texture. The module of

image component decomposition decomposes an image into a collection of sub-units, which could be segmented regions, equal-size blocks, or an entire image, etc. Such image components are used as a basis for image representation and analysis. The image content representation module models each content unit based on a feature representation scheme. The visual features used for image content representation could be different from those for image component decomposition. The module of association modeling computes the associations between image content representations and textual concepts and assigns appropriate high-level concepts to image.

2.2 Image Feature Extraction

Features are “the measurements which represent the data” (Minka 2005). Features not only influence the choice of subsequent decision mechanisms, their quality is also crucial to the performance of learning systems as a whole. For any image database, a feature vector, which describes the various visual cues, such as shape, texture or color, is computed for each image in the database. Nowadays, almost all AIA systems use color, shape and texture features to model image contents. In this Section, we briefly review the color-, shape-, and texture-based image features.

2.2.1 Color

Color is a dominant visual feature and widely used in all kinds of image and video processing/retrieval systems. A suitable color space should be uniform, complete, compact and natural. Digital images are normally represented in RGB color space used

by CRTs. However, RGB color space is perceptual non-uniform, i.e., it does not model the human perception of color. To overcome this problem, some linear color spaces, such as LUV, LAB, HSV, YCrCb color spaces, have been developed to best matches user's ability to perceive and differentiate colors in natural images (Hall 1989; Chua et al. 1998, 1999; Carson et al. 1999, 2002; Furht 1998; Manjunath et al. 2001). A comparison of color features and color spaces suitable for image indexing and retrieval can be found in (Furht 1998). Furht reported that while no single color feature or color space was best, the use of color moment and color histogram features in the LUV and HSV color spaces yielded better retrieval results than in the RGB color space.

Color features can be categorized as global or local ones depending on the range of spatial information used. Global color features capture the global distribution or statistics of colored pixels, such as the color histogram which computes the distribution of pixels in quantized color space (Hafner 1995), or the color moments which compute the moment statistics in each color channel (Stricker and Orengo 1995). Color histogram is generally invariant to translation and rotation of the images, and the normalized color histogram leads to scale invariance.

However, color histogram can not capture any local information, and thus images with very different image appearances can have similar histogram (Hsu et al. 1995). To overcome this problem, new representations have been developed to incorporate spatial distributions of colors in (Chua et al. 1997; Vailaya et al. 1999). Examples include color coherence vector (CCV) (Pass et al. 1996), color region model (Smith and Chang 1996), color pair model (Chua et al. 1994) and the color correlogram (Huang et al. 1997). These features have been demonstrated to be effective in color image classification and retrieval

(Smith 1997, Tong and Chang 2001), object matching and detection under controlled conditions (Fergus et al. 2003; Lowe 2004).

2.2.2 Texture

Variations of image intensities that form certain repeated patterns are called visual texture (Tuceryan and Jain 1993). These Patterns can be the result of physical properties of the object surface (i.e. roughness and smoothness), or the result of reflectance differences such as the color on a surface. Human can easily recognize a texture, yet it is very difficult to define it. Most natural surfaces exhibit texture and it may be useful to extract texture features for querying. For example, images of wood and fabric can be easily classified based on the texture rather than shape or color.

Tuceryan and Jain (1993) identified four major categories of features for texture identification: statistical (Jain et al. 1995), geometrical (Tuceryan and Jain 1990), model-based (Besag 1974; Pentland 1984; Mao and Jain 1992) and signal processing features (Coggins and Jain 1985; Jain and Farrokhnia 1991; Manjunath and Ma 1997). In particular, signal processing features, such as DCT, wavelets and Gabor filters, have been used effectively for texture analysis in many retrieval systems (Picard and Minka 1995; Manjunath and Ma 1997; Wang and Li 2002). The main advantage of signal processing features is that they can characterize the local properties of an image very well in different frequency bands. However, there are often a lot of different local properties that need to be characterized for images, such as clouds and buildings. In order to facilitate adaptive image representation, an adaptive MP texture feature and a feature extraction algorithm are proposed in (Shi et al. 2004) by borrowing the concept from matching

pursuit (Mallat 1993; Bergeaud and Mallat 1995) and using the different properties of some signal processing textures to represent image details.

2.2.3 Shape

Shape is a concept which is widely understood yet difficult to define formally. Therefore, at least yet, there exists no uniform theory of shape. Usually the techniques of shape descriptions can be categorized as boundary- or region-based methods depending on whether the boundary or the area inside the boundary is coded (Marshall 1989; Mehtre et al. 1997). The boundary-based features include histogram of edge directions, chord distribution, aspect ratio, boundary length and so on. The region-based features include Zernike moments, area, eccentricity, elongatedness, direction and so on. A good survey of shape features is presented in (Brandt 1999).

Since AIA is a general task and not for a specific domain, a major limitation of using shape model is that the shape features are often unreliable and easily affected by noise. Thus only color and texture features are normally employed to model and represent the image contents in most existing AIA models.

2.3 Image Content Decomposition

As discussed in Section 2.2, image component decomposition aims to decompose the image into some meaningful units for image analysis. As shown in Figure 2.2, three kinds of image components, entire image, segmented regions and equal-size blocks, are often

used as image analysis units in most content-based image retrieval and automatic image annotation systems,

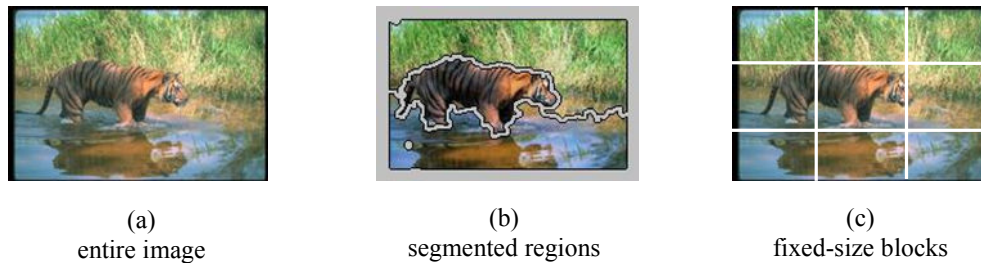


Figure 2.2: Three kinds of image components

The entire image was used as a unit in (Swain 1991; Manjunath and Ma 1996), and only global features were used to represent images. However, such systems are usually not effective since only global features cannot capture the local properties of an image well. Thus some recent systems use segmented regions as sub-units in images (Deng et al. 1999; Deng and Manjunath 2001; Carson et al. 1999, 2002). Many techniques have been reported in the literature for image segmentation (Jain and Farrokhnia 1991; Manjunath and Ma 1997; Morris et al. 1997; Carson et al. 1999). However, segmenting images into meaningful units is a very difficult task, and the accuracy of segmentation is still an open problem. As a compromise, several systems adopt fixed-size sub-image blocks as sub-units for an image (Szummer 1998; Mori et al. 2000; Feng et al. 2004). The main advantage is that fixed-size block-based methods can be implemented easily. In order to compensate potential drawbacks of block-based methods, hierarchical multi-resolution structure is employed in (Wang and Li 2002). Intuitively the retrieval or annotation performance based on the segmented regions should be better than those based on fixed-sized blocks, but there is no definite conclusion on which one is better. Generally

speaking, most existing AIA models employ segmented regions or fixed-size blocks as the image analysis units.

2.4 Image Content Representation

Image content representation aims to model each content unit based on a feature representation scheme. The visual features used for image content representation could be different from those for image component decomposition (Carson et al. 2002; Shi et al. 2004). For example, some global features, such as average of LUV color components and DCT textures, are used for image segmentation in (Shi et al. 2004), since the segmentation based on global features can achieve good object-level results. But some local features, such as LUV histogram and adaptive matching pursuit (MP) textures (Shi et al. 2004), are used for content representation by combining with the global features, since these local features can characterize the local properties of image segmentations very well.



Figure 2.3: An illustration of region tokens (Jeon et al. 2003)

Another popular method to represent image content is based on region tokens (Mori et al. 2000; Duygulu et al. 2002; Jeon et al. 2003; Shi et al. 2006, 2007). In such methods all the images are first segmented into regions, and each region is described by some set of visual features. Then all the regions are clustered into some region clusters which are so-called ‘region tokens’ represented by the centroids of region clusters. Thus given an image with a set of segmented regions, each segmented region is assigned to a unique region token whose centroid is closest to the given segmented region. The main advantage of such methods is that we can construct a limited size of region token vocabulary to cover all the image variations in the space of visual features. Thus we can give a simple representation for images based on such a vocabulary of region tokens.

2.5 Association Modeling

In the previous sub-sections, we have discussed how to decompose and represent image contents. In the following, we will focus on the module of association modeling which is the most important part of AIA models. This module aims to compute the associations between image content representations and high-level textual concepts.

2.5.1 Statistical Learning

“Nothing is more practical than a good theory” (Vapnik 1998). Statistical learning theory plays a central role in many areas of science, finance and industry. The main goal of statistical learning is to study the properties of learning algorithms, such as gaining knowledge, making predictions, making decisions or constructing models, from a set of

data in a statistical framework (Bousquet et al. 2004). As noted in (Vapnik 1995, 1998; Cherkassky and Mulier 1998), statistical learning theory gives a formal and precise definition of the basic concepts like learning, generalization, overfitting, and also characterizes the performance of learning algorithms. Thus such a theory may ultimately help to design better learning algorithms.

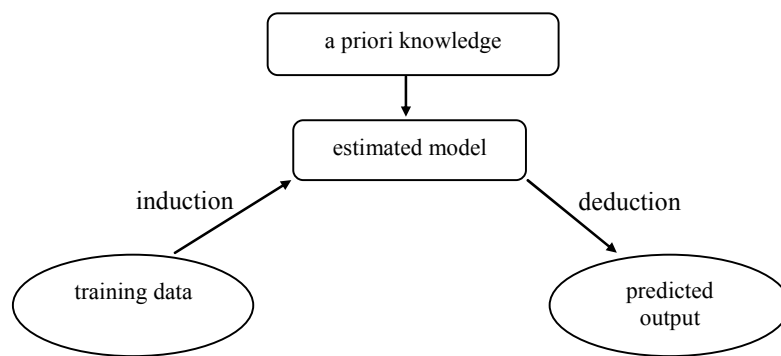


Figure 2.4: The paradigm of supervised learning (Vapnik 1995)

A majority of statistical learning scenarios generally follows the classical paradigm as shown in Figure 2.4, including two steps: induction (i.e. progressing from training data to a general or estimated model) and deduction (i.e. progressing from a general or estimated model to a particular case or some output values). A training sample consists of a pair of an input representing the sample (typically a feature vector) and a desired output describing a corresponding concept. The output of the function can be a continuous value, or can predict a class label of the input. The task of learning is to predict the value of the function for any valid input after having seen a set of training examples (i.e. pairs of input and target output). In the current AIA field, most existing models follow this

classical paradigm, so first we will give a general formulation for AIA, and then illustrate the paradigm in detail.

2.5.2 Formulation

Consider that we have a predefined concept or keyword vocabulary $\mathcal{C} = \{c_1, c_2, \dots, c_V\}$, of semantic labels, ($|\mathcal{C}| = V$), and a set of training images $\mathcal{I} = \{I_1, I_2, \dots, I_U\}$, ($|\mathcal{I}| = U$). Given an image $I_j \in \mathcal{I}$, $1 \leq j \leq U$, the goal of automatic image annotation is to extract the set of concepts or keywords from \mathcal{C} , $C_j = \{c_{j,1}, c_{j,2}, \dots, c_{j,k_j}\} \subseteq \mathcal{C}$, that best describes the semantics of I_j . In AIA, any training image is labeled with a set of concepts from \mathcal{C} , thus the learning is based on a training set, $\mathcal{D} = \{(I_j, C_j): 1 \leq j \leq U\}$, of image-annotation pairs.

We now define additional notations as follows. (1) We denote an input variable by symbol X , where X is usually a random vector of image representations, and I_j is the j^{th} observed value of X . (2) We denote an output variable W , which takes values in $\{1, \dots, V\}$, so that $W = i$ if and only if X is a sample from the concept $c_i \in \mathcal{C}$. Thus given the training set \mathcal{D} , we can use two ways for learning, namely the joint probability-based and classification-based models.

For the joint probability-based models, we assume that (X, W) is a pair of random variables represented by some joint probability distributions, $P_{X,W}(X, W)$. Then based on a set of observations $\mathcal{D} = \{(I_j, C_j): 1 \leq j \leq U\}$ of (X, W) , the goal of association learning is to infer the properties of this joint probability density. At the annotation stage, given an

image represented by a vector I , we obtain a function of \mathbf{W} to rank all concepts as shown in Eq. (2.1).

$$P_{\mathbf{W}|\mathbf{X}}(\mathbf{W} | I) = P_{\mathbf{X},\mathbf{W}}(\mathbf{X}, \mathbf{W}) / P_{\mathbf{X}}(I) \quad (2.1)$$

In the classification models, each label $c_i \in \mathcal{C}$ is taken as a semantic class, and then a set of class-conditional distributions or likelihood densities $P_{\mathbf{X}|\mathbf{W}}(\mathbf{X}|\mathbf{W} = i)$ are estimated for each concept class. As pointed out in the well-known statistical decision theory (Duda et al. 2001), it is not difficult to show that labeling at the annotation stage can be solved with a minimum probability of error if the posterior probabilities

$$P_{\mathbf{W}|\mathbf{X}}(\mathbf{W} = i | \mathbf{X}) = P_{\mathbf{X}|\mathbf{W}}(\mathbf{X} | \mathbf{W} = i) P_{\mathbf{W}}(\mathbf{W} = i) / P_{\mathbf{X}}(\mathbf{X}) \quad (2.2)$$

are available, where $P_{\mathbf{W}}(\mathbf{W} = i)$ is the prior probability of the i^{th} semantic concept class. In particular, given an image vector I for testing, the label that achieves a minimum probability of an error for that image is

$$\bar{i} = \arg \max_i P_{\mathbf{W}|\mathbf{X}}(\mathbf{W} = i | I) \quad (2.3)$$

In summary, in order to illustrate the classical paradigm of the learning process in AIA, we summarize both formulations as follows:

1. A set of training data $\mathcal{D} = \{(I_j, C_j): 1 \leq j \leq U\}$ for learning.
2. A prior knowledge used to impose constraints on the posterior or likelihood densities, $P_{\mathbf{X}}(\mathbf{X})$ or $P_{\mathbf{W}}(\mathbf{W})$. In AIA, $P_{\mathbf{X}}(\mathbf{X})$ and $P_{\mathbf{W}}(\mathbf{W})$ are often assumed to be uniform distributions.

3. A set of learning models needs to be estimated, $P_{X,W}(X, W)$ for joint probability-based models, and $P_{W|X}(W|X)$ or $P_{X|W}(X|W)$ for classification-based models.
4. An inductive principle, namely a general prescription for combining prior knowledge with available training data in order to produce an estimate of the learning model in Eq. (2.2).
5. A deduction principle, i.e. Eqs. (2.1) and (2.3).

Generally speaking, most existing AIA research work can be categorized into groups learning of either joint probability-based models or classification-based models. Before we review the existing work in Section 2.7, we will give a brief introduction on the performance measure used in the field of AIA.

2.5.3 Performance Measurement

Currently most AIA models adopt the common performance measures derived from information retrieval. Given some un-annotated images for testing, the AIA system will automatically generate a set of concept annotations for each image. Thus we can compute the recall, precision and F_1 of every concept in the testing set. Given a particular concept c , if there are $|c_g|$ images in ground truth labeled with this concept, while the AIA system annotates $|c_{auto}|$ images with concept c , where $|c_r|$ are correct, then we can compute the following measurements: $\text{recall} = |c_r| / |c_g|$, $\text{precision} = |c_r| / |c_{auto}|$, and $F_1 = 2 * \text{recall} * \text{precision} / (\text{recall} + \text{precision})$.

Based on the definition of performance measurements, the expected values for recall, precision and F_1 can be obtained if an algorithm randomly annotates an image. Here we

take recall measurement as an example to explain the best value of this metric. In our research work, we use the public CorelCD dataset containing 500 testing images for 263 concept classes to test our models. The average number of testing images for each concept class is 10 in the CorelCD dataset. Thus the expected value of recall can be calculated as follows:

$$E(R) = \sum_{i=1}^{10} r_i P(R = r_i) \quad (2.4)$$

where R is a random discrete variable for recall, r_i denotes the recall value, $r_1 = 0.1$, $r_2 = 0.2$, $r_3 = 0.3$, ..., $r_{10} = 1$. Here $P(R = r_i) = C_{10}^i C_{500-10}^{10-i} / C_{500}^{10}$ denotes the probability of taking the recall value as r_i , and C_m^n denotes the number of choices on randomly extracting n objects from m different objects. For example, $P(R = r_1) = C_{10}^1 C_{490}^9 / C_{500}^{10} \approx 0.02$. So the expected value of recall, $E(R)$, is less than 0.03.

2.6 Overview of Existing AIA Models

Next we will review existing AIA models by following the general formulation in Section 2.5.2. That is to say, most AIA models can be divided into two categories, namely the joint probability-based and classification-based models.

2.6.1 Joint Probability-Based Models

The first category of AIA models is based on learning the joint probability of concepts and image representations (Barnard 2001; Blei and Jordan 2003; Duygulu et al. 2002; Feng et al. 2004; Carbonetto et al. 2004; Lavrenko et al. 2003; Monay and Perez 2003, 2004). As discussed in Section 2.5.2, most approaches in this category focus on finding joint probabilities of images and concepts, $P_{\mathbf{X},\mathbf{W}}(\mathbf{X}, \mathbf{W})$. In these approaches a hidden variable L is introduced to encode the states of the world. Each of these states then defines a joint distribution for semantic concepts and image representations.

The various methods differ in the definition of the states of the hidden variable: some associate a state to each image in the database (Feng et al. 2003; Lavrenko et al. 2003), some associate them with image clusters (Barnard and Forsyth 2001; Duygulu et al. 2002), while others model high-level groupings by topic (Blei and Jordan 2003; Monay et al. 2003, 2004). The overall model is of the form:

$$P_{\mathbf{X}\mathbf{W}}(\mathbf{X}=\mathbf{I}, \mathbf{W}=i) = \sum_{s=1}^S P_{\mathbf{X}\mathbf{W}|S}(\mathbf{X}=\mathbf{I}, \mathbf{W}=i|s)P(s) \quad (2.5)$$

where S is the number of possible states, \mathbf{I} is the vector of image representation, and i denotes the i^{th} concept in the vocabulary \mathcal{C} . In order to avoid the difficulties of joint inference over the random variables on visual and text components, these two types of components are usually assumed to be independent given the state of the hidden variable.

$$P_{\mathbf{X}\mathbf{W}|S}(\mathbf{X}=\mathbf{I}, \mathbf{W}=i|s) = P_{\mathbf{X}|L}(\mathbf{X}=\mathbf{I}|s)P_{\mathbf{W}|L}(\mathbf{W}=i|s) \quad (2.6)$$

Since Eq. (2.4) is a form of mixtures, learning is usually based on the expected-maximization (EM) (Dempster et al. 1977) algorithm, with details depending on the definition of a hidden variable and the probability model adopted for $P_{X,W}(X, W)$. The simplest model in this family (Lavrenko 2003; Feng et al. 2004), which assumes each image in the training database as a state of the latent variable,

$$P_{X,W}(X=I, W=i) = \sum_{s=1}^{|D|} P_{X|S}(X=I|s)P_{W|S}(W=i|s)P(s) \quad (2.7)$$

where $|D|$ is the size of training set. This enables individual estimation of $P_{X|S}(X=I|s)$ $P_{W|S}(W=i|s)$ from each training image, as is common in the probabilistic literature (Smeulders et al. 2000; Vasconcelos et al. 1997, 2004), therefore eliminating the need to iterate the EM algorithm over the entire database (a procedure of significant computational complexity). At the annotation stage, Eq. (2.1) is used to rank all the annotation concepts. But as pointed out in (Carneiro and Vasconcelos 2007), there are some contradictions with this naïve assumption as shown in Eq. (2.5) because the annotation process is based on the Bayes decision rule which relies on the dependency between concepts and the vectors of image representations.

2.6.2 Classification-Based Models

In the second category of AIA models, each concept corresponds to a class, and AIA is formulated as a classification problem. The earliest efforts in the area of image classification were directed to the reliable extraction of specific semantics, e.g., differentiating indoor from outdoor scenes (Szummer and Picard 1998), cities from

landscapes (Vailaya et al. 1998), and detecting trees (Haering et al. 1997), horses (Forsyth and Fleck 1997), or buildings (Li and Shapiro. 2002), among others. These efforts posed semantics extraction as a binary classification problem. A set of training images with and without the concept of interest was collected, and then a binary classifier was trained to detect the concept in a one-vs-all mode (the concept of interest versus everything else). The classifier was then applied to all database images which were, in this way, annotated with respect to the presence or absence of the concept.

However, the one-vs-all training model in these efforts is not appropriate for AIA. There are several reasons. (a) Any images containing a concept c but not explicitly annotated with this concept are incorrectly taken as the negative samples. (b) In AIA, a training image is usually annotated by multiple concepts, thus a training image could be both positive and negative samples for a given concept. This is in conflict with the definition of binary classification. (c) If the size of concept vocabulary is large, the size of negative training samples for a given concept class is likely to be quite large, so the training complexity could be dominated by the complexity of negative learning.

Thus some approaches formulate AIA as a multi-class classification problem where each of the semantic concepts of interest defines an image class (Mori et al 2000; Carneiro and Vasconcelos 2007; Fan et al. 2005a, 2005b; Srikanth et al. 2005; Gao et al. 2006). At the annotation stage, these classes all directly compete for the image to annotate, which no longer faces a sequence of independent binary tests. Furthermore, by not requiring the modeling of the joint likelihood of concepts and image representations, the classification-based approaches do not require the independence assumptions usually associated with the joint probability-based models.

As shown in Eq. (2.2), there are two key issues for such approaches, namely: (a) how to define the likelihood density function, $P_{\mathbf{X}|\mathbf{W}}(\mathbf{X} | \mathbf{W})$; and (b) how to specify the parameters of the likelihood density function. Since we will focus on the likelihood function in the later chapters, we simply denote the likelihood density function as $p(\mathbf{X}|\Lambda_i)$, where i denotes the i^{th} concept class and Λ_i denotes the parameters of the likelihood density function for the i^{th} concept class. Most approaches in this area characterize the likelihood density typically by a mixture model, since the mixture model is an easy way to combine multiple simple distributions to form more complex ones and effectively cover the large variations in images. Thus given a total of J mixture components and the i^{th} concept class, the observed image vector I from this class is assumed to have the following probability:

$$p(I | \Lambda_i) = \sum_{j=1}^J \alpha_{i,j} p(I | \theta_{i,j}) \quad (2.8)$$

where $\Lambda_i = \{\alpha_{i,1}, \dots, \alpha_{i,J}, \theta_{i,1}, \dots, \theta_{i,J}\}$ is the parameter set for the above mixture model, including mixture weight set $\{\alpha_{i,j}\}_{j=1}^J$ ($\sum_{j=1}^J \alpha_{i,j} = 1$), and mixture parameter set $\{\theta_{i,j}\}_{j=1}^J$. $p(I | \theta_{i,j})$ is the j^{th} mixture component with the parameters $\theta_{i,j}$.

For example, Gaussian mixture model is employed in (Carneiro and Vasconcelos 2007; Fan et al. 2005a, 2005b) and the image is represented by a continuous feature vector. In (Carneiro and Vasconcelos 2007), they first estimated a single Gaussian distribution for each image in a concept class, and then organized the collection of single mixtures hierarchically to estimate the final mixture components for this concept class. In

(Fan et al. 2005a, 2005b), they focused on finding the optimal mixture structures for higher-level concept classes given a predefined concept hierarchy. The EM algorithm was used to estimate the parameters of mixture models in both approaches. Different from approaches in (Carneiro and Vasconcelos 2007; Fan et al. 2005a, 2005b), ontologies were used in (Srikanth et al. 2005) to build a hierarchical classification model with a concept hierarchy derived from WordNet (Miller et al. 1990) to model the concept dependencies. In this approach, they assumed a single multinomial distribution for each concept class, and an improved estimate for each leaf concept node was obtained by “shrinking” its estimate towards the ML estimates of all its ancestors tracing back from that leaf to the root of the concept hierarchy.

2.6.3 Comparison of Performance

To compare the performance of the state-of-the-art AIA models, we tabulate the published results in Table 2.1 based on the Corel dataset. The state-of-the-art AIA models include translation model (TM) (Duygulu et al. 2002), cross-media relevance model (CMRM) (Jeon et al. 2003), hierarchical classification approach (HC) (Srikanth et al. 2005), multiple Bernoulli relevance model (MBRM) (Feng et al. 2004), and mixture hierarchy approach (MH) (Carneiro and Vasconcelos 2007). In this comparison, TM, CMRM and HC share the same experimental settings based on region tokens, while MBRM and MH share the same experimental settings based on continuous visual feature representations.

Table 2.1: Published results of state-of-the-art AIA models

Models	TM	CMRM	HC	MBRM	MH
# of concepts (recall>0)	49	66	93	122	137
Mean Per-concept metrics on all 263 concepts on the Corel dataset					
Mean Precision	0.040	0.090	0.100	0.240	0.230
Mean Recall	0.060	0.100	0.176	0.250	0.290

As shown in Table 2.1, we can draw the following observations: (a) in terms of the performance measurements of mean precision and recall, classification-based approaches are more effective than joint probability-based approaches in AIA, since the performance of MH is better than that of MBRM, and HC is better than that of TM and CMRM; and (b) mixture model is effective in covering the image variations for AIA, since MH, CMRM and MBRM can be viewed as a kind of mixture model.. Thus in our work we also formulate AIA as a multi-class classification problem and adopt the mixture model as our baseline. In Chapter 3, we will present the details of mixture model and our baseline model.

2.7 Challenges

As we discussed in the previous sections, we are mainly relying on statistical learning approaches to build AIA models. But as pointed out in (Vapnik 1995, 1998; Cherkassky and Mulier 1998), such statistical learning approaches often need a large amount of labeled images for effective training. In terms of the published results of CMRM for each concept class, we tabulate in Table 2.2 the average number of training images in two categories: in terms of concept class with zero recall vs. those with recall greater than

zero. As shown in Table 2.2, the average number of training images for concept classes with non-zero recall values is much larger than that of concept classes with zero recall.

Table 2.2: The average number of training images for each class of CMRM

	CMRM (recall>0)	CMRM (recall=0)
# of concept classes in each category	66	197
Average number of training images for each concept class	164	23

However, it is well known that labeling large amounts of training data for statistical learning is tedious and time-consuming, especially for multimedia data. Compared with the large variations of visual contents, we often have a limited set, or even a small set, of labeled training data. This could result in some potential difficulties, such as the mismatch between training set and testing set, and inaccurate parameter estimations. In particular, these potential difficulties could be more serious when a large-scale mixture model is employed to cover the large image variations, which often leads to poor AIA performance as our baseline shows in Chapter 3. It is therefore important to develop novel AIA models which can achieve effective training with the limited set of labeled training images, especially with the small set of labeled training images. Next we will start from mixture model to present how we tackle this challenge by three different perspectives as introduced in Chapter 1.

Chapter 3

Finite Mixture Models

In this Chapter, we first give a brief introduction to the finite mixture model. We then present two popular forms of mixture models, i.e. Gaussian mixture model (GMM) and multinomial mixture model (MMM) for continuous and discrete-value observations, respectively. In this dissertation we employ multinomial mixture model as our baseline. We next discuss how to estimate the parameters of multinomial mixture model with the EM algorithm based on a maximum likelihood estimation (MLE) criterion. Finally, we discuss the experimental results.

3.1 Introduction

Finite mixtures are a flexible and powerful probabilistic modeling tool for univariate and multivariate data. The usefulness of mixture models is currently widely acknowledged in many areas, such as pattern recognition, computer vision, signal and image analysis, machine learning, etc. In statistical pattern recognition, mixture models are able to represent arbitrarily complex probability density functions (pdf's). This makes them an excellent choice for representing complex class-conditional pdf's (i.e., likelihood functions) in supervised learning scenarios (Hastie and Tibshirani 1996; Hinton et al. 1997), or priors for Bayesian parameter estimation (Dalal and Hall 1983).

The basic principle for setting up and computing with mixture models is to introduce unobserved indicators – random variables, which we denote as a random vector X . Let I

be one particular outcome or observed vector of X . It is said that X follows a J -component finite mixture distribution if its probability density function can be written as

$$p(I|\Lambda) = \sum_{j=1}^J \alpha_j p(I|\theta_j) \quad (3.1)$$

where $\Lambda = \{\alpha_1, \dots, \alpha_J, \theta_1, \dots, \theta_J\}$ is the complete set of parameters for the above mixture model, including mixture weight set $\{\alpha_j\}_{j=1}^J$ ($\sum_{j=1}^J \alpha_j = 1, \alpha_j > 0$), and mixture parameter set $\{\theta_j\}_{j=1}^J$. In this dissertation we assume that all the components have the same functional form, and each $p(I|\theta_j)$ is thus fully characterized by the parameter vector θ_j . The commonly used functional forms for mixtures are Gaussian and multinomial distributions.

3.1.1 Gaussian Mixture Model (GMM)

GMM has been a popular technique in practice because of the isotropic nature of Gaussian functions and their capability of representing the distribution compactly by a mean vector and covariance matrix (Medasani and Krishnapuram 1999). For example, GMM has been successfully applied in the area of automatic speech and speaker recognition to model non-Gaussian speech features (Lee et al. 1996). In computer vision applications, GMM can also be used to organize image collection as well as for color image segmentation, restoration and texture processing, and content-based image retrieval (Jain et al. 2000; Carson et al 2002).

If we assume that the j^{th} mixture component is a multivariate Gaussian density parameterized by θ_j (i.e., μ_j and Σ_j), then the form of density is as follows:

$$p(I | \theta_j) = \frac{1}{(2\pi)^{L/2} \det \Sigma_j^{1/2}} e^{-\frac{1}{2}(I-\mu_j)^T \Sigma_j^{-1} (I-\mu_j)} \quad (3.2)$$

where I is a L -dimensional feature vector, μ_j is a mean vector and Σ_j is a covariance matrix.

3.1.2 Multinomial Mixture Model (MMM)

Multinomial mixture model can be used to model discrete-valued observations, and has successfully been applied to text document classification (Novovicova and Malik 2002, 2003) and clustering (Zhang et al. 2004). The multinomial distribution has been one of the most frequently used models for language modeling of text documents in information retrieval.

We use $I = (n_1, n_2, \dots, n_L)$ to represent a text document vector where each element n_l denotes the term frequency of the l^{th} corresponding word in the document I , and L is the total size of the vocabulary. If we assume that the j^{th} mixture is a multinomial distribution parameterized by $\theta_j = (\theta_{j,1}, \theta_{j,2}, \dots, \theta_{j,L})$, then a document I is generated with the following probability:

$$p(I | \theta_j) = \frac{(\sum_{l=1}^L n_l)!}{\prod_{l=1}^L n_l!} \prod_{l=1}^L \theta_{j,l}^{n_l} \quad (3.3)$$

where $\theta_j = (\theta_{j,1}, \theta_{j,2}, \dots, \theta_{j,L})$, $\theta_{j,l} > 0$, $\sum_{l=1}^L \theta_{j,l} = 1$, and each element $\theta_{j,l}$ ($0 \leq \theta_{j,l} \leq 1$) can be interpreted as the probability of the l^{th} word generated from the j^{th} mixture

component. From Eq. (3.3) we can see the so-called naïve assumption: words are assumed to be independent of each other.

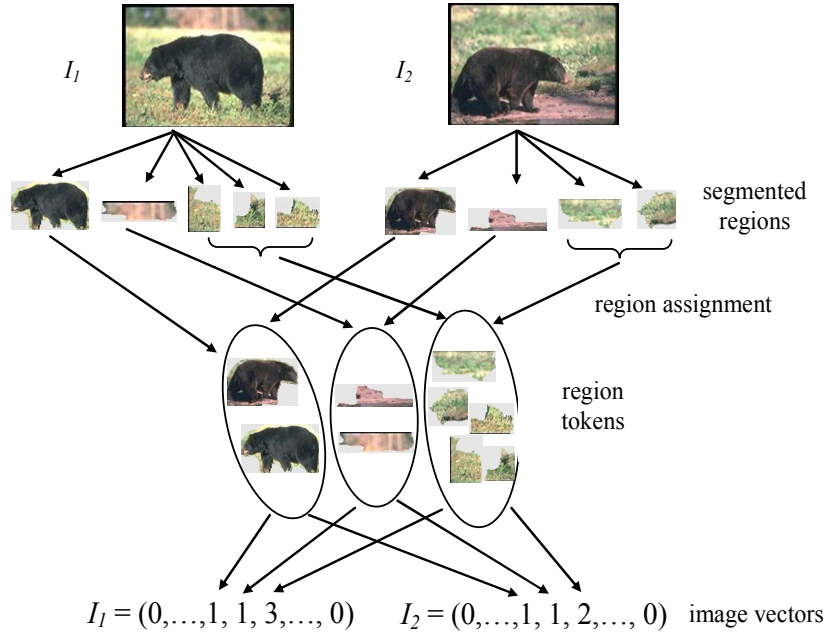


Figure 3.1: An example of image representation in this dissertation

In this dissertation we represent each image based on the vocabulary of region tokens. Thus given an image vector $I = (n_1, n_2, \dots, n_L)$, each element n_l denotes the observed count of the l^{th} corresponding region token in the document I as shown in Figure 3.1, and L is the total size of the region token vocabulary. Given a concept class c_i , we assume that $\theta_{i,j} = (\theta_{i,j,1}, \theta_{i,j,2}, \dots, \theta_{i,j,L})$ is the parameters for the j^{th} multinomial mixture component, and we rewrite Eq. (3.3) as follows:

$$p(I | \theta_{i,j}) = \frac{(\sum_{l=1}^L n_l)!}{\prod_{l=1}^L n_l!} \prod_{l=1}^L \theta_{i,j,l}^{n_l} \quad (3.4)$$

where the element $\theta_{i,j,l}$ ($1 \leq l \leq L$) represents the probability of the l^{th} region token occurring in the j^{th} mixture component of the i^{th} concept class.

3.2 Maximum Likelihood Estimation (MLE)

In this Section we focus on estimating the parameter set of $\{\theta_{i,j}\}_{j=1}^L$ for the i^{th} concept class. Based on Eq. (3.4), we assume that the likelihood functions are given a parametric form of multinomial and the corresponding parameters from each vector $\theta_{i,j}$ are unknown. Thus a classical approach to estimating these parameters is based on a maximum likelihood criterion, since MLE methods nearly always have good convergence properties as the number of training samples increases (Duda et al. 2001).

Suppose that we separate a whole collection of training image samples into each concept class based on the image annotations, so that we have a set of D_1, D_2, \dots, D_V for each corresponding concept class $\{c_1, c_2, \dots, c_V\}$, and the samples $D_i = \{I_{i,1}, I_{i,2}, \dots, I_{i,|D_i|}\}$ have been drawn independently according to the probability law $p(I | \Lambda_i)$. Since we only care about the concept class c_i in the later parts of this dissertation, we simply use the $\{I_1, I_2, \dots, I_{|D_i|}\} \in D_i$ to denote the training samples in the concept class c_i . Then we have

$$p(D_i | \Lambda_i) = \prod_{t=1}^{|D_i|} p(I_t | \Lambda_i) \quad (3.5)$$

where $\Lambda_i = \{\alpha_{i,1}, \dots, \alpha_{i,J}, \theta_{i,1}, \dots, \theta_{i,J}\}$ is the parameter set for multinomial mixture model.

Viewed as a function of Λ_i , $p(D_i | \Lambda_i)$ is called the *likelihood* with respect to observing

the set of training samples. *Maximum likelihood estimation* of Λ_i is, by definition, the value $\bar{\Lambda}_i^{ml}$ that maximizes $p(D_i | \Lambda_i)$. Intuitively, this estimate corresponds to the value that in some sense best agrees with or supports the actually observed training samples. For analytical purposes, it is usually easier to work with the logarithm of the likelihood than with the likelihood itself, because the logarithm is monotonically increasing, $\bar{\Lambda}_i^{ml}$ that maximizes the log-likelihood also maximizes the likelihood. So the estimate to Eq. (3.5) based on the ML criterion can be written as

$$\bar{\Lambda}_i^{ml} = \arg \max_{\Lambda_i} \log p(D_i | \Lambda_i) \quad (3.6)$$

Of course, if $p(D_i | \Lambda_i)$ is a well-behaved, differentiable function of Λ_i , $\bar{\Lambda}_i^{ml}$ can be found by standard methods of differential calculus. In Section 3.3, we will present an Expected-Maximization (EM) solution to Eq. (3.6).

3.3 EM Algorithm

As discussed in the previous section, maximum likelihood estimation leads to an optimization of the log-likelihood function of the parameters Λ_i . Thus given a training set D_i , we have

$$\mathcal{L}(\Lambda_i) = \log \prod_{t=1}^{|D_i|} p(I_t | \Lambda_i) = \sum_{t=1}^{|D_i|} \log \left[\sum_{j=1}^J \alpha_{i,j} p(I_t | \theta_{i,j}) \right] \quad (3.7)$$

The ML estimate, $\bar{\Lambda}_i^{ml} = \arg \max_{\Lambda_i} \mathcal{L}(\Lambda_i)$, can not be solved analytically in the case of mixture models. The usual choice for obtaining ML estimates of the mixture parameters is the classical EM algorithm (Dempster et al. 1977; McLachlan and Krishnan 1997).

The EM algorithm is based on the interpretation of D_i as incomplete data. In the case of finite mixtures, the missing part is the correspondences between mixture components and training samples. That means, given a training sample, we do not know which mixture component produced this sample. The EM algorithm maximizes $\mathcal{L}(\Lambda_i)$ iteratively by maximizing the so-called Q function given the previous estimate $\Lambda_i^{(k)}$

$$Q(\Lambda_i, \Lambda_i^{(k)}) = \sum_{j=1}^J \sum_{t=1}^{|D_i|} p^{(k)}(j | I_t) \log \left[\sum_{j=1}^J \alpha_{i,j} p(I_t | \theta_{i,j}) \right] \quad (3.8)$$

where $p^{(k)}(j | I_t)$ denotes the posterior probability given D_i and $\Lambda_i^{(k)}$. In (Dempster et al. 1977), it is proven that maximizing $Q(\Lambda_i, \Lambda_i^{(k)})$ is equivalent to maximizing $\mathcal{L}(\Lambda_i)$. This maximization problem can be solved by the method of Lagrange multipliers since we have the parameter constraints.

The EM algorithm starts with some initial guess at the ML parameters, $\Lambda_i^{(0)}$ and then proceeds iteratively to generate estimates $\Lambda_i^{(1)}$, $\Lambda_i^{(2)}$, ... by repeatedly applying the following two steps until some convergence criterion is met.

E-step: For $j = 1, 2, \dots, J$ and $t = 1, 2, \dots, |D_i|$ compute posterior probabilities

$$p^{(k)}(j | I_t) = \frac{\alpha_{i,j}^{(k)} p^{(k)}(I_t | \theta_{i,j})}{\sum_{r=1}^J \alpha_{i,r}^{(k)} p^{(k)}(I_t | \theta_{i,r})} \quad (3.9)$$

M-step: Updates the parameter estimates according to

$$\Lambda_i^{(k+1)} = \arg \max_{\Lambda_i} Q(\Lambda_i, \Lambda_i^{(k)}) \quad (3.10)$$

under the constraints $\sum_{j=1}^J \alpha_{i,j} = 1$. It leads to

$$\alpha_{i,j}^{(k+1)} = \arg \max_{\alpha_{i,j}} Q_\alpha = \arg \max_{\alpha_{i,j}} \sum_{t=1}^{|D_i|} p^{(k)}(j | I_t) \log \alpha_{i,j} \quad (3.11)$$

$$\theta_{i,j}^{(k+1)} = \arg \max_{\theta_{i,j}} Q_\theta = \arg \max_{\theta_{i,j}} \sum_{t=1}^{|D_i|} p^{(k)}(j | I_t) \log p(I_t | \theta_{i,j}) \quad (3.12)$$

Since there are some necessary parameters constraints for $\alpha_{i,j}$, $\sum_{j=1}^J \alpha_{i,j} = 1$, we apply the

method of Lagrange multipliers to optimize Eq. (3.11). Then we have

$$\alpha_{i,j}^{(k+1)} = \frac{1}{|D_i|} \sum_{t=1}^{|D_i|} p^{(k)}(j | I_t) \quad (3.13)$$

The exact formulas for the $p^{(k)}(j | I_t)$, $\alpha_{i,j}^{(k+1)}$ and $\theta_{i,j}^{(k+1)}$ depend on the involved parametric family of distributions.

3.4 Parameter Estimation with the EM Algorithm

In this dissertation, we employ the multinomial mixture model to characterize each concept class, and the EM algorithm is used to find the ML estimate of the parameters of

multinomial mixtures given the training set. Thus we focus on the problem of optimizing Eq. (3.12) of multinomial mixtures in this sub-section.

In terms of the definition of multinomial distribution based on Eq. (3.4), there are some necessary parameter constraints, $\sum_{l=1}^L \theta_{i,j,l} = 1$ ($\theta_{i,j} = (\theta_{i,j,1}, \theta_{i,j,2}, \dots, \theta_{i,j,L}), \theta_{i,j,l} > 0$). Thus we apply the method of Lagrange multipliers to optimize Eq. (3.12). Then we have the appropriate Lagrangian for $\theta_{i,j} = (\theta_{i,j,1}, \theta_{i,j,2}, \dots, \theta_{i,j,L})$

$$Q_\lambda = Q_\theta + \lambda \left(\sum_{l=1}^L \theta_{i,j,l} - 1 \right) \quad (3.14)$$

where λ is the Lagrange multiplier. By differentiating Q_λ with respect to each $\theta_{i,j,l}$, λ and setting them equal to zero, we can yield the estimate of the $\theta_{i,j,l}$ as follows:

$$\bar{\theta}_{i,j,l} = \frac{\sum_{t=1}^{|D_i|} p^{(k)}(j | I_t) n_{t,l}}{\sum_{l=1}^L \sum_{t=1}^{|D_i|} p^{(k)}(j | I_t) n_{t,l}} \quad (3.15)$$

From this estimate, we can interpret the $\theta_{i,j,l}$ as the average distribution of the l^{th} region token for images belonging to the j^{th} mixture component of the i^{th} concept class. Now we give two basic equations of the EM algorithm for fitting the multinomial mixture model are as follows:

E-step: For $j = 1, 2, \dots, J$ and $t = 1, 2, \dots, |D_i|$ compute posterior probabilities using

the current parameter estimates $\{\alpha_{i,j}^{(k)}, \theta_{i,j}^{(k)}\}$ at iteration k .

$$p^{(k)}(j | I_t) = \frac{\alpha_{i,j}^{(k)} \prod_{l=1}^L (\theta_{i,j,l}^{(k)})^{n_{t,l}}}{\sum_{r=1}^J \alpha_{i,r}^{(k)} \prod_{l=1}^L (\theta_{i,r,l}^{(k)})^{n_{t,l}}} \quad (3.16)$$

M-step: Updates $\{\alpha_{i,j}^{(k+1)}, \theta_{i,j}^{(k+1)}, j = 1, \dots, J\}$ according to

$$\alpha_{i,j}^{(k+1)} = \frac{1}{|D_i|} \sum_{t=1}^{|D_i|} p^{(k)}(j | I_t) \quad (3.17)$$

$$\theta_{i,j,l}^{(k+1)} = \frac{\sum_{t=1}^{|D_i|} p^{(k)}(j | I_t) n_{t,l}}{\sum_{s=1}^L \sum_{t=1}^{|D_i|} p^{(k)}(j | I_t) n_{t,s}} \quad (3.18)$$

3.5 Baseline Model

In this dissertation, we employ multinomial mixture model to characterize each concept class, and follow the EM algorithm in Section 3.4 to estimate the model parameters with ML criterion. In the following, we will use this model as our baseline. The advantage of mixture model is that it is a simple way to combine multiple distributions to form more complex one.

However, as discussed in Chapter 2, compared with the large variations of visual contents, we often have a limited set, or even a small set, of labeled training data, which could result in some potential difficulties such as inaccurate parameter estimations. In particular, those potential difficulties could be more serious when a large-scale mixture model is employed to cover large image variations, which could lead to poor AIA performance. Thus there are two goals in the experiments described in the next subsection: (a) we need to verify the effectiveness of our baseline model as compared to the state-of-the-art models under the same experimental settings; and (b) we need to compare

the performance of our baseline model with different numbers of mixtures, especially when we need a large number of mixtures to cover wide image variations.

3.6 Experiments and Discussions

Following the experimental settings in (Duygulu et al. 2002; Jeon et al. 2003; Srikanth et al. 2005), we conduct our experiments on the same Corel CD data set, consisting of 4500 images for training and 500 images for testing. The total number of region tokens is $L = 500$. In this corpus there are 371 concepts in the training set but only 263 such concepts appearing in the test set, with each image assigned 1-5 concepts. As with the previous studies on this AIA task, the performance is evaluated by comparing the generated annotations with the ground truth of image annotations in the testing set. We assign a set of top five concepts to each test image based on their likelihoods.

To compare the performance of a few representative state-of-the-art AIA models, we tabulate their published results on the Corel dataset in Table 3.1. These are all discrete models based on the same set of region tokens.

Table 3.1: Performance comparison of a few representative state-of-the-art AIA models and our baseline

Models	TM	CMRM	HC	Baseline ($J=1$)	Baseline ($J=5$)	Baseline ($J=25$)
# of concepts (recall>0)	49	66	93	93	104	101
Mean Per-concept metrics on all 263 concepts on the Corel dataset						
Mean Precision	0.040	0.090	0.100	0.091	0.102	0.095
Mean Recall	0.060	0.100	0.176	0.143	0.168	0.159

In order to highlight the ability to cover large variations in the image set, we select three different numbers of mixtures ($J=1$, $J=5$ and $J=25$) to emulate image variations. These three numbers are obtained by our empirical experience. The results in terms of averaging precision and recall are tabulated in Table 3.1. From Table 3.1, we can draw the following observations. (a) Among these models, HC achieved the best performance, since HC incorporated the concept hierarchy derived from WordNet into the classification. This reinforces the importance of utilizing hierarchical knowledge for AIA task. (b) As compared with our baseline ($J=1$) which used only one multinomial mixture for each concept class, our baseline ($J=5$, 25) achieved the better performance. This demonstrates again that mixture model is an effective way to cover image variations for AIA. (c) The performance of baseline ($J=25$) is worse than that of baseline ($J=5$). This is because the number of training image samples are the same in both cases and we are able to estimate the small number of parameters for baseline ($J=5$) more accurately. This result highlights the limitation of mixture models when there are large variations in image samples.

Generally speaking, when the number of mixtures is more than 25, the performance of mixture model will be worse and worse with the number of mixtures increasing, since the same limited set of training images cannot handle more and more complex model. Meanwhile, the appropriate number of mixtures should be between 2 and 24, and then some approaches like MDL (Carson et al. 1999, 2002) can be used to find such a number. But until now how to find the appropriate number of mixtures is still a hard research topic.

3.7 Summary

In this chapter, we briefly introduced the multinomial mixture model. We estimate the parameters of multinomial mixtures based on the ML criterion and EM algorithm. By taking MMM as our baseline, we compared the performance of our baseline with a few representative state-of-the-art models. The results not only indicate that our baseline is effective for AIA, but also reveal the limitations of our baseline. In next Chapter we will propose a Bayesian hierarchical multinomial mixture model to tackle this problem by incorporating prior hierarchical knowledge.

Chapter 4

Bayesian Hierarchical Multinomial Mixture Model

Having discussed our baseline model, we first present the potential difficulties resulting from the limited set of training data in this Chapter. Then we briefly introduce the Bayesian estimation, and compare the maximum likelihood estimation and Bayesian estimation. Based on the principle of Bayesian estimation, we propose a Bayesian hierarchical multinomial mixture model (BHMMM) to improve the ML estimates of our baseline model by incorporating the prior hierarchical knowledge. We then focus on addressing a few key issues in our proposed model. Finally, we discuss the experimental results on Corel CD image dataset by comparing the performance of BHMMM with our baseline and some representative state-of-the-art AIA models.

4.1 Problem Statement

As discussed in the previous chapters, we always have a limited set (even a small set) of training samples, which could lead to some potential difficulties such as mismatches between training sets and testing sets and inaccurate parameter estimations. Now we use an example to explain these difficulties by using a single multinomial distribution. As shown in Figure 4.1, we have two training image samples I_1 and I_2 for the concept class ‘black bear’ in the grass background, but they are different from the testing sample T_1 on ‘black bear’ in the water background. In terms of Eq. (3.15), the ML estimation of parameters on region tokens are $\bar{\theta}_{b_1} = 2 / 9$, $\bar{\theta}_{b_2} = 2 / 9$, $\bar{\theta}_{b_3} = 5 / 9$, $\bar{\theta}_{b_4} = 0$,

respectively. If we employ multinomial distribution to model the concept class ‘black bear’, then the likelihood that T_1 is generated from ‘black bear’ is closed to zero according to Eq. (3.4), since the ML estimation of parameter on region token b_4 corresponding to ‘water’ is zero, $\bar{\theta}_{b_4} = 0$.

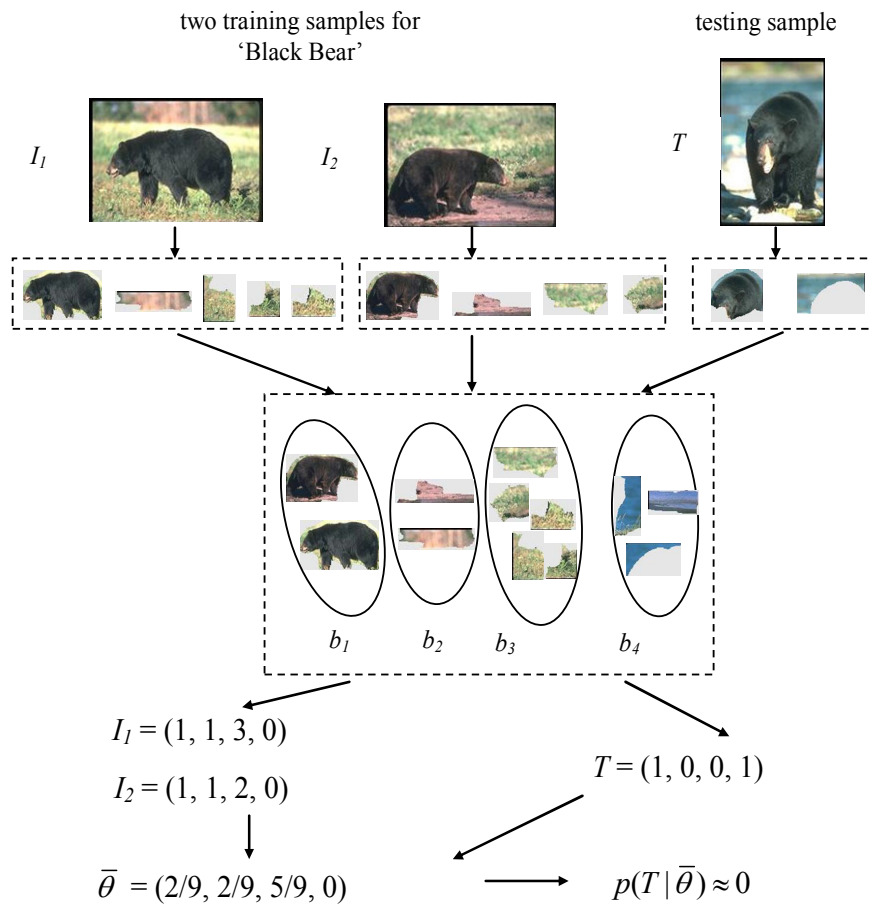


Figure 4.1: An example of potential difficulty for ML estimation

Obviously, the key reason that such potential difficulty arises is because the MLE criterion only depends on the training data. Such difficulty could be more serious when we employ more mixtures to model the concept class, as shown in the Table 3.1. Thus

starting from the introduction of Bayesian estimation in the next section, we present how to alleviate the difficulties by incorporating prior hierarchical knowledge in the following sections.

4.2 Bayesian Estimation

The problem of parameter estimation is a classical one in statistics. There are two common and reasonable procedures, namely maximum-likelihood estimation and Bayesian estimation, which are quite different conceptually (Duda et al. 2001). In this Section, we use our mixture model as an example to explain such differences. As shown in Figure 4.2 (a), the maximum likelihood estimation only depends on the training data, and the best estimation of parameter values is defined to be the one that maximizes the probability of obtaining the samples actually observed. Thus MLE views the parameters as quantities whose values are fixed but unknown.

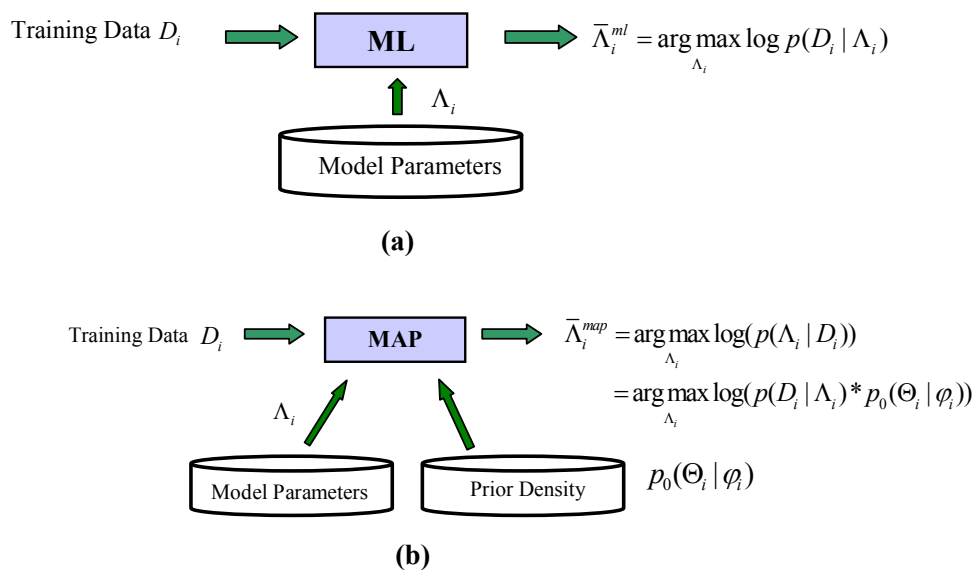


Figure 4.2: The principles of MLE and Bayesian estimation

In contrast, Bayesian estimation (or Bayesian learning) views all the parameters, $\Theta_i = \{\theta_{i,j}\}_{j=1}^L$, as random variables having some prior distribution parameterized by φ_i (often referred to as *hyperparameters*). Observation of the training samples converts this prior distribution to a posterior density, thereby revising our opinion about the true values of parameters. Obviously, Bayesian estimation facilitates a statistical combination of training data and prior information by using the criterion of the so-called maximum a posterior criterion (MAP). Thus compared with the formulation of ML estimation (Eq. (3.6)), the Bayesian estimation formulates the parameter estimation as follows:

$$\begin{aligned}\bar{\Lambda}_i^{map} &= \arg \max_{\Lambda_i} \log(p(\Lambda_i | D_i)) \\ &= \arg \max_{\Lambda_i} \log(p(D_i | \Lambda_i) * p_0(\Theta_i | \varphi_i))\end{aligned}\tag{4.1}$$

From the Eq. (4.1), we simply assume that all the mixture parameters $\Theta_i = \{\theta_{i,j}\}_{j=1}^L$ share the single prior distribution p_0 with the same set of hyperparameters, φ_i . Of course, we may not use such an assumption, but in Section 4.4 we will explain why we take this assumption in our scenario. With a posterior density $p(\Lambda_i | D_i)$, Bayesian learning approach brings more information into the problem of estimation than maximum likelihood estimation does. If the prior information is reliable, Bayesian estimation can be expected to give better results. Thus, we propose a Bayesian hierarchical multinomial mixture model (BHMMM) to enhance the ML estimation of our baseline model parameters.

Based on the formulation of Eq. (4.1), we need to address three key issues in BHMMM, namely: (a) the definition of the prior density, p_0 ; (b) the specification of the hyperparameters, φ_i ; and (c) the MAP estimation of the mixture model parameters, $\bar{\Lambda}_i^{map}$. The Bayesian estimation for Gaussian mixture model has been studied in the field of speech recognition. For example, in the research work of (Gauvain and Lee 1994; Shinoda and Lee 2001), a hierarchical prior framework for Bayesian estimation is established for Gaussian mixture model. Now we focus on the above key issues of Bayesian estimation for multinomial mixture model.

4.3 Definition of Prior Density

Generally speaking, the definition of prior density p_0 may derive from subject matter considerations and/or from previous experience. Since AIA task is a general problem in pattern recognition, we are always in absence of some special information on the definition of such a prior density. However, we have to consider the computational complexity -- an important factor that will influence our choice.

Thus conjugate prior becomes a common choice for such a consideration. In Bayesian learning theory, a class of prior probability distributions $p_0(\theta)$ is *conjugate* to a class of likelihood functions $p(I|\theta)$ if the resulting posterior distributions $p(\theta|I)$ are in the same family as $p_0(\theta)$ (Raiffa and Schlaifer 1961; Gelman et al. 2003). For example, the Gaussian family is conjugate to itself. If the likelihood is Gaussian, choosing a Gaussian prior will ensure that the posterior distribution is also Gaussian. A conjugate is an

algebraic convenience, otherwise a difficult numerical method may be necessary to find a solution to optimize the posterior density.

It is well known that a Dirichlet density is the conjugate prior for estimating the parameters of multinomial distribution so that the posterior distribution has a similar form to the Dirichlet density, which makes it easy to estimate its parameters. Such methods have been used successfully in automatic speech recognition for adaptive estimation of histograms, mixture gains, and Markov chains (Huo et al. 1995; Lee and Huo 2000). We adopt Dirichlet distribution as the prior distribution p_0 with the hyperparameter φ_i

$$p_0(\theta_{i,j} | \varphi_i) = \frac{\Gamma(\sum_{l=1}^L \varphi_{i,l})}{\prod_{l=1}^L \Gamma(\varphi_{i,l})} \prod_{l=1}^L \theta_{i,j,l}^{(\varphi_{i,l}-1)} \quad (4.2)$$

where $\varphi_i = (\varphi_{i,1}, \varphi_{i,2}, \dots, \varphi_{i,L})$, $\varphi_{i,l} > 0$, $1 \leq l \leq L$, and $\Gamma(x)$ is the Gamma function. Compared with the interpretations for multinomial distribution in Section 3.1.2, the hyperparameters $\varphi_{i,l}$ can be interpreted as the ‘prior observed count’ for the l^{th} region token in the i^{th} concept class. In Bayesian learning as shown in Eq. (4.1), the posterior density (also Dirichlet density) facilitates a statistical combination of the observed count of region tokens from training set D_i and the prior observed count of region tokens φ_i from prior density p_0 . Thus in next Section, we will discuss how to specify the hyperparameters φ_i based on our hierarchical prior knowledge.

4.4 Specifying Hyperparameters Based on Concept Hierarchy

As discussed in Section 4.3, we choose the Dirichlet distribution which is the conjugate prior of multinomial distribution as the prior density for the sake of computation

complexity. Thus, it is natural for us to incorporate some useful information into the hyperparameters φ_i to enhance the ML estimates $\bar{\Theta}_i^{ml}$, and such useful information should be obtained not only from the training set but also from some human prior knowledge.

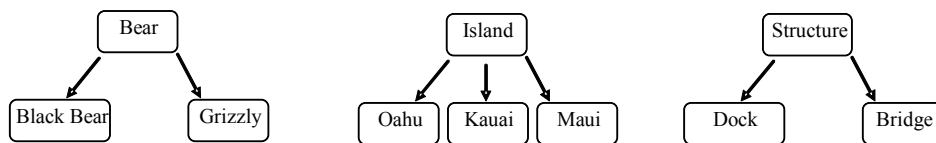


Figure 4.3 The examples of concept hierarchy

In most practical settings, we do have some domain knowledge or ontology resources that describe the dependencies among concepts often in terms of a hierarchical structure. For example, in Section 3.6, we mentioned the importance of utilizing hierarchical knowledge for AIA task, and HC approach (Srikanth et al. 2005) based on a multi-level concept hierarchy achieved the best performance among the AIA models under the same experimental settings. Figure 4.3 gives some examples on a concept hierarchy. From these examples, we can see that there are always some similar contexts shared among the sibling concepts, say, the similar wild living environment for ‘black bear’ and grizzly, tropic island sea scenes for ‘oahu’, ‘kauai’ and ‘maui’, and the structures around the water for ‘dock’ and ‘bridge’. In Figure 4.4, we show some training examples on ‘grizzly’, ‘water’ and ‘grass’.



Figure 4.4: Training image samples for the concept class of ‘grizzly’

As shown in Figure 4.1 we don’t have the training samples on ‘black bear’ and ‘water’, we want to incorporate such context information from ‘grizzly’ into hyperparameters to enhance the ML estimation of model parameters for ‘black bear’. The basic idea is that we view the hyperparameters as the shared knowledge among sibling concepts to simulate the similar context, and then we use the MAP criterion to estimate the model parameters of these sibling concepts. Obviously how to specify the hyperparameters relies on what hierarchical structure we use to model the concept dependencies. In next Section, we will discuss how to derive such a concept hierarchy.

4.4.1 Two-Level Concept Hierarchy

As shown in Figure 4.5, the simplest concept hierarchy is a two-level one in which all the concepts in (c_1, c_2, \dots, c_V) are derived from the root node labeled with ‘entity’. The advantage of using such a two-level concept hierarchy is that we do not need any prior domain knowledge. However, the two-level concept hierarchy cannot capture all the concept dependencies accurately. For instance, there is not much dependency between the concepts of ‘buildings’, ‘street’ and the concept of ‘anemone’, and most region tokens from ‘buildings’, ‘street’ are irrelevant to those from ‘anemone’.

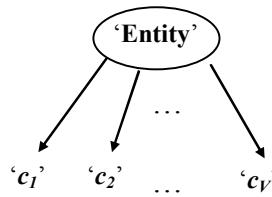


Figure 4.5: Two-level concept hierarchy

4.4.2 WordNet

Now we are interested in modeling the concept dependencies. Ontologies, such as the WordNet (Miller et al. 1990), are convenient specifications of such relationships. WordNet is an electronic thesaurus used popularly in lexical semantic acquisition. It was developed by the Cognitive Science Laboratory at Princeton University under the direction of Professor George A. Miller. It contains approximately 140'000 unique words with 111'000 different senses.

In WordNet, the meaning of English nouns, verbs, adjectives and adverbs are organized into synonym sets. Different relations, such as hypernym or hyponym relations, link the synonym sets. "Representations in WordNet are not on the level of individual words or word forms, but on the level of word meanings (lexemes). A word meaning, in turn, is characterized by simply listing the word forms that can be used to express it in a synonym set (synset). As a result, the meaning of the word in WordNet is determined by its sets of synonyms. This is essentially a recursive definition of word meaning. Hence meaning in WordNet is a structural notion: the meaning of a concept is determined by its position relative to the other words in the larger WordNet structure (Kamps 2001). For example, the word 'path' is a concept in our corpus. 'Path' has four senses in WordNet

and each sense is characterized by a sequence of words (hypernyms): (a) path←course←action←activity←abstract←entity; (b) path←way←artifact←object←entity; (c) path, route←line←location←object←entity, and (d) path, track←line←location←object←entity.

WordNet is an open source resource. Several contributions have been made to interface the WordNet Thesaurus. The Visual Thesaurus Software, for instance, gives a visual representation of WordNet Structure. Different tools can be used to visualize the Word-Net lexical database structure. In this study, we will focus on using WordNet which contains all words and has a user-friendly API for accessing its dictionary.

4.4.3 Multi-Level Concept Hierarchy

The key for building a concept hierarchy is to disambiguate the senses of words. Since the words used as annotations in our data set (Corel CD) are nouns, we only use the ‘hypernym’ relation which points to a word that is more generic than a given word in order to disambiguate the sense of words. We further assume that one word corresponds to only one sense in the whole corpus. This is reasonable as a word naturally has only one meaning within a context.

With this assumption, we adopt the basic idea that the sense of a word is chosen if the hypernyms that characterize this sense are shared by its co-occurred words in our data set. For example, the co-occurred words of ‘path’ from its training images are ‘tree’, ‘mountain’, ‘wall’, ‘flower’ and so on. Thus path←way←artifact←object←entity is chosen since this sense is mostly shared by these

co-occurred words of ‘path’. Our approach for disambiguating the senses of words is similar to that used in (Barnard et al. 2001). After this step of word sense disambiguation, every word is assigned a unique sense characterized by its hypernyms. Thus, we can easily build a multi-level concept hierarchy with ‘entity’ as the root node of the overall concept hierarchy.

4.4.4 Specifying Hyperparameters

In this Section, we discuss how to specify hyperparameters based on a concept hierarchy. As shown in Figures 4.6 (a), we have a two-level concept hierarchy in which c_i is the root node of ‘entity’ ($M=V$), or a two-level sub-tree of multi-level concept hierarchy in which c_i is the parent node of the concept set $\{c_1, c_2, \dots, c_M\}$. As shown in Figure 4.6 (b), we assume that all mixture parameters of sibling concepts, $\{\theta_{1,1}, \theta_{1,2}, \dots, \theta_{M,1}, \dots, \theta_{M,J}\}$, share the same set of hyperparameters, φ_i , we can then adopt an empirical Bayes approach (Huo et al. 1995) to estimate these hyperparameters, φ_i .

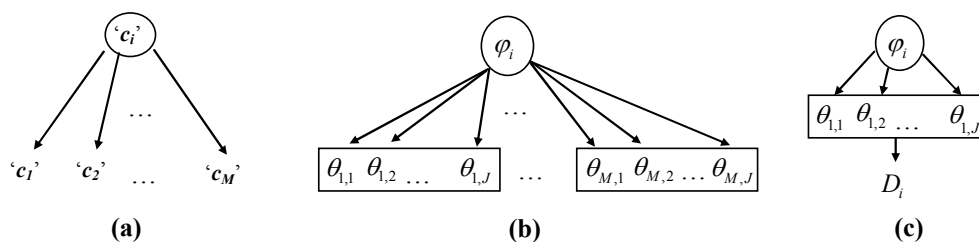


Figure 4.6: An illustration of specifying hyperparameters

Let $\bar{\Theta}_i = \{\bar{\theta}_{i,1}^{ml}, \bar{\theta}_{i,2}^{ml}, \dots, \bar{\theta}_{i,J'}^{ml}\}$ denote the mixture parameter set estimated with a ML criterion as shown in Eq. (3.6) for concept class c_i , and J' is the number of mixtures which depends on the total number of mixtures from sibling concept classes. We then pretend to view $\bar{\Theta}_i$ as a set of random samples from the Dirichlet prior $p_0(\varphi_i)$ in Eq. (3.3). Thus the ML estimate of φ_i maximizes the logarithm of the likelihood function, $\log p_0(\bar{\Theta}_i | \varphi_i)$. As pointed out in (Minka, 2003; Huang 2005), there exists no closed-form solution to this ML estimate, and the fixed-point iterative approach, can be adopted to solve for the ML estimate based on a preliminary estimate of φ_i^{old} that satisfies the following:

$$\Psi(\varphi_{i,l}^{new}) = \Psi\left(\sum_{l=1}^L \varphi_{i,l}^{old}\right) + \frac{1}{J'} \sum_{j=1}^{J'} \log \bar{\theta}_{i,j,l}^{ml} \quad (4.3)$$

where $\Psi(x) = \frac{d\Gamma(x)}{dx}$ is known as the digamma function. More details can be found in (Minka 2003; Huang 2005).

4.5 MAP Estimation

With the prior density given in Eq. (4.2) and the hyperparameters specified in Eq. (4.3), we are now ready to solve the MAP estimation in Eq. (4.1). Based on Eq. (4.1), we have a MAP estimation of model parameters as follows:

$$\bar{\Lambda}_i^{map} = \arg \max_{\Lambda_i} \{\log p(D_i | \Lambda_i) + \log p_0(\bar{\Theta}_i | \bar{\varphi}_i^{ml})\} \quad (4.4)$$

where $\bar{\varphi}_i^{ml} = (\bar{\varphi}_{i,1}^{ml}, \bar{\varphi}_{i,2}^{ml}, \dots, \bar{\varphi}_{i,L}^{ml})$, $\bar{\varphi}_{i,l}^{ml} > 0$, $1 \leq l \leq L$. Since we cannot find the analytical solution to $\log p(D_i | \Lambda_i)$ in the case of mixture model, the same is true for the MAP estimate, $\bar{\Lambda}_i^{map}$ (Figueiredo and Jain 2002). Thus we also apply the classical EM algorithm (Dempster et al. 1977; Malachlan and Krishnan 1997) to optimize Eq. (4.4). The EM algorithm maximizes Eq. (4.4) iteratively by maximizing the so-called Q' function given the previous estimate $\Lambda_i^{(k)}$ (Figueiredo and Jain 2002):

$$Q'(\Lambda_i, \Lambda_i^{(k)}) = Q(\Lambda_i, \Lambda_i^{(k)}) + \log p_0(\Theta_i | \bar{\varphi}_i^{ml}) \quad (4.5)$$

where $Q(\Lambda_i, \Lambda_i^{(k)})$ is defined in Eq. (3.8). Here we use Q' to denote the log likelihood function for MAP estimation. As discussed in Section 3.3, it is obvious that maximizing $Q'(\Lambda_i, \Lambda_i^{(k)})$ is equivalent to Eq. (4.4). This maximization problem can be solved by the method of Lagrange multipliers since we have the parameter constraints.

The EM algorithm starts with some initial guess at the parameters, $\Lambda_i^{(0)}$ and then proceeds iteratively to generate estimates $\Lambda_i^{(1)}$, $\Lambda_i^{(2)}$, ... by repeatedly applying the following two steps until some convergence criterion is met:

E-step: For $j = 1, 2, \dots, J$ and $t = 1, 2, \dots, |D_i|$ compute a posterior probability

$$p^{(k)}(j | I_t) = \frac{\alpha_{i,j}^{(k)} p^{(k)}(I_t | \theta_{i,j})}{\sum_{r=1}^J \alpha_{i,r}^{(k)} p^{(k)}(I_t | \theta_{i,r})} \quad (4.6)$$

M-step: Updates the parameter estimates according to

$$\Lambda_i^{(k+1)} = \arg \max_{\Lambda_i} Q'(\Lambda_i, \Lambda_i^{(k)}) \quad (4.7)$$

under the constraints $\sum_{j=1}^J \alpha_{i,j} = 1$. It leads to

$$\alpha_{i,j}^{(k+1)} = \arg \max_{\alpha_{i,j}} Q_\alpha = \arg \max_{\alpha_{i,j}} \sum_{t=1}^{|D_i|} p^{(k)}(j | I_t) \log \alpha_{i,j} \quad (4.8)$$

$$\begin{aligned} \theta_{i,j}^{(k+1)} &= \arg \max_{\theta_{i,j}} (Q_\theta + \log p_0(\Theta_i | \bar{\varphi}_i^{ml})) \\ &= \arg \max_{\theta_{i,j}} \left\{ \left[\sum_{t=1}^{|D_i|} p^{(k)}(j | I_t) \log p(I_t | \theta_{i,j}) \right] + \log p_0(\Theta_i | \bar{\varphi}_i^{ml}) \right\} \end{aligned} \quad (4.9)$$

Since there are some necessary parameter constraints for $\alpha_{i,j}$, $\sum_{j=1}^J \alpha_{i,j} = 1$, we apply the method of Lagrange multipliers to optimize Eq. (4.8). Thus we have

$$\alpha_{i,j}^{(k+1)} = \frac{1}{|D_i|} \sum_{t=1}^{|D_i|} P^{(k)}(j | I_t) \quad (4.10)$$

The form of Eq. (4.10) is the same as Eq. (3.13), but the computation of a posterior probability $p^{(k)}(j | I_t)$ is based on the parameters of MAP estimations.

In terms of the definition of Dirichlet distribution based on Eq. (4.2), there are some necessary parameter constraints, $\sum_{l=1}^L \theta_{i,j,l} = 1$ ($\theta_{i,j} = (\theta_{i,j,1}, \theta_{i,j,2}, \dots, \theta_{i,j,L})$, $\theta_{i,j,l} > 0$). Thus we apply the method of Lagrange multipliers to optimize Eq. (4.9). Then we have the appropriate Lagrangian for $\theta_{i,j} = (\theta_{i,j,1}, \theta_{i,j,2}, \dots, \theta_{i,j,L})$

$$Q'_\lambda = Q_\theta + \log \prod_{j=1}^J p_0(\theta_{i,j} | \bar{\varphi}_i^{ml}) + \lambda \left(\sum_{l=1}^L \theta_{i,j,l} - 1 \right) \quad (4.11)$$

where λ is the Lagrange multiplier, and Q_θ is the same as in Eq. (3.12). By differentiating Q'_λ with respect to each $\theta_{i,j,l}$, λ and setting them equal to zero, we can yield the estimate of $\theta_{i,j,l}$ as follows:

$$\bar{\theta}_{i,j,l} = \frac{\sum_{t=1}^{|D_i|} p^{(k)}(j | I_t) n_{t,l} + (\bar{\varphi}_{i,l}^{ml} - 1)}{\sum_{l=1}^L \sum_{t=1}^{|D_i|} p^{(k)}(j | I_t) n_{t,l} + \sum_{l=1}^L (\bar{\varphi}_{i,l}^{ml} - 1)} \quad (4.12)$$

So we can see that the MAP estimation facilitates a statistical combination of observed count of region tokens ($n_{t,l}$) from the training set of concept c_i and the count of region tokens ($(\bar{\varphi}_{i,l}^{ml} - 1)$) learned from concept c_i and its sibling concepts. Now we give two basic equations of EM algorithm for Bayesian MAP estimation as follows:

E-step: For $j = 1, 2, \dots, J$ and $t = 1, 2, \dots, |D_i|$ compute posterior probabilities using the current parameter estimates $\{\alpha_{i,j}^{(k)}, \theta_{i,j}^{(k)}\}$ at iteration k .

$$p^{(k)}(j | I_t) = \frac{\alpha_{i,j}^{(k)} \prod_{l=1}^L (\theta_{i,j,l}^{(k)})^{n_{t,l}}}{\sum_{r=1}^J \alpha_{i,r}^{(k)} \prod_{l=1}^L (\theta_{i,r,l}^{(k)})^{n_{t,l}}} \quad (4.13)$$

M-step: Updates $\{\alpha_{i,j}^{(k+1)}, \theta_{i,j}^{(k+1)}, j = 1, \dots, J\}$ according to

$$\alpha_{i,j}^{(k+1)} = \frac{1}{|D_i|} \sum_{t=1}^{|D_i|} p^{(k)}(j | I_t) \quad (4.14)$$

$$\theta_{i,j,l}^{(k+1)} = \frac{\sum_{t=1}^{|D_i|} p^{(k)}(j | I_t) n_{t,l} + (\bar{\varphi}_{i,l}^{ml} - 1)}{\sum_{l=1}^L \sum_{t=1}^{|D_i|} p^{(k)}(j | I_t) n_{t,l} + \sum_{l=1}^L (\bar{\varphi}_{i,l}^{ml} - 1)} \quad (4.15)$$

4.6 Exploring Multi-Level Concept Hierarchy

Given a multi-level concept hierarchy derived from WordNet, we need to explore the whole hierarchical structure to perform the MAP estimation for each concept class. Figure 4.7 shows three examples of 3-level concept hierarchy extracting from the 7-level concept hierarchy derived from WordNet.

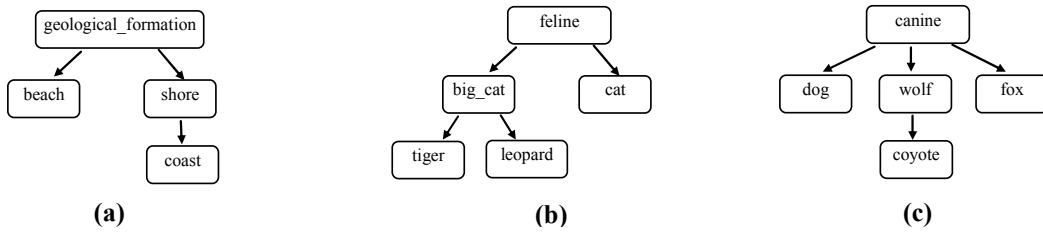


Figure 4.7: Some examples of 3-level concept hierarchy

By traversing the nodes one by one from left to right in the same level, and from root level down to the leaf level, for each node c_i in the concept hierarchy:

- 1) Let c_{ip} denote the parent node of c_i , and $p_0(\bar{\varphi}_{ip}^{ml})$ denotes the prior density function with the hyperparameters $\bar{\varphi}_{ip}^{ml}$, we have:

$$\bar{\Lambda}_i^{map} = \arg \max_{\Lambda_i} \{ \log(p(D_i | \Lambda_i)) + \log p_0(\Theta_i | \bar{\varphi}_{ip}^{ml}) \}$$

- 2) If c_i has the child nodes, then the prior density function $p_0(\bar{\varphi}_i^{ml})$ for mixture parameters of c_i can be calculated by the approach described in Section 4.4.4.

$$\bar{\varphi}_i^{ml} = \arg \max_{\varphi_i} \log p_0(\bar{\Theta}_i | \varphi_i)$$

4.7 Experiments and Discussions

Following the experimental settings in Section 3.6, we conduct our experiments on the same Corel CD data set, consisting of 4500 images for training and 500 images for testing. The total number of region tokens is $L=500$. After the derivation of concept hierarchy as discussed in Section 4.4.3, we obtained a 7-level concept hierarchy containing a total of 513 concepts, including 322 leaf concepts and 191 non-leaf concepts. The average number of children of non-leaf concepts is about 3. If a non-leaf concept node in the concept hierarchy does not belong to the concept set in Corel CD corpus, then its training set will consist of all the images from its child nodes. As with the previous studies on this AIA task, the AIA performance is evaluated by comparing the generated annotations with the actual image annotations in the test set. We assign a set of five top concepts to each test image based on their likelihoods.

4.7.1 Baseline vs. BHMMM

We first compare the performance of BHMMM (based on two-level and 7-level concept hierarchy) with the baseline mixture model. In order to highlight the ability of BHMMM to cover large variations in the image set, we select two different numbers of mixtures (5 and 25) to emulate image variations. These two numbers are obtained by our empirical experience. The results in terms of averaging precision, recall and F_1 are tabulated in Table 4.1 where TL and ML denote the 2-level and 7-level concept hierarchies respectively.

Table 4.1: Performance summary of baseline and BHMMM

Models (mixture number)	Baseline ($J=5$)	Baseline ($J=25$)	BHMMM ($J=5;TL$)	BHMMM ($J=25;TL$)	BHMMM ($J=5;ML$)	BHMMM ($J=25;ML$)
# of concepts (recall>0)	104	101	107	110	117	122
Mean Per-concept metrics on all 263 concepts on the Corel dataset						
Mean Precision	0.102	0.095	0.114	0.121	0.137	0.142
Mean Recall	0.168	0.159	0.185	0.192	0.209	0.225
Mean F1	0.117	0.109	0.133	0.140	0.160	0.169

From Table 4.1, we can draw the following observations. (a) The F1 measure of Baseline ($J=5$) is better than that of Baseline ($J=25$). This confirms our believe that with higher number of mixture models, the traditional multi-mixture model with limited amount of training samples does not perform well because of difficulty in estimating the much higher number of model parameters (when $J=25$). (b) The F1 performance of all variants of BHMMMs are better than that of the baseline ($J=5$). This indicates that the use of prior information is important to overcome the limitation of training samples in our baseline of mixture model. (c) The F1 performance of BHMMM ($J=25; ML$) is better than BHMMM ($J=5; ML$). This indicates that the use of prior information and domain hierarchy is important to alleviate the sparse training sample problem of large-scale mixture model. (d) As compared to BHMMM ($J=5, 25; TL$), BHMMM ($J=5, 25; ML$) achieves about 20% and 21% improvements on F_1 measure. This shows that the use of multi-level concept hierarchy in BHMMM (ML) can model the concept dependency more accurately, since BHMMM (ML) permits a concept node to inherit the prior information only from its parent node. Overall, BHMMM ($J=25; ML$) achieves the best performance of 0.169 in terms of F_1 measure. In the later parts of this thesis, we will use BHMMM ($J=5, 25$) to denote BHMMM ($J=5, 25; ML$) for the sake of simplicity.

4.7.2 State-of-the-Art AIA Models vs. BHMMM

For further comparison, we tabulate the performance of a few representative state-of-the-art AIA models in Table 4.2. These are all *discrete* models that used the same experimental settings as shown in Table 4.1 and Table 3.1. The discrete models refer to translation model (TM) (Duygulu et al. 2002), cross-media relevance model (CMRM) (Jeon et al. 2003) and hierarchical classification approach (HC) (Srikanth et al. 2005).

Table 4.2: Performance comparison of state-of-the-art AIA models and BHMMM

Models	TM	CMRM	HC	BHMMM ($J=25$)
# of concepts (recall>0)	49	66	93	122
Mean Per-concept metrics on all 263 concepts on the Corel dataset				
Mean Precision	0.040	0.090	0.100	0.142
Mean Recall	0.060	0.100	0.176	0.225

From Table 4.2, we can draw the following observations. (a) Among these models, HC achieved the best performance in terms of precision and recall measures, since HC also incorporated the concept hierarchy derived from the WordNet into the classification. This further reinforces the importance of utilizing the hierarchical knowledge for AIA task. (b) As compared with HC which used only one mixture for each concept class and adopted ML criterion to estimate the parameters, BHMMM ($J=25$) achieved about 40% and 28% improvements on the measure of mean per-concept precision and mean per-concept recall respectively. This demonstrates again that our proposed BHMMM is effective to AIA.

4.7.3 Performance Evaluation with Small Set of Samples

This Section analyzes the effect of our proposed BHMMM when the number of original training images is small. We selected a subset of 132 testing concepts in Corel CD dataset in which the number of training samples in each class is no more than 21.

Table 4.3: Performance summary of baseline and BHMMM on the concept classes with small number of training samples

Models	Baseline ($J=5$)	BHMMM ($J=25$)
# of concepts (recall>0)	14	25
Mean Per-concept metrics on all 132 concepts on the Corel dataset (# of original training samples ≤ 21)		
Mean Precision	0.023	0.059
Mean Recall	0.061	0.106
Mean F_1	0.033	0.069

In Table 4.3 we compare two models, the baseline ($J=5$) and BHMMM ($J=25$). It is clear that for this set of concepts, the performances were in general much worse than those shown in Table 4.1. For example, the mean F_1 was degraded from 0.169 in Table 4.1 to 0.069 in Table 4.3 for BHMMM ($J=25$). Compared with the baseline ($J=5$), BHMMM achieved much better performance in terms of mean precision, recall and F_1 measures. This again demonstrates that prior knowledge is critical for parameter estimation of visual mixture models, especially when the number of training examples is small.

4.8 Summary

In this chapter, we incorporated prior knowledge into hierarchical representation of concepts to facilitate modeling of multi-level concept structures. To alleviate the potential difficulties arising from limited set of (even a small set of) training images, we proposed a Bayesian hierarchical mixture model (BHMMM) framework. By treating the mixture model parameters as random variables characterized by a joint conjugate prior density, BHMMM facilitates a statistical combination of the likelihood function of the available training data and the prior density of the concept parameters into a well-defined posterior density whose parameters can now be estimated via a maximum a posteriori criterion. Conceptually the training set for BHMMM and our baseline multinomial mixture model (MMM) is the same, and BHMMM does not need more training images than our baseline model does.

On the one hand when no training data are used, the MAP estimate can only depend on the prior density. On the other hand when a large amount of training data is available the MAP estimate can be shown to asymptotically converge to the conventional maximum likelihood estimate. This desirable property makes the MAP estimate an ideal candidate for parameter estimation when we have a limited set of (even a small set of) training data.

Experimental results on the Corel image dataset showed that our proposed BHMMM approach, using a multi-level structure of 371 concepts with a maximum of 25 mixture components per concept, achieves a mean F_1 measure of 0.169, which outperforms many state-of-the-art techniques for automatic image annotation. In particular, our proposed

BHMMM outperforms our baseline model on a subset of 132 testing concepts in Corel CD dataset in which the number of training samples in each class is no more than 21.

Chapter 5

Extended AIA Based on Multimodal Features

In this Chapter, we first introduce the motivations to propose extended AIA to alleviate the potential difficulties. Then we extend the traditional AIA to three modes, namely visual-AIA, text-AIA and text-visual-AIA that are used to effectively expand the original image annotations and acquire more training samples for concept classes, and discuss these extended AIA models respectively. Finally, by comparing our extended AIA models with our baseline and some state-of-the-art AIA models, we discuss the experimental results on Corel image dataset by combining the additional training images acquired from annotation expansions.

5.1 Motivation

As discussed in Chapter 4, we proposed a BHMMM framework to enhance the ML estimation of large-scale multinomial mixture models, which can alleviate the potential difficulties resulting from limited set of training data by incorporating prior hierarchical knowledge. Since most existing AIA models, especially mixture models, depend heavily on a large number of training samples for effective training, we therefore study the issues related to acquiring more training samples automatically for each concept class in this Chapter.

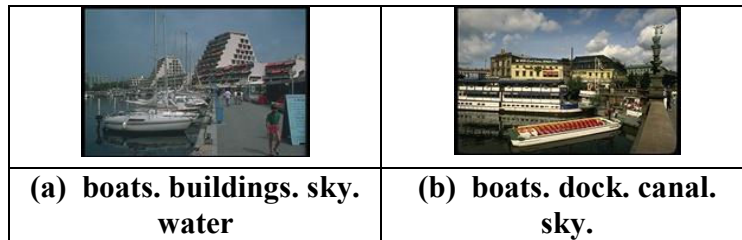


Figure 5.1: Two image examples with incomplete annotations

Our research work is motivated by two aspects. (a) Most image collections often come with few and incomplete annotations. For example, Figure 5.1 shows the original annotations of two images coming from the Corel image corpus. Given the predefined set of concepts $\{\text{'boats'}$, 'buildings' , 'sky' , 'water' , 'dock' , 'canal' , $\dots\}$, The possible missing annotation for the image in Figure 5.1a could be 'dock' , and that for the image in Figure 5.1b could be 'buildings' . (b) As discussed in Section 2.6, most existing AIA approaches, including both classification-based models and joint probability-based models, neglect to use the available text information from the training set and ontological information from prior knowledge to effectively annotate the training images or expand the original annotations of training images.

5.2 Extended AIA

Two groups of information, i.e. text and visual features, are available for a given training image. Thus, there are several key issues related to fusing text and visual information to acquire more training samples: (a) accurate parameter estimation especially when the number of training samples is small; and (b) dependency between visual and text features. To tackle the first issue, we incorporate prior knowledge into the hierarchical

concept representation, and extend our proposed BHMMM to different features to estimate the parameters of concept mixture models. To address the first and second issues, we propose a text-visual hierarchical multinomial mixture model to model the dependencies between text and visual mixtures and expand the annotations.

To better explain our proposed framework to obtain additional training samples for each concept we assume that the original set of concept labels associated with training images is incomplete. We extend the definition of conventional AIA to three modes, namely associating concepts to images represented by visual features, briefly called as visual-AIA, by text features as text-AIA, and by both text and visual features as text-visual-AIA. Clearly visual-AIA is similar to conventional AIA, since both approaches can be used to associate visual features to concepts. But visual-AIA is performed on the training images to obtain extra labels. Here we emphasize that only models for visual-AIA can be used in the testing phase to perform the conventional AIA, but all the models for visual-AIA, text-AIA or text-visual-AIA can be employed in the training phase for acquiring more image annotations for each concept.

As shown in Figure 5.2, we propose a novel framework to expand the image annotations and acquire image samples for concept classes. Given the annotated images in training phase, extended AIA model is first used to expand the original image annotations, and then the images with expanded annotations are taken as the new set of training samples for the conventional AIA model. Here the conventional AIA refers to associating visual features to text annotations. Obviously, our proposed framework is general, and a lot of models can be also used to expand the annotations or perform the conventional AIA.

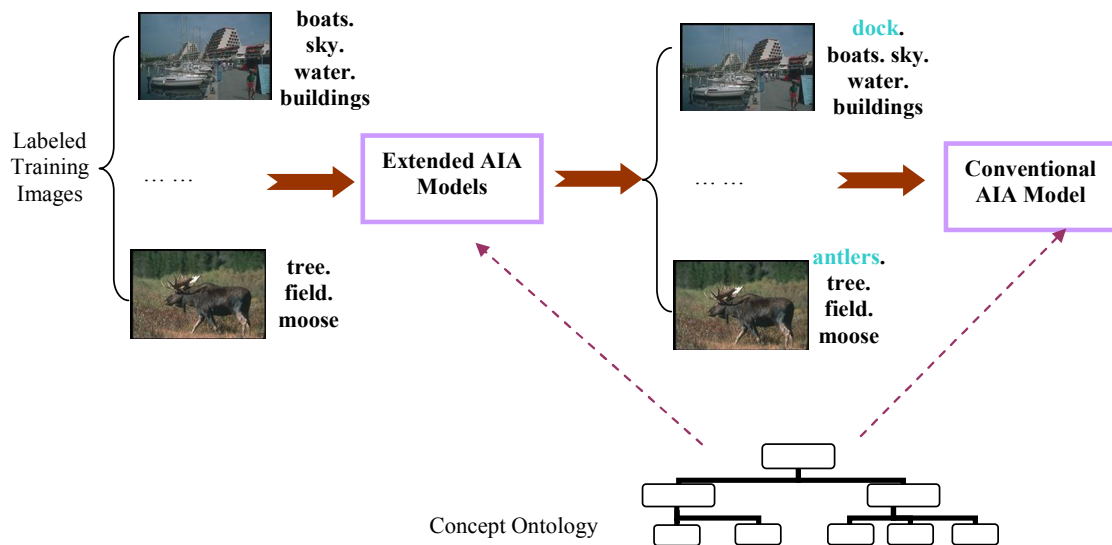


Figure 5.2: The proposed framework of extended AIA

In this dissertation, we employ our proposed BHMMM ($J=25$) as our baseline to perform conventional AIA, since BHMMM ($J=25$) is one of the state-of-the-art conventional AIA models. The concept ontology derived from WordNet in Section 4.4.3 is used to model the concept relationships and estimate the hyperparameters of BHMMM ($J=25$) and extended AIA models. The algorithm of our proposed framework is as follows:

<p>Input: The set of training images D_i for a given concept class c_i;</p> <p>Output: The estimated model parameters with MAP criterion, $\bar{\Lambda}_i^{map}$;</p> <ol style="list-style-type: none"> 1) Given training set of images, estimate the parameters of extended AIA models of each concept class with a MAP criterion. 2) For any concept c_i, expand the annotations of images related to c_i by extended AIA models. 3) Generate a rank list of images for c_i based on their likelihoods. 4) Expand training set of c_i by combining the fixed top <i>percentage</i> of candidate images. 5) Estimate the parameters of BHMMM for c_i by combining the additional and original training set, and then perform conventional AIA model to annotate the test images.

Given a concept in step 2 of this algorithm, we do not perform extended AIA models in the whole training set to expand the annotations of all the training images, since the size of the whole training corpus can be very large. Instead, extended AIA models are performed only on the set of images that are related to the given concept. The set of related concepts includes three parts: (a) the closest hypernym concept; (b) the co-occurred concepts in the training corpus; and (c) the co-occurring concepts from its sibling concepts.

5.3 Visual-AIA Models

Here the visual-AIA model is used to expand the original image annotations only based on visual features. In terms of the comparisons in Sections 4.7.1 and 4.7.2, our proposed BHMMM ($J=25$) achieved the best performance as compared to the other representative AIA models, such as HC. Hence we employ BHMMMs ($J=25$) as the visual-AIA model

to effectively perform annotation expansions by incorporating the concept ontology information.

5.3.1 Experiments and Discussions

In this Section we want to testify the effectiveness of visual-AIA model for acquiring additional image samples. We use the conventional AIA performance of BHMMM ($J=25$) to evaluate such effectiveness. In step 4 of our pipeline in Section 5.3, we picked the top 5% (the mean number of the increased training samples for each concept class is about 7 images), and top 10% (the mean number of the increased training samples for each concept class is about 15 images) of additional samples, which will be the same for all the models from text-AIA and text-visual-AIA.

Table 5.1 Performance of BHMMM and visual-AIA

Models	BHMMM ($J=25$)	visual-AIA (top 5%)	visual-AIA (top 10%)
# of concepts (recall>0)	122	133	141
Mean Per-concept metrics on all 263 concepts on the Corel dataset			
Mean Precision	0.142	0.143	0.147
Mean Recall	0.225	0.261	0.282
Mean F_1	0.169	0.171	0.174

The results in Table 5.1 indicate a clear trend that the use of additional training examples is beneficial in our visual-AIA framework, since the performance of both visual-AIA (top 5%) and visual-AIA (top 10%) were better than that of BHMMM ($J=25$). In particular, visual-AIA (top10%) gave the best performance 0.147, 0.282 and 0.174 in

terms of the precision, recall and F_1 measurements respectively. In next Section we focus on how to apply text-AIA models to perform annotation expansions.

5.4 Text-AIA Models

In the training set, a given image I has been labeled by some concept annotations, and can be represented by a concept vector, $I_c = (m_1, m_2, \dots, m_V)$, where V is the total number of predefined concepts ($\mathcal{C} = \{c_1, c_2, \dots, c_V\}$), and $m_v (1 \leq v \leq V)$ denotes the observed count of the v^{th} concept in image I . We use $D_i (I_i \in D_i)$ to denote a collection of independent training images for concept class c_i . In this Section, we introduce two text-based models, text mixture model and text-based Bayesian model, to perform the annotation expansions by utilizing the text annotations from the training set.

5.4.1 Text Mixture Model (TMM)

In the scenario of AIA, each labeled training image is a text document represented by a concept vector, I_c . Here we formulate the task of expanding annotations as a multi-class text classification problem. The objective of text classification is to assign one or more predefined set of topic classes to a text document. As pointed out in (Novovicova and Malik 2002, 2003), mixture models are suitable for text classification since each class often consists of multiple topics. This is also true for the scenario of AIA task. For example, the concept ‘arts’ consists of the topics on ‘sculpture’, ‘paintings’, ‘carvings’ and so on, in the Corel dataset. Furthermore, the multinomial mixture model has been demonstrated to be effective on the dataset of Reuters-21578 (Novovicova and Malik

2002, 2003). Thus we take the multinomial mixture model as the text classifier. Given a total of H text mixture components, the observed vector I_c from concept class c_i is assumed to have the following probability:

$$p(I_c | \Omega_i) = \sum_{h=1}^H \beta_{i,h} p(I_c | \chi_{i,h}) \quad (5.1)$$

where $\Omega_i = \{\beta_{i,1}, \dots, \beta_{i,H}, \chi_{i,1}, \dots, \chi_{i,H}\}$ is the parameter set for the text mixture model, including mixture weight set $\{\beta_{i,h}\}_{h=1}^H$ ($\sum_{h=1}^H \beta_{i,h} = 1$), mixture parameter set $\Gamma_i = \{\chi_{i,h}\}_{h=1}^H$, and $p(I_c | \chi_{i,h})$ is the h^{th} mixture component to characterize the class distribution. Here each parameter $\chi_{i,h,v}$ in $\chi_{i,h}$ can be interpreted as the average distribution of the v^{th} concept for images belonging to h^{th} mixture component of the i^{th} concept class. We call the Eq. (5.1) as the text mixture model.

5.4.2 Parameter Estimation for TMM

Maximum likelihood estimation is the usual choice to estimate the parameters. But as discussed in Sections 3.3 and 3.4, maximum likelihood estimation cannot be solved analytically in the case of mixture models. Thus given the training images represented by text vectors, we employ the EM algorithm in Section 3.4 to find the maximum likelihood estimation of the parameters of multinomial mixtures. Here the log likelihood function for TMM is as follows:

$$\mathcal{L}(\Omega_i) = \log \prod_{t=1}^{|D_i|} p(I_{c,t} | \Omega_i) = \sum_{t=1}^{|D_i|} \log \left[\sum_{h=1}^H \beta_{i,h} p(I_{c,t} | \chi_{i,h}) \right] \quad (5.2)$$

The EM algorithm starts with some initial guess at the ML parameters, $\Omega_i^{(0)}$ and then proceeds iteratively to generate estimates $\Omega_i^{(1)}$, $\Omega_i^{(2)}$, ... by repeatedly applying the following two steps until some convergence criterion is met. The algorithm is as follows:

E-step: For $h = 1, 2, \dots, H$ and $t = 1, 2, \dots, |D_i|$ compute posterior probabilities using the current parameter estimates $\{\beta_{i,h}^{(k)}, \chi_{i,h}^{(k)}\}$ at iteration k .

$$p^{(k)}(h | I_{c,t}) = \frac{\beta_{i,h}^{(k)} \prod_{v=1}^V (\chi_{i,h,v}^{(k)})^{m_{t,v}}}{\sum_{r=1}^H \beta_{i,r}^{(k)} \prod_{v=1}^V (\chi_{i,r,v}^{(k)})^{m_{t,v}}} \quad (5.3)$$

M-step: Updates $\{\alpha_{i,h}^{(k+1)}, \beta_{i,h}^{(k+1)}, h = 1, \dots, H\}$ according to

$$\beta_{i,h}^{(k+1)} = \frac{1}{|D_i|} \sum_{t=1}^{|D_i|} p^{(k)}(h | I_{c,t}) \quad (5.4)$$

$$\chi_{i,h,v}^{(k+1)} = \frac{\sum_{t=1}^{|D_i|} p^{(k)}(h | I_{c,t}) m_{t,v}}{\sum_{s=1}^V \sum_{t=1}^{|D_i|} p^{(k)}(h | I_{c,t}) m_{t,s}} \quad (5.5)$$

where we use $I_{c,t} = (m_{t,1}, m_{t,2}, \dots, m_{t,V})$ to denote a test image in D_i , and $m_{t,v}$ ($1 \leq v \leq V$) denotes the observed count of the v^{th} concept in image $I_{c,t}$.

So far, we have not discussed how to choose H , the number of mixture components. In our proposed BHMMM, we choose two fixed numbers of mixture components (i.e. 5 and 25) to emulate the large variations among images. As our experience, however, we do not need a large number of mixtures for text as compared to visual modality in order to emulate the text variations in Corel dataset. Thus we would like to choose the value of H which can best suit the natural number of text groups of training images in a concept

class. Given the log likelihood function (Eq. (5.2)), we can apply the Minimum Description Length (MDL) principle to select among values of H by maximizing the followed measure (Rissanen 1978, 1989):

$$\mathcal{L}(\Omega_i) - \frac{T_H}{2} \log(|D_i|) \quad (5.6)$$

where T_H is the number of free parameters needed for a model with H mixture components. In the case of our scenario, we have $T_H = H * V$. As a consequence of this principle, when models use two values of H to fit the data equally well, the simpler model will be chosen. For our experiments, H ranges from 1 to 12.

5.4.3 Text-Based Bayesian Model (TBM)

In terms of our observations, TMM does not always work well if there is a mismatch between training and test image samples when we have a limited set of (even a small set of) training data. Here the test image means the other set of training images labeled with concept annotations. For example, the bag-of-keywords in the class of ‘dock’ is a set of annotations, {‘boats’, ‘mountain’, ‘water’, ‘sky’, ‘clouds’, ‘ships’, ‘canal’}, and the four training images and their annotations for the class of ‘dock’ are shown in Figure. 5.3.

			
boats. mountain. water.	boats. sky. water.	clouds. ships. water.	boats. dock. canal. sky.

Figure 5.3: Four training images and their annotations for the class of ‘dock’

Due to the incomplete annotations for the first, third and fourth training images, the concept annotation ‘buildings’ does not appear in this training set. Thus given a training image labeled with ‘buildings’, ‘boats’, ‘sky’ and ‘water’ as shown in Figure. 5.1a, this image could not be annotated with the concept ‘dock’ by TMM, since TMM employs MLE to estimate the model parameters only based on the training data in the concept class.

As pointed out in (Zhai and Lafferty 2001), smoothing of the maximum likelihood estimation is extremely important for the text classification problem when the number of training samples is small. They summarized that the basic idea behind the current smoothing methods lies in the linear combination between maximum likelihood estimations of multinomial parameters and a vector of $(\mu p(c_1), \mu p(c_2), \dots, \mu p(c_v))$. Here μ is an empirical constant, and $p(c_v)$ is the relative frequency of observing the keyword c_v in the whole training set of all the classes. However, these smoothing methods ignore the concept dependency. For example, if we want to estimate the model parameters of the ‘tiger’ class, then the relative frequency of observing the keyword of ‘street’, ‘buildings’ should be lower. But if we want to estimate the model parameters of ‘city’, then the relative frequency of observing the keyword of ‘street’, ‘buildings’ should be higher.

Thus we would like to take another way to enhance the ML estimations by incorporating prior knowledge. We assume the mixture parameters in $\chi_{i,k}$ as random variables with a joint prior density $p_0(\chi_{i,k} | \tau_i)$ with parameters τ_i (referred to as *hyperparameters*). Thus, the posterior probability of observing the training set can be evaluated as:

$$P(\Omega_i | D_i) \propto \left\{ \prod_{t=1}^{|D_i|} \sum_{h=1}^H [\beta_{i,h} P(I_{c,t} | \chi_{i,h})] \right\} * p_0(\Gamma_i | \tau_i) \quad (5.7)$$

In contrast to conventional ML estimation, we can impose a maximum a posterior (MAP) criterion to estimate the parameters as follows:

$$\bar{\Omega}_i^{map} = \arg \max_{\Omega_i} \log \left\{ \prod_{t=1}^{|D_i|} \sum_{h=1}^H [\beta_{i,h} P(I_{c,t} | \chi_{i,h})] \right\} * p_0(\Gamma_i | \tau_i) \quad (5.8)$$

where $\Omega_i = \{\beta_{i,1}, \dots, \beta_{i,H}, \chi_{i,1}, \dots, \chi_{i,H}\}$ is the parameter set for the text mixture model, and $\Gamma_i = \{\chi_{i,h}\}_{h=1}^H$.

To better model the concept dependencies, we derive concept ontology through WordNet as shown in Section 4.4.3. Thus, we propose a text-based Bayesian learning model to characterize the concept ontology structure. Here we also assume that the mixtures from the sibling concepts share the same set of hyperparameters and these concept mixture models are constrained by a common prior density parameterized by this set of hyperparameters. This is reasonable since given a concept (say, ‘dock’), the image annotations from its sibling concept (say, ‘bridge’) are often related. For example, the keywords in the class of the concept ‘bridge’ are ‘water’, ‘boats’, ‘buildings’, ‘canal’, ‘sky’, etc., which are closely relevant to the concept ‘dock’. We also call such similar context among sibling concepts as the ‘shared knowledge’. Thus the hyperparameters can be interpreted as the shared prior knowledge among the sibling concepts. Figure 5.4a shows a sub-tree of a multi-level concept ontology in which the child concepts (c_1, c_2, \dots, c_M) are derived from their parent node, labeled ‘ c_i ’. As shown in Figure 5.4b, we assume that all mixture parameters of the child concepts, $\{\chi_{1,1}, \chi_{1,2}, \dots, \chi_{M,1}, \dots, \chi_{M,J}\}$, share the same set of hyperparameters, τ_i ,

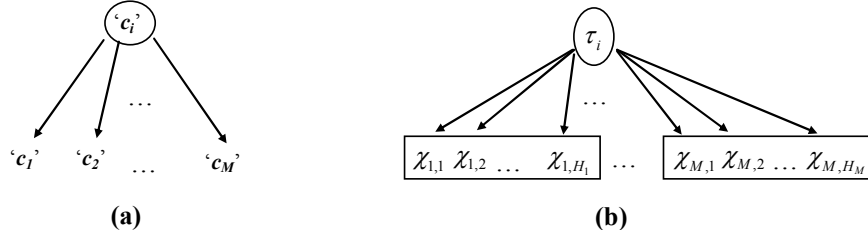


Figure 5.4: An illustration of TBM

Thus based on the Eq. (5.7) and (5.8), TBM needs to address three key issues, namely: (a) the definition of the prior density, p_0 ; (b) the specification of the hyperparameters based on concept ontology, τ_i ; and (c) the MAP estimation of the mixture model parameters, $\bar{\Omega}_i^{map}$.

5.4.4 Parameter Estimation for TBM

As discussed in Section 4.3, we also define p_0 as the Dirichlet density, and employ the same approach as in Section 4.4.4 to specify the hyperparameters, τ_i . With the Dirichlet prior density and the specified hyperparameters $\bar{\tau}_i^{ml}$, we have a MAP estimation of model parameters by rewriting Eq. (5.8) as follows:

$$\bar{\Omega}_i^{map} = \arg \max_{\Omega_i} \log \left\{ \prod_{t=1}^{|D_i|} \sum_{h=1}^{H_i} [\beta_{i,h} p(I_{c,t} | \chi_{i,h})] \right\} * p_0(\Gamma_i | \bar{\tau}_i^{ml}) \quad (5.9)$$

where $\bar{\tau}_i^{ml} = (\bar{\tau}_{i,1}^{ml}, \bar{\tau}_{i,2}^{ml}, \dots, \bar{\tau}_{i,V}^{ml})$ is the specified hyperparameter, $\bar{\tau}_{i,v}^{ml} > 0, 1 \leq v \leq V$. As discussed in Section 4.5, we still employ the EM algorithm to find the analytical solution of MAP estimations.

The EM algorithm starts with some initial guess at the parameters, $\Omega_i^{(0)}$ and then proceeds iteratively to generate estimates $\Omega_i^{(1)}$, $\Omega_i^{(2)}$, ... by repeatedly applying the following two steps until some convergence criterion is met. The algorithm is as follows:

E-step: For $h = 1, 2, \dots, H_i$ and $t = 1, 2, \dots, |D_i|$ compute posterior probabilities using the current parameter estimates $\{\beta_{i,h}^{(k)}, \chi_{i,h}^{(k)}\}$ at iteration k .

$$p^{(k)}(h | I_{c,t}) = \frac{\beta_{i,h}^{(k)} \prod_{v=1}^V (\chi_{i,h,v}^{(k)})^{m_{t,v}}}{\sum_{r=1}^{H_i} \beta_{i,r}^{(k)} \prod_{v=1}^V (\chi_{i,r,v}^{(k)})^{m_{t,v}}} \quad (5.10)$$

M-step: Updates $\{\beta_{i,h}^{(k+1)}, \chi_{i,h}^{(k+1)}, h = 1, \dots, H_i\}$ according to

$$\beta_{i,h}^{(k+1)} = \frac{1}{|D_i|} \sum_{t=1}^{|D_i|} p^{(k)}(h | I_{c,t}) \quad (5.11)$$

$$\chi_{i,h,v}^{(k+1)} = \frac{\sum_{t=1}^{|D_i|} p^{(k)}(h | I_{c,t}) m_{t,v} + (\bar{\tau}_{i,h}^{ml} - 1)}{\sum_{s=1}^V \sum_{t=1}^{|D_i|} p^{(k)}(h | I_{c,t}) n_{t,s} + \sum_{v=1}^V (\bar{\tau}_{i,h}^{ml} - 1)} \quad (5.12)$$

Here H_i is the number of mixture components of concept class c_i . In our experiments, we take the same H_i as the component number obtained by MDL principle (Eq. (5.6)) of ML estimation in Section 5.4.1.

5.4.5 Experiments and Discussions

As BHMMM with $J=25$ (where J is the number of the mixtures) without expanding the annotations by TBM and TMM, achieved the best performance among conventional AIA systems. We thus take BHMMM with $J=25$ as our baseline.

First, we want to verify the effectiveness of our proposed framework and TBM. We use BHMMM ($J=25$) as the conventional AIA. In the step 4 of our pipeline in Section 5.3, we pick the top 5% (the mean number of the increased training samples for each testing concept class is about 7 images) or top 10% (the mean number of the increased training samples for each testing concept class is about 15 images) of additional samples.

Table 5.2: Performance comparison of TMM and TBM for text-AIA

Models	BHMMM ($J=25$)	TMM (top5%)	TMM (top10%)	TBM (top5%)	TBM (top10%)
# concepts (recall>0)	122	134	143	152	153
Mean Per-concept metrics on all 263 concepts on the Corel dataset					
Mean Precision	0.142	0.143	0.145	0.152	0.156
Mean Recall	0.225	0.278	0.301	0.330	0.341
Mean F_1	0.169	0.177	0.181	0.184	0.188

We tabulate the performance of TBM and TMM in Table 5.2. We derive the following observations from Table 5.2. (a) The use of additional training examples derived from TMM and TBM is beneficial, since the performance of TMM- and TBM models are better than that of BHMMM ($J=25$). This demonstrates that text information is important and effective to expand the original annotations. (b) As compared with TMM (top 5% and 10%), TBM achieved even better performance in mean precision, and recall mF_1 measures. In particular, TBM (top 10%) achieves the best performance 0.188 of mF_1 measures, and detects 153 of 263 testing concepts.









<i>antlers</i>	<i>coast</i>	<i>railroad</i>	<i>jet</i>
			
bulls. field. elk. grass.	boats. harbor. sky. water.	locomotive. tracks. train. snow.	plane. prop. runway.
			
field. moose. tree.	beach. sand. sky. water.	locomotive. road. train.	formation. sky. plane. prop.

Figure 5.5: Examples of top additional training samples obtained from both TMM and TBM

Figure 5.5 shows some examples of top training samples obtained from both TMM and TBM. The blue italic keywords denote the concept class or the additional annotation for the corresponding training images, and black keywords denote the original image annotations. From these examples, we can easily observe the problem of incomplete annotations. Meanwhile, the additional annotations added in these examples are detected correctly by both TMM and TBM, which demonstrate the effectiveness of text features and our text-based models.



<i>beach</i>	<i>dock</i>
	
sand. water. seals.	boats. sky. water. buildings.
<pre> graph TD A[geological_formation] --> B[beach] A --> C[shore] C --> D[coast] </pre>	<pre> graph TD A[structure] --> B[dock] A --> C[bridge] </pre>

Figure 5.6: Examples of top additional training samples obtained from TBM

Figure 5.6 shows two examples of the top additional annotations or training samples obtained only from TBM. The red keywords (‘seals’ and ‘buildings’) do not occur in the training set of the corresponding concept class (‘beach’ and ‘dock’). But the keyword annotation ‘seals’ occurs in the training set of the concept classes ‘shore’ and ‘coast’, and the keyword annotation ‘buildings’ occurs in the training set of the concept class ‘bridge’. From these examples, we derive the following observations. (a) The ML estimations does not work if there is a mismatch between training and testing samples. (b) TBM can effectively enhance the ML estimations of model parameters by incorporating the prior knowledge into the text models.

Now we want to analyze the effectiveness of our proposed framework with TBM when the number of original training images is small. We still selected a subset of 132 test concepts in Section 4.1 in which the number of training examples in each class is less than 21.

Table 5.3: Performance summary of TMM and TBM on the concept classes with small number of training samples

Models	BHMMM ($J=25$)	TMM (top10%)	TBM (top10%)
# of concepts (recall>0)	25	50	57
Mean Per-concept metrics on all 132 concepts on the Corel dataset (# of original training samples ≤ 21)			
Mean Precision	0.059	0.071	0.090
Mean Recall	0.106	0.264	0.333
Mean F1	0.069	0.104	0.128

Table 5.3 compares three models, BHMMM ($J=25$), TMM (top10%) and TBM (top10%). Obviously, TBM achieves the best performance 0.090, 0.333 and 0.128 in terms of mean precision, recall and F1 measurements. This indicates again that our proposed framework and TBM are effective in acquiring more training samples even when the number of training samples is small.

5.5 Text-Visual-AIA Models

In this Section, we discuss the problem on combining text and visual modalities to acquire ‘more appropriate’ training samples in this section. We mainly focus on two fusion models. One is the linear fusion model, and the other is our proposed text-visual Bayesian model.

5.5.1 Linear Fusion Model (LFM)

Given an image represented by text and visual feature vectors, the easiest way to deal with these two feature vectors is to concatenate them into an extended feature vector

instead of using individual component vectors, which can be thought of as “feature-level fusion”. Machine learning algorithms, such as SVM, can then be used to train classifiers for the extended feature vectors. As pointed out in (Hastie et al. 2001), this creates a major problem of the curse of dimensionality. For example, there are 500 region tokens in Corel dataset to represent image visual contents, and we have a vocabulary of 374 concepts to represent a text vector of an image.

Therefore linear and non-linear fusions of scores produced by different features are popular alternatives to fuse the multi-modal features. Some of them have led to better performance than the concatenation method (Chen and Hauptman 2004; Naphade et al. 1998; Smith et al. 2003; Tong et al. 2005; Yan et al. 2003). The basic idea behind linear fusion is that the outputs (likelihood, posterior probability, et al.) from different modalities are taken as new feature vectors to represent each class, and then the coefficient of each feature for the final combination is learned in the new feature space. Based on the experiments on large-scale TRECVID’02 video (Yan and Hauptman 2003), it was concluded that linear fusion can be an appropriate choice when fusing small number of modalities. Thus we take linear fusion model as one of our text-visual-AIA models. Since our proposed visual-AIA model in Section 5.4.2 and TBM achieve the best performance on visual and text features respectively, we would like to fuse their likelihoods by the followed formulation:

$$p(I | c_i) = (1 - a) \times p_{BHMMM}(I_b | c_i) + a \times p_{TBM}(I_c | c_i) \quad (5.13)$$

Here, $p(I | c_i)$ denotes the final likelihood of generating an image I from the concept class c_i . $p_{BHMMM}(I_b | c_i)$ denotes the likelihood of generating I_b from c_i based on our

proposed BHMMM which is taken as our visual-AIA model in Section 5.3 , and I_b is the image representation of region tokens. $p_{TBM}(I_c | c_i)$ denotes the likelihood of generating I_c from c_i based on our proposed TBM model in Section 5.4.2. In essence, Eq.(5.13) is an example of score-level fusion. a is a constant used to combine the likelihoods from visual-AIA model and TBM, and the value of a is set empirically as 0.7 in our experiments.

5.5.2 Text and Visual-based Bayesian Model (TVBM)

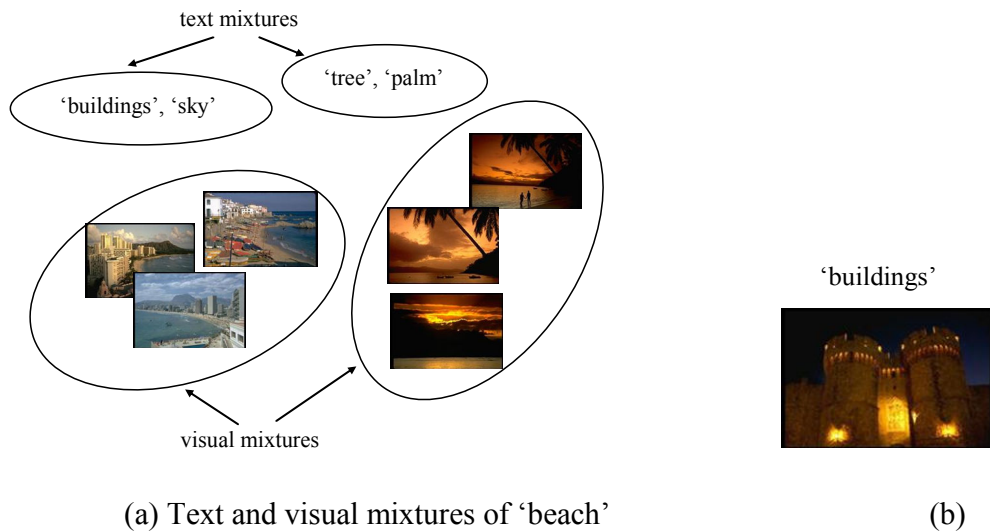


Figure 5.7: An illustration of the dependency between visual and text modalities

Although linear fusion model is an easy way to fuse multi-modal features, and work well when the number of modality is small, linear fusion is not capable of modeling the inter-dependencies between modalities. Figure 5.7 shows an example of illustrating the dependencies between visual and text modalities.

As shown in Figure 5.7a, there are four possible mixtures for the concept class ‘Beach’, two of them are based on text features (i.e. ‘buildings’ and ‘sky’, ‘tree’ and ‘palm’), and the other two are based on visual features. Given an image labeled with ‘buildings’ in the training corpus as shown in Figure 5.7b, it is likely to be chosen as an additional sample for ‘beach’, since it could be supported with high confidence by both the text mixture of ‘buildings’ and ‘sky’ and visual mixture on the right. But this image is not an appropriate additional sample for the concept class ‘beach’, so we need to explore the inter-dependency of text and visual modalities to acquire ‘more appropriate’ training samples.

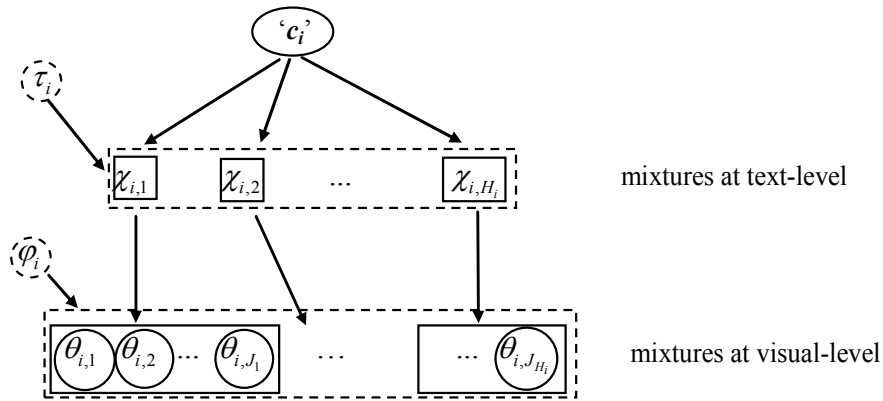


Figure 5.8: An illustration of structure of the proposed text-visual Bayesian model

As shown in Figure 5.8, we model the inter-dependency of text and visual features by building the correspondences between text and visual mixtures. For example, the first text mixture with the parameter $\chi_{i,1}$ corresponds to the visual mixtures with parameter $\theta_{i,1}, \theta_{i,2}, \dots, \theta_{i,J_1}$. Meanwhile, in terms of previous discussions, we also employ the Bayesian approach to estimate the parameters in our proposed model by incorporating the prior knowledge. Thus τ_i and φ_i are the hyperparameters of prior density to

emulate the similar context of text and visual features respectively among sibling concepts.

5.5.3 Parameter Estimation for TVBM

Text features have been shown in the related work to provide an excellent recall performance (Chua et al. 2005, Hauptmann et al. 2006). The reason could be that text features are more useful to find the relevant images than visual features. For example, the results in (Chua et al. 2005) show that more than half of the positive shots in a video corpus can be found by using simple text retrieval. Thus we first estimate the parameters of text mixtures and the hyperparameters $\bar{\tau}_i^{ml}$ by using the same approach for TBM in Section 5.4.3. Then given the estimated parameters $\{\bar{\chi}_{i,h}^{map}\}_{h=1}^{H_i}$, the likelihood of generating I_b from the concept class c_i can be computed as follows:

$$p(I_b | \Lambda_i, I_c) = \sum_{h=1}^{H_i} \left\{ \sum_{j_h=1}^{J_h} [\alpha_{i,j_h} p(I_b | \theta_{i,j_h})] p(I_c | \bar{\chi}_{i,h}^{map}) \right\} \quad (5.14)$$

where $\sum_{h=1}^{H_i} J_h = J$, J is the total number of mixtures at the visual level, and we separate J mixtures into each group of J_h mixtures equally in our experiments. $I_b = (n_1, n_2, \dots, n_L)$ is the image representation based on region tokens as shown in Section 3.1.2. Thus the followed equation is the posterior probability of observing the training samples.

$$p(\Lambda_i | D_i) \propto \left\{ \prod_{t=1}^{|D_i|} \sum_{h=1}^{H_i} \left[\sum_{j_h=1}^{J_h} \alpha_{i,j_h} p(I_{b,t} | \theta_{i,j_h}) p(I_{c,t} | \bar{\chi}_{i,h}^{map}) \right] \right\} * p_0(\Theta_i | \varphi_i) \quad (5.15)$$

Here we still define the prior density p_0 as the Dirichlet distribution, and employ the same approach in Section 4.4.4 to estimate the hyperparameters, φ_i . With the specified hyperparameters $\bar{\varphi}_i^{ml}$, we then impose a MAP criterion to estimate the parameters as follows:

$$\begin{aligned} \bar{\Lambda}_i^{map} &= \arg \max_{\Lambda_i} \log p(\Lambda_i | D_i) \\ &\propto \arg \max_{\Lambda_i} \left\{ \prod_{t=1}^{|D_i|} \sum_{h=1}^{H_i} \sum_{j_h=1}^{J_h} [\alpha_{i,j_h} p(I_{b,t} | \theta_{i,j_h})] p(I_{c,t} | \bar{\chi}_{i,h}^{map}) \right\} * p_0(\Theta_i | \bar{\varphi}_i^{ml}) \end{aligned} \quad (5.16)$$

where $\bar{\varphi}_i^{ml} = (\bar{\varphi}_{i,1}^{ml}, \bar{\varphi}_{i,2}^{ml}, \dots, \bar{\varphi}_{i,L}^{ml})$, $\bar{\varphi}_{i,l}^{ml} > 0, 1 \leq l \leq L$. As discussed in Section 4.5, we will employ the EM algorithm to find the analytical solution of MAP estimation. The two basic equations of the EM algorithm for MAP estimation are as follows:

E-step: For all the $j_h = 1, 2, \dots, J_h$, $h = 1, 2, \dots, J_{H_i}$, and $t = 1, 2, \dots, |D_i|$, we compute

posterior probabilities using the current parameter estimates $\{\alpha_{i,j_h}^{(k)}, \theta_{i,j_h}^{(k)}\}$ at the iteration k .

$$p^{(k)}(j_h | I_{b,t}) = \frac{\alpha_{i,j_h}^{(k)} P(I_{b,t} | \theta_{i,j_h}) p(I_{c,t} | \bar{\chi}_{i,h}^{map})}{\sum_{h=1}^{H_i} \sum_{j_h=1}^{J_h} \alpha_{i,j_h}^{(k)} p(I_{b,t} | \theta_{i,j_h}) p(I_{c,t} | \bar{\chi}_{i,h}^{map})} \quad (5.17)$$

M-step: Updates $\{\alpha_{i,j_h}^{(k+1)}, \theta_{i,j_h}^{(k+1)}, j_h = 1, \dots, J_h\}$, $h = 1, 2, \dots, J_{H_i}$ according to

$$\alpha_{i,j_h}^{(k+1)} = \frac{1}{|D_i|} \sum_{t=1}^{|D_i|} p^{(k)}(j_h | I_{b,t}) \quad (5.18)$$

$$\theta_{i,j_h,l}^{(k+1)} = \frac{\sum_{t=1}^{|D_l|} p^{(k)}(j_h | I_{b,t}) n_{t,l} + (\bar{\varphi}_{i,l}^{ml} - 1)}{\sum_{l=1}^L \sum_{t=1}^{|D_l|} p^{(k)}(j_h | I_{b,t}) n_{t,l} + \sum_{l=1}^L (\bar{\varphi}_{i,l}^{ml} - 1)} \quad (5.19)$$

As compared with the EM algorithm for TBM and BHMMM as shown in Eq. (5.10) and (4.13), the main difference of the EM algorithm for TVBM lies in the computation of the posterior probabilities as shown in Eq. (5.17).

5.5.4 Experiments and Discussions

Now we want to compare the effectiveness of two text-visual-AIA models, namely, (a) LFM (linear fusion model) to merge the two lists of additional sample obtained with visual-AIA and TBM, respectively, and (b) TVBM as discussed in Sections 5.5.2 and 5.5.3. With the same configuration as the experiments in the previous sections, we list the corresponding AIA results in Table 5.4. In contrast with the results in Table 5.4, LFM produced performance similar to what is achieved with TBM. Nevertheless the best results were obtained with TVBM in which the combined text and visual features were used to estimate better models, and consequently better set of additional training samples to training better BHMMM ($J=25$) models for AIA. For example, we achieved the best mean recall of 0.385 among all competing models with TVBM (top 10%). The mean F_1 obtained with the same model is 0.230, which is also the best among all of our experimental results.

Table 5.4: Performance comparison of LFM and TVBM for text -visual-AIA

Models	LFM (top5%)	LFM (top10%)	TVBM (top5%)	TVBM (top10%)
# of concepts (recall>0)	150	154	161	166
Mean Per-concept metrics on all 263 concepts on the Corel dataset				
Mean Precision	0.157	0.163	0.181	0.190
Mean Recall	0.288	0.302	0.363	0.385
Mean F ₁	0.183	0.190	0.218	0.230

In cases when the number of training samples is limited for some concept classes, we expect the fusion models to improve over the baseline results. This can be demonstrated by using the models in Table 5.5, and testing them on the subset of text concept classes with less than 21 training samples defined in the previous Sections. The corresponding results with the two fusion models are listed in Table 5.5. Comparing with the results in Table 4.3, it is clear that both LFM and TVBM provided much better results as compared to those obtained with the baseline BHMMM ($J=25$) without incorporating the extra training samples. Furthermore the improvement from BHMMM ($J=25$) to TVBM (top 10%) was very significant, from a mean recall of 0.106 to 0.320. In the mean time, the mean F₁ was improved from 0.069 to 0.162, for the set of 132 concept classes that have less than 21 training samples.

Table 5.5: Performance summary of LFM and TVBM on the concept classes with small number of training examples

Model	LFM (top5%)	LFM (top10%)	TVBM (top5%)	TVBM (top10%)
# of concepts (recall>0)	44	47	50	52
Mean Per-concept metrics on all 132 concepts on the Corel dataset (# of training samples≤21)				
Mean Precision	0.102	0.109	0.118	0.129
Mean Recall	0.221	0.237	0.298	0.320
Mean F ₁	0.121	0.129	0.147	0.162

5.6 Summary

In this chapter, since the initial collection of concept annotations for each image in the training set is usually incomplete, we explore the use of mixture models to generate more concept labels to each image using the same training set of images so that a new set of mixture models can be built with the same image data coupled with an expanded collection of acquired annotations. For primarily labeled images in the training set, the new additional annotations can be obtained with mixture models built from text and visual features. These models can now be used to associate additional concepts to the images and their original set of concept labels by AIA. We called the text-AIA, visual-AIA and combined text-visual-AIA, respectively.

Experimental results on the Corel image dataset showed that the inclusion of more concept annotations with text-AIA, visual-AIA and text-visual-AIA, gave a significant improvement over the results obtained without the additional training annotations. The best results were achieved with the expanded concept labels obtained with TVBM in which both text and visual features are fused to build a joint models for text-visual-AIA. In summary by incorporating the newly acquired annotations and the corresponding samples into the existing training set, we achieved an even better per-concept F_1 of 0.230 over the top results obtained with our proposed BHMMM, LFM and other extended AIA models.

Chapter 6

Annotating and Filtering Web Images

Having discussed how to alleviate potential difficulties resulting from a limited set of training data by our proposed BHMMM in Chapter 4 and extended AIA in Chapter 5, we now explore the use of external data sources (i.e. World Wide Web) to automatically acquire more training samples. We discuss how to annotate web images and filter out low-quality annotations to collect high-quality additional web image samples for training. Our aim is to circumvent the requirements of a large amount of labeled images by resorting to open sources of web images.

6.1 Introduction

With the explosive growth of multimedia information such as images and videos on the internet, the World-Wide Web (WWW) has been a popular external data source for acquiring additional training samples. In particular, some search engines, such as Google, Yahoo and AltaVista, offer a search function for images. These image search engines provide a convenient way for users to search or collect large-scale web images. Different from the traditional image collections that contain very little information, the web images contain many context information like image's filename, ALT-tag and/or associated web pages. Thus to address the problem of effectively annotating a large amount of images from the web and collecting high-quality additional training samples automatically, we need to tackle three key issues: (a) extract appropriate textual hints from the associated

HTML pages of images; (b) fuse the text and visual contents to model the dependencies between them; and (c) due to large variations among web images, we need an effective strategy to check the ‘goodness’ or quality of annotations for web images. Thus in the following Sections, we will discuss these three key issues.

6.2 Extracting Text Descriptions

The text descriptions of a web page often give useful hints on what an embedded image is about. However, textual contents may contain not only information that captures the semantics of the embedded image, but also other descriptions that are not directly relevant to the image. There are several places in a webpage where relevant texts may be found: (a) image file name; (b) page title; (c) alternate text (ALT-tag); and (d) surrounding text. For example, the first three features are employed in (Shen et al. 2000; Zhang and Chen 2002), since these features are easy to extract. However, the empirical studies in (Feng et al. 2004) show that these three features do not often give sufficient semantic information on an image. The image file name is often abbreviated and may not be recognized as meaningful words. The page title may be inaccurate to semantic contents of the embedded image as there is often multiple images or topics in a web page. Moreover, a lot of web images do not even have alternate text.

In order to provide a more complete description of image contents, we need to incorporate relevant surrounding text. However, the great variety in style and web page layout makes the automated extraction of surrounding text a challenging task. This is partly why most existing approaches do not consider surrounding text. Fortunately, there

is regularity to the appearance of relevant surrounding text with respect to the position of an image in an HTML document. For example, relevant surrounding text often appears adjacent to or below an image, or in the table cell next to the one containing the image. An earlier study of over 1,000 web pages (Feng et al. 2004) arrives at the following observations. (a) Surrounding text may appear to the left or right of the image in the HTML document. The probability of finding relevant surrounding text to the right is 73% while that to the left is 27%. (b) According to the survey conducted by Google, the first or last 32 words in the text nearest to an image appear to be most descriptive of the image. So if the text description extracted in the left or right direction is longer than 32 words, we only keep the first 32 words as surrounding text. Further details of our algorithm for finding relevant surrounding text can be found in (Pan 2003).

Given a predefined concept vocabulary $\mathcal{C} = \{c_1, c_2, \dots, c_V\}$ and a bag of keywords derived from the associated HTML pages, we employ only those keywords contained in the concept vocabulary to represent a web image I . We use a vector $I_c = (m_1, m_2, \dots, m_V)$ as the text representation of an image, where V is the total number of the predefined concepts, and $m_v (1 \leq v \leq V)$ denotes the observed count of the v^{th} concept in the bag of keywords, as shown in Section 5.4.

6.3 Fusion Models

In Section 5.5.2, we have discussed how to fuse text and visual features to model the dependencies between them. For web image annotation, most approaches explore the fusion of multi-source evidences by employing either heuristic techniques such as convex

combination or voting scheme (Hauptman et al. 2003; Chaisorn et al. 2003), or the Dempster-Shafer combination technique (Aslandogan and Yu 2000). In essence, these fusion approaches ignore the dependencies between text and visual features. Thus we apply our proposed TVBM model in Section 5.5.2 to fuse the text and visual features and to annotate web images.

The visual representation of web images follows the approach in CMRM (Jeon et al. 2003) and TM (Duygulu et al. 2002), which is the same as the experimental settings in previous Sections. That is to say, each web image is first segmented into regions by BlobWorld and region visual features are computed. A region is assigned a region token whose centroid is the closest to the region in the feature space. As a result, each web image I is represented by an image vector $I_b = (n_1, n_2, \dots, n_L)$, where each element n_l denotes the observed count of the l^{th} corresponding region tokens in the image I as shown in Figure 3.1.

6.4 Annotation Filtering Strategy

Web images often have extensive semantics and large variations on visual contents. Thus we need a strategy to evaluate the ‘goodness’ or quality of newly annotated web image samples. Some approaches evaluate the quality of newly annotated samples by the so-called ‘co-training’ technique. For example, Feng et al. developed two ‘view-independent’ classifiers in (Feng et al. 2004) – one based on text, and the other on visual features. Thus a web image is likely to be chosen as an additional sample if this image is supported by both text and visual classifiers. As discussed in Section 5.5.2, such

approaches ignore the dependencies between text and visual features, and we apply our proposed TVBM to annotate web images. Thus in this section we design two strategies to filter out the low-quality web image samples based on their likelihoods from TVBM.

6.4.1 Top N_P

As far as we know, top N strategy is a common approach to ranking and filtering candidate concept annotations or images in the field of image annotation and retrieval. For the top N , a fixed number of newly annotated images with the highest N ranking values (i.e. likelihoods) are chosen. For example, most existing AIA models (Duygulu et al. 2002; Jeon et al. 2003; Srikanth et al. 2005) assign a set of top five concepts to each test image based on the concept likelihoods, and the model performance is evaluated by comparing the generated annotations with the ground truth of image annotations in the testing set. In (Rui et al. 2007), the strategy of top N is used to filter out the low quality annotations for web images.

In our proposed extended AIA models, a fixed *percentage*, rather than a fixed N , of the newly annotated images with the highest likelihoods are chosen as the additional training samples in our proposed extended AIA models. Here we call this strategy as ‘top N_P ’. Empirically, one can use cross-validation experiments to set appropriate values of a fixed *percentage* (Rui et al. 2007), but in our experiment we simply tried different *percentage* values. The disadvantage of top N or top N_P is that if the number of high-quality images for each concept class varies, it is hard to select an accurate number or percentage for annotations.

6.4.2 Likelihood Measure (LM)

In order to evaluate the ‘goodness’ or quality of annotations for web images, a natural way is to attach to each concept label a number that indicates how confident the AIA model is about accepting this concept label for the given web image. This number is often referred to as a confidence measure, serving as a reference guide to evaluate the quality of new annotations.

In essence, our scenario is a problem of pattern verification. Generally speaking, it is formulated as follows: given a test signal Y , we want to verify if the signal Y is generated from a signal source, S_0 . Thus we need to consider two types of errors. First, one could have decided that Y was not generated from the signal source S_0 , while it was indeed coming from the source. Second, one could have verified the given Y as coming from the signal source S_0 while it was actually generated from a different source. The verification performance is often evaluated as a combination of these two types of errors. From the viewpoint of statistical inference (Duda et al. 2001), the pattern verification scenario is closely related to a *hypothesis testing* problem. That is to say, given the test signal Y , we want to test the null hypothesis H_0 , against the alternative hypothesis, H_1 , where H_0 assumes that Y is generated from the source S_0 , and H_1 assumes that Y is generated from another source S_1 .

Thus based on the above analysis, we perform pattern verification as follows: given a test signal Y , a test statistic $T(Y)$ is formed, and the hypothesis H_0 is accepted if

$$T(Y) \geq \omega \tag{6.1}$$

where ω is a test threshold. A test statistic commonly used is likelihood ratio test (LRT) as shown in Eq. (6.2), which has been adopted as a way to perform speaker and utterance verification (Lee 2001; Jiang 2005) in the field of speech recognition.

$$T(Y) = p(Y | \lambda_0) / p(Y | \lambda_1) \geq \omega \quad (6.2)$$

where λ_0 and λ_1 are model parameters characterizing H_0 and H_1 , respectively, and $p(Y | \lambda_0)$ and $p(Y | \lambda_1)$ are the likelihoods that the test signal Y is generated by the two competing sources, S_0 and S_1 . Based on Eq. (6.2), the approaches of LRT-based confidence measures in speaker and utterance verification focus on how to approximate $p(Y | \lambda_1)$ (Cox and Rose 1996; Gillick et al. 1997; Kemp and Schaaf 1997; Modi and Rahim 1997). Thus a key issue for LRT-based confidence measures is how to find the competing sources S_1 .

A common way to find the competing sources S_1 is the one-against-all criterion. That is to say, given a concept $c_i \in \mathcal{C} = \{c_1, c_2, \dots, c_V\}$, the source S_0 is the training set of c_i , D_i , while the competing source S_1 is the set of training samples from the other concepts, $\{c_1, c_2, \dots, c_V\} / c_i$. However, in our scenario, each image can be labeled with multiple concepts and the original annotations are often incomplete, so such a binary criterion is not appropriate. Furthermore, one-against-all criterion could often result in the very large size of competing source S_1 as compared with the size of source S_0 . Therefore we simplify Eq. (6.2) by only considering the likelihood distribution of source S_0 as shown in Eq. (6.3)

$$T(Y) = p(Y | \lambda_0) \geq \omega \quad (6.3)$$

Thus given a concept c_i and model parameter Λ_i , a test statistic $T(Y)$, can be rewritten as follows, and the hypothesis H_0 is accepted if

$$T(Y) = P(Y | \Lambda_i) \geq \omega_i \quad (6.4)$$

We call our strategy in Eq. (6.4) as likelihood measure (LM). In this strategy, we set the threshold ω_i for a concept class c_i adaptively in terms of the likelihood distribution over its training set. The basic idea is that if the likelihood of an additional web image is consistent with or no less than the likelihood values of most training samples, then we can trust and accept this additional sample.

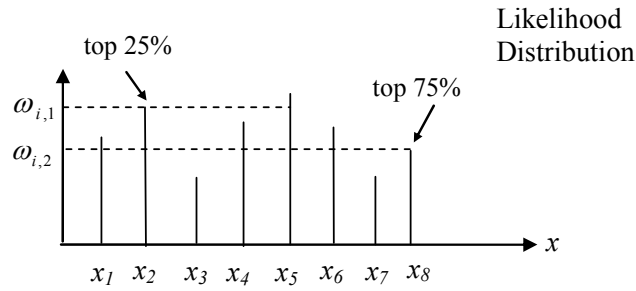


Figure 6.1: Likelihood measure

Compared with the ‘top N_P ’ strategy, LM does not set a fixed percentage number of the annotated images with highest likelihoods for all the concept classes, but set an adaptive threshold for each concept class according to the likelihood distribution of the training set. In our experiment we want to investigate how the different ω values affect the performance of confidence measure. Figure 6.1 shows an example of likelihood distribution over the training set of a concept class c_i . $\omega_{i,1}$ corresponds to the threshold of

highest likelihood for top 25% training samples, and $\omega_{i,2}$ corresponds to the threshold of highest likelihood for top 75% training samples.

6.5 Experiments and Discussions

In this Section, we want to compare the annotation performance of top N_P and LM for web images. We still use the Corel CD dataset, including 4500 training images and 500 testing images, to verify these two strategies. The experimental settings are the same as the other experiments in this dissertation, but we mainly focus on the concept classes in which the number of original training samples is less than 21, and the total number of such concept classes is 132.

6.5.1 Crawling Web Images

We use Google Image Search to collect additional web images. Given a concept c_i , we first find the co-occurred words from its training set. Here we denote the set of co-occurred words of c_i as cw_i , where $cw_i \subseteq \mathcal{C} = \{c_1, c_2, \dots, c_V\}$, and any concept $c_j \in cw_i$ $i \neq j$, $1 \leq j \leq V$. So the pair of concepts (c_i, c_j) ($c_j \in cw_i$) is labeled for at least one training image of concept class c_i . Then we submit the one word c_i and the word pairs (c_i, c_j) ($c_j \in cw_i$) as queries to Google Image Search. Finally we retrieve top 20 resulting web images for each query as the candidates for additional training samples (Liu et al. 2007). For example, if we want to collect additional web images for the concept ‘tiger’, then we submit five queries such as ‘tiger’, ‘tiger’ and ‘water’, ‘tiger’ and ‘grass’, ‘tiger’ and ‘tree’ to Google Image Search, thus we have a total of 100

additional web images for the concept class ‘tiger’. In our experiment we crawled a total of 14,726 web images for 132 concept classes by Google Image Search.

6.5.2 Pipeline

As shown in Section 5.6.3, TVBM (top 10%) achieves the best performance both on the 263 concepts of the whole testing set and on the 132 concepts with small number of training samples, thus we employ the trained TVBM (top 10%) to annotate and filter the candidate web images. Given a concept class c_i , we denote the original training set as D_i , the additional training set obtained by TVBM as $S_{tv}^{(i)}$ and the set of selected web images as $S_{web}^{(i)}$, then we have the following algorithm to annotate and filter out the web images:

<p>Input: The set of training images D_i for a given concept class c_i;</p> <p>Output: The estimated model parameters with MAP criterion, $\bar{\Lambda}_i^{map}$;</p> <ol style="list-style-type: none"> 1) Given the training set of images, estimate the parameters of TVBM by MAP criterion. 2) For any concept c_i, expand the annotations of images related to c_i by TVBM. 3) Generate a rank list of images for c_i based on their likelihoods. 4) Expand training set of c_i by combining top 10% images ($S_{tv}^{(i)}$) and D_i. 5) Re-estimate the parameters of TVBM for c_i based on the expanded training set. 6) Crawl web images for each concept by using Google Image Search and the expanded set of queries as explained in Section 6.5.1. 7) Apply the re-estimated TVBM to compute the likelihoods of additional web images. 8) Set fixed percentage or the threshold ω_i to filter out the low-quality web images based on the strategy of top N_P or LM, and obtain the additional set $S_{web}^{(i)}$. 9) Estimate the parameters of BHMMM ($J=25$) for concept class c_i by combining D_i, $S_{tv}^{(i)}$ and $S_{web}^{(i)}$, to perform conventional AIA.

6.5.3 Experimental Results Using Top N_P

The aim of this experiment is to verify the effectiveness of the top N_P strategy for checking the ‘goodness’ of web images. That is to say, given the list of likelihoods of web candidate images for a concept class, we only select the fixed top *percentage* of candidate images as the additional set $S_{web}^{(i)}$. We set four different values for *percentage*, i.e. at 20%, 25%, 30% and 35%, respectively.

Table 6.1: Performance of TVBM and top N_P strategy

Models	Top 20%	Top 25%	Top 30%	Top 35%	TVBM (top 10%)
# of additional images	3000	3728	4475	5218	--
# of concepts (recall>0)	62	62	62	61	52
Mean Per-concept metrics on the 132 concepts on the Corel dataset (#. of original training samples of each concept class ≤ 21)					
Mean Precision	0.242	0.246	0.236	0.226	0.190
Mean Recall	0.371	0.374	0.369	0.357	0.385
Mean F1	0.264	0.269	0.258	0.253	0.230

With the same configuration as the experiments in previous sections, we list the corresponding conventional AIA results in Table 6.1. From Table 6.1, we can draw the following observations. (a) Compared with the results of TVBM (top 10%) from Table 5.5, top N_P based methods achieve better performance in terms of mean precision, recall and mean F1. This indicates that WWW is useful to provide the additional training images for users and most of the selected web images are positive so that the final performance can be improved. (b) It is clear that the top 25% achieves the best performance, but the performance of top 30% and 35% degrade continuously, which indicates that more noisy web images are accepted as the additional training samples

when we set the threshold to top 30% and 35%. Obviously this highlights the disadvantage of top N_P strategy – that if the number of high-quality images for each concept class varies, it is hard to select an accurate number of annotations.

6.5.4 Experimental Results Using LM

This Section analyzes the effectiveness of LM for selecting high quality web images for training. That is to say, given the list of likelihoods of training set for a concept class, we set the threshold to filter out the web images in terms of the top likelihoods of training samples. We set four thresholds ω in terms of different top likelihoods of training samples i.e. at 25%, 50%, 75% and 100%, respectively (see Figure 6.1).

Table 6.2: Performance of LM with different thresholds

Models	LM (top 25%)	LM (top 50%)	LM (top 75%)	LM (top 100%)
# of additional images	753	2136	3512	4557
# of concepts (recall>0)	60	67	69	65
Mean Per-concept metrics on the 132 concepts on the Corel dataset (#. of original training samples of each concept class ≤ 21)				
Mean Precision	0.1975	0.240	0.255	0.247
Mean Recall	0.370	0.416	0.449	0.407
Mean F1	0.2290	0.269	0.291	0.277

With the same configuration as the experiments in the previous sections, we tabulate the corresponding conventional AIA results in Table 6.2. From Table 6.2, we can draw the following observations. (a) Compared with the results in Table 6.1, the LM with higher thresholds (top 50%, 75% and 100%) achieve better performance in terms of mean F1. In particular, LM with top 75% achieves the best performance in terms of all

measurements. This indicates that LM is more effective to filter out low-quality additional web images than the top N_P strategy. (b) The performance of LM with top 100% is a little worse than that of LM with top 75%. This indicates that not all the original annotations could be correct, which leads to some noisy web images included into the additional set $S_{web}^{(i)}$.

6.5.5 Refinement of Web Image Search Results

The previous two experiments have demonstrated the effectiveness of crawling additional web images refined by TVBM and various filtering strategies to support AIA task. This seems to indicate that TVBM can be used as a method to improve the retrieved results of web image search. In particular, top N_P (25%) and LM (top 75%) achieved the best performance in their respective group. Thus in order to validate this claim, we tabulate the performance of Google Image Search, top N_P (25%) and LM (top 75%) in Table 6.3 by manually checking the retrieved results obtained by Google Image Search and the refined results obtained by TVBM coupled with the strategies of top N_P and LM.

Table 6.3: Performance comparison of top N_P and LM for refining the retrieved web images

Models	Google Image Search	Top N_P (25%)	LM (top 75%)
Mean Precision	0.55	0.62	0.76

In the absence of ground truth for the retrieved web images, we manually estimate the performance of the retrieval in terms of precision based on the 132 concepts on the Corel dataset. As shown in Table 6.3, TVBM coupled with top N_P (25%) and LM (top 75%)

is able to improve the precision of the original Google Image Search results by 0.62 and 0.76 respectively. This indicates that TVBM could be used as an effective model to refine the results of web image search. In particular, TVBM coupled with LM (top 75%) achieved the best precision of 0.76, with an improvement of about 21% over the original retrieved results by Google Image Search.

6.5.6 Top N_P vs. LM

As discussed in Section 6.4, LM can set an adaptive threshold for each concept class to filter out low-quality web images. To compare the effectiveness of two strategies on top N_P and LM in detail, we further analyze the results by splitting the set of 132 concepts into two groups, i.e. I and II. In group I, LM select less additional web images for each concept than top N_P , while LM select more additional web images for each concept in group II than top N_P . As a result, there are a total of 70 concepts in group I, and 62 concepts in group II. We hope that LM is more adaptive for setting thresholds than top N_P in both groups. Since the top N_P (25%) and LM (top 75%) achieved the best performance in their respective experiments, we compare their performance in group I and II.

Table 6.4: Performance comparison of top N_P and LM in Group I

Models	Top N_P (25%)	LM (top 75%)
# of additional images	2534	1944
# of concepts (recall>0)	28	32
Mean Per-concept metrics on the 70 concepts in group I (#. of original training samples of each concept class ≤ 21)		
Mean Precision	0.193	0.204
Mean Recall	0.310	0.422
Mean F1	0.218	0.255

We first compared the performance of the top N_P and LM on the concepts in group I. As shown in Table 6.4, more additional web images are obtained by the top N_P (25%). But the performance of the top N_P is worse than that of LM in terms of the recall, precision and F_1 . This indicates that many noisy additional web images are incorporated by the top N_P but not as many by LM. Figure 6.2 shows some noisy additional web image samples of the concepts in group I obtained by the top N_P but not by LM.













Face			
Lake			
Hillside			
Caribou			

Figure 6.2: Some negative additional samples obtained from top N_P

We then compared the performance of the top N_P and LM on the concepts in group II. As shown in Table 6.5, fewer additional web images are obtained by the top N_P (25%). However, the performance of the top N_P strategy is still worse than that of LM in terms of the recall, precision and F_1 . This indicates that some positive additional web

images are incorporated by LM but not by the top N_P strategy. Figure 6.3 shows some examples of positive additional web image samples of the concepts in group II obtained by LM but not by the top N_P .

Table 6.5: Performance comparison of top N_P and LM in Group II

Models	Top N_P (25%)	LM (top 75%)
# of additional images	1194	1568
# of concepts (recall>0)	34	37
Mean Per-concept metrics on the 62 concepts in group II (#. of original training samples of each concept class ≤ 21)		
Mean Precision	0.306	0.312
Mean Recall	0.446	0.480
Mean F1	0.326	0.332

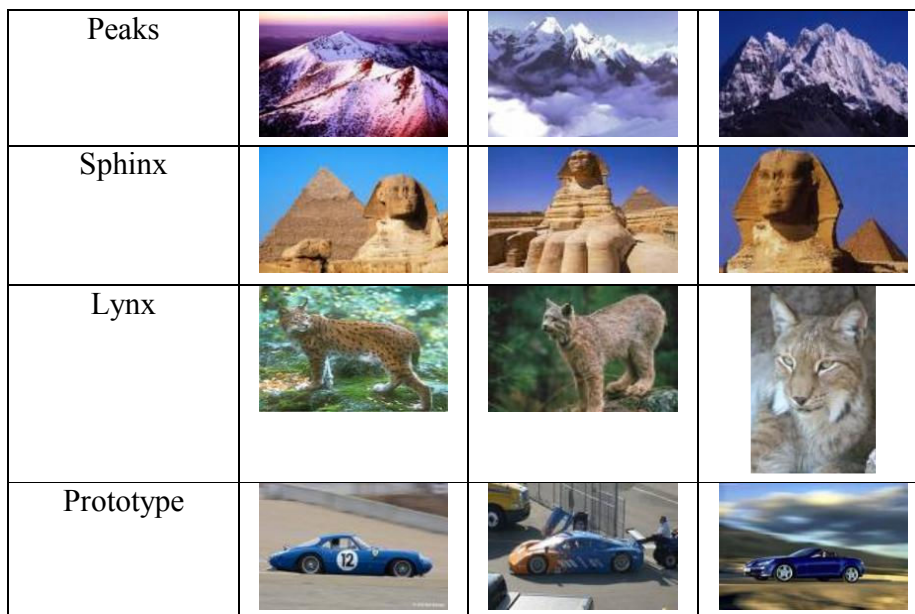


Figure 6.3: Some positive additional samples obtained from LM

6.5.7 Overall Performance

In this experiment, we want to verify the overall performance on the full 263 testing concept classes of the Corel dataset. Here we incorporate all the additional images obtained from TVBM and LM (TVBM (top 10%) + LM (top 75%)) for effective training of BHMMM ($J=25$) which is used for conventional AIA.

Table 6.6 tabulates the performance of MH (MH = Mixture Hierarchy (Carneiro et al. 2007)), and TVBM (top 10%). From the Table, it is clear that our strategy of TVBM (top 10%) + LM (top 75%) achieves the best performance in terms of the recall, precision and F1 measurements. In particular, the recall measurement is significantly improved from 0.290 to 0.458, which is the best recall performance on this dataset as compared with all reputed state-of-the-art AIA models.

Table 6.6: Overall performance

Models	MH	TVBM (top 10%)	TVBM (top 10%) + LM (top 75%)
# of concepts (recall>0)	137	166	186
Mean Per-concept metrics on the 263 concepts on the Corel dataset			
Mean Precision	0.230	0.190	0.248
Mean Recall	0.290	0.385	0.458
Mean F1	--	0.231	0.298

6.6 Summary

In this chapter, we discussed the problem of annotating and filtering the web images. We first downloaded the web images by Google Image Search, and then applied our proposed TVBM to annotate these web images. We presented two strategies for filtering out the

low-quality web images, top N_P strategy and LM. Experimental results on the Corel image dataset show that the inclusion of web images as additional training samples gives a significant improvement over the results obtained without using additional web images. The best results were achieved with the LM (top 75%), which indicates that the LM is more effective for filtering out the low-quality web images than the top N_P strategy. In summary, by incorporating the newly acquired image samples from the internal dataset as well as the external dataset from the web into the existing training set, we achieved the best per-concept precision of 0.248 and per-concept recall of 0.458, as compared to all reputed AIA models.

Chapter 7

Conclusions and Future Work

In this chapter, we first summarize the work presented in this dissertation based on our proposed Bayesian learning framework used to alleviate the potential problems arising from the limited size of training samples, including Bayesian Hierarchical Multinomial Mixture Model (BHMMM), Extended AIA models (i.e. TBM and TVBM) and applying TVBM for web image annotation coupled with the likelihood measure. We then discuss some work that we plan to pursue in the near future.

7.1 Conclusions

In this dissertation, we circumvented the potential problems arising from the limited size of labeled training images by proposing a Bayesian learning framework. The framework includes three key aspects: 1) incorporating prior knowledge of concept ontology to improve the maximum-likelihood estimations of model parameters; 2) effectively expanding the original annotations of training images based on multi-modality analysis to acquire more training samples without collecting new images; and 3) resorting to open image sources on the web for new additional training images. Thus we summarize our conclusions in the following sections.

7.1.1 Bayesian Hierarchical Multinomial Mixture Model

We proposed BHMMM to enhance the maximum likelihood estimate of our baseline model (multinomial mixture model) by incorporating prior knowledge into hierarchical representation of concepts, since the ML estimates in our baseline model depend heavily on a large set of labeled training images. The formulation of BHMMM facilitates a statistical combination of the likelihood function of the available training data and the prior density of the concept parameters into a well-defined posterior density, by treating the mixture model parameters as random variables characterized by a joint conjugate prior density. The model parameters can then be estimated via a maximum a posteriori criterion.

Experimental results on the Corel image dataset showed that the proposed BHMMM approach, using a multi-level concept hierarchy of 371 concepts with a maximum of 25 mixture components per concept, achieves a mean F_1 measure of 0.169, which outperforms our baseline model and many state-of-the-art techniques under the same experimental settings for automatic image annotation. In particular, BHMMM outperforms our baseline model by 0.069 in terms of F_1 measurement on a subset of 132 test concepts in Corel CD dataset in which the number of training samples in each class is no more than 21.

7.1.2 Extended AIA Based on Multimodal Features

We extended the conventional AIA by three modes (visual-AIA, text-AIA and text-visual-AIA) to effectively expand the annotations and acquire more training samples for

each concept class. The advantage of such an approach is that we can augment the training set of each concept class without the need of additional human labeling efforts or collecting additional training images from other data sources. By utilizing the text and visual features from the training set and concept ontology derived from prior knowledge, we employed BHMMM as visual-AIA, and then proposed a text-based Bayesian model (TBM) as text-AIA by extending BHMMM to text modality, and finally proposed a text-visual Bayesian hierarchical multinomial mixture model (TVBM) as text-visual-AIA.

Experimental results on the Corel image dataset showed that the inclusion of more concept labels with text-AIA, visual-AIA and text-visual-AIA, gives a significant improvement over the results obtained without the additional training labels. The best results were achieved with the expanded concept labels obtained with TVBM in which both text and visual features are fused to build a joint models for text-visual-AIA. In summary by incorporating the newly acquired annotations and the corresponding samples into the existing training set, we achieved an even better per-concept F_1 of 0.230 over the top results of 0.169 obtained with our proposed baseline BHMMM.

7.1.3 Likelihood Measure for Web Image Annotation

Nowadays, images have become widely available on the World Wide Web (WWW). Different from the traditional image collections where very little information is provided, the web images tend to contain a lot of contextual information like surrounding text and links. Thus we want to annotate web images to collect additional samples for training. However, due to large variations among web images, we focused on finding an effective strategy to check the ‘goodness’ of annotations for additional web images. We first

applied our proposed TVBM to annotate web images. Given the likelihoods of web images, we investigated two strategies to check the ‘goodness’ of additional annotations for web images, i.e. top N_P and likelihood measure. Compared with the strategy of fixed top N_P , LM can set an adaptive threshold for each concept class as a confidence measure to select the additional web images in terms of the likelihood distributions of the training samples.

Based on a subset of 132 testing concepts in Corel CD dataset in which the number of training samples in each class is no more than 21, experimental results showed that the inclusion of web images as additional training samples gave a significant improvement over the results obtained without using additional web images. The best results were achieved with the LM (top 75%). In particular, by incorporating the newly acquired image samples from the internal dataset (TVBM (top 10%)) and the external dataset from the web LM (top 75%) into the existing training set, we achieved the best per-concept precision of 0.248 and per-concept recall of 0.458. This result is far superior to those of state-of-the-arts AIA models as reputed in Table 2.1.

7.2 Future Work

Automatic image annotation is a challenging task. While this thesis proposed a novel framework to tackle several important aspects of this problem, there are necessarily gaps to be bridged in the framework that should be addressed in the future. In the following, we discuss some work that we are going to pursue in the near future.

1. Image Content Representations. To represent image contents, in this dissertation we first segmented the images into regions and then clustered all the regions into some region clusters which are the so-called ‘region tokens’. The main advantage of such methods is that we can construct a limited size of region token vocabulary to cover all the image variations in the space of visual features. However, the above clustering methods could lead to poor clustering performance if we were to use only visual features of regions as basis for clustering. This is because the regions with different semantic concepts but share similar appearance may be easily grouped together. Thus such clustering methods were improved by using the annotations of training images to impose additional semantic pair-wise constraints when clustering the regions (Jin et al. 2004; Shi et al. 2005; Yang et al. 2007). Recently research on clustering (Wagstaff 2001; Yan and Hauptman 2004) showed that clustering with pair-wise constraints, a kind of realistic semi-supervised clustering method, performs considerably better than the unconstrained methods. But it is still an open research problem on how to improve the traditional clustering methods by incorporating the constraints and how to assign the region token to a segmented region based only on visual features.

2. Statistical Confidence Measures. To check the ‘goodness’ of annotations for additional web images, we formulated our problem as a *hypothesis testing problem*, and simplified LRT-based confidence measures (Lee 2001; Jiang 2005) to our likelihood measure by only considering the likelihood distribution of source S_0 and ignoring the competing source S_1 . As presented in Section 6.4.2, there is some research work on LRT-based confidence measures in the field of speaker and

utterance verification (Cox and Rose 1996; Gillick et al. 1997; Kemp and Schaaf 1997; Modi and Rahim 1997). But these approaches employed a binary way to find the competing source S_i , which is not appropriate in our scenario due to incomplete original annotations. So we are interested in studying a new scheme in which we can perform confidence measures adaptively. For example, given the images with incomplete annotations, we can first expand the annotations with likelihood measure. With increase number of annotations, we could assume that the annotations are more complete and then apply the LRT-based confidence measures to filter out low-quality image annotations.

3. Refinement of web image search. As discussed in Chapter 6, the performance of the current image search engines is not very good due to the lower ranks of some relevant retrieved images. Thus we need to develop an effective model to refine the retrieved results. In Section 6.5.5, we demonstrated that our proposed TVBM coupled with LM (top 75%) was able to achieve the best performance with an improvement of about 21% in precision over the original Google Image Search results. Therefore, we will further investigate the problems of scalability and speed of online interactive use which will be an important area of our future research work.

4. TRECVID video dataset. TRECVID video dataset is a large-scale video collection available on TREC video forum (Chua et al. 2005, Hauptmann et al. 2006). For the video data, we have more features to describe the video contents, such as visual, text, audio, motion, face detection and recognition, etc. Obviously the dependencies among multimodal features are more complex than just text and visual features. So

we will consider extending our proposed TVBM to model the complex dependencies among multi-modal features to annotate the video data.

Bibliography

- Y. A. Aslandogan and C. T. Yu (2000). Multiple Evidence Combination in Image Retrieval: Diogenes Searches for People on the Web. *in* Proceedings of ACM Conference on Research and Development in Information Retrieval, pages: 88-95.
- K. Barnard, P. Duygulu and D. Forsyth (2001). Clustering Art. *in* Proceedings of IEEE Conf. on Computer Vision and Pattern Recognition, pages: 434-441.
- K. Barnard and D. A. Forsyth (2001). Learning the Semantics of Words and Pictures. *in* Proceedings of IEEE Intl. Conf. on Computer Vision, pages: 408-415.
- L. Bauer (2003). Introducing Linguistic Morphology (2nd edition). Georgetown University Press.
- F. Bergeaud and S. Mallat (1995). Matching Pursuits of Images. *in* Proceedings of IEEE Intl. Conf. on Image Processing, pages: 53-56.
- J. Besag (1974). Spatial Interaction and the Statistical Analysis of Lattice System. *Journal of the Royal Statistical Society* 36:192-236.
- D. Blei and M. Jordan (2003). Modeling Annotated Data. *in* Proceedings of ACM Conference on Research and Development in Information Retrieval, pages: 127-134.
- O. Bousquet, S. Boucheron and G. Logosi (2004). Introduction to Statistical Learning Theory. *Advanced Lectures on Machine Learning, Lecture Notes in Artificial Intelligence* 3176, pages: 169-207.
- S. Brandt (1999). Use of Shape Features in Content-Based Image Retrieval. Ph.D. Dissertation. Department of Engineering Physics and Mathematics, Helsinki University of Technology.
- P. Carbonetto, H. Kueck and N. Freitas (2004). A Constrained Semi-Supervised Learning Approach to Data Association. *in* Proceedings of European Conference on Computer Vision, pages: 1-12.

- G. Carneiro and N. Vasconcelos (2007). Supervised Learning of Semantic Classes for Image Annotation and Retrieval. *IEEE Transaction on Pattern Analysis and Machine Intelligence* 29(3): 394-410.
- C. Carson, M. Thomas, S. Belongie, J. M. Hellerstein and J. Malik (1999). BlobWorld: A System for Region-Based Image Indexing and Retrieval. *in Proceedings of Intl. Conf. on Visual Information Systems*, pages: 509-516.
- C. Carson, S. Belongie, H. Greenspan and J. Malik (2002). Blobworld: Image Segmentation Using Expectation-Maximization and Its Application to Image Querying. *IEEE Transaction on Pattern Analysis and Machine Intelligence* 24(8): 1026-1038.
- L. Chaisorn, T. S. Chua, C. K. Koh, Y. L. Zhao, H. X. Xu, H. M. Feng and Q. Tian (2003). TREC 2003 Video Retrieval and Story Segmentation Task at NUS PRIS. [Online] http://www-nlpir.nist.gov/projects/tvpubs/tvpapers03/nus.final_paper.pdf.
- N. S. Chang and K. S. Fu (1980). A Query-by Pictorial Example. *IEEE Transaction on Software Engineering* 6:519-524.
- S. F. Chang (2002). The Holy Grail of Content-based Media Analysis. *IEEE Multimedia* 9(2): 6-10.
- S. K. Chang and A. Hsu (1992). Image Information Systems: Where We Go from Here. *IEEE Transaction on Knowledge and Data Engineering* 4:441-442.
- M. Y. Chen and A. Hauptmann (2004). Multi-modal Classification in Digital News Libraries. *in Proceedings of Joint IEEE Conf. on Digital Libraries*, pages: 212-213.
- V. Cherkassky and F. Mulier (1998). *Learning From Data: Concepts, Theory and Methods*. New York: Wiley.
- T. S. Chua, S. K. Lim and H. K. Pung (1994). Content-Based Retrieval of Segmented Images. *in Proceedings of ACM Intl. Conf. on Multimedia*, pages: 211-218.
- T. S. Chua, K. L. Tan and B. C. Ooi (1997). Fast Signature-Based Color-Spatial Image Retrieval. *in Proceedings of IEEE Intl. Conf. on Multimedia Computing and Systems*, pages: 362-369.

- T. S. Chua and C. X. Chu (1998). Color-Based Pseudo Object Model for Image Retrieval with Relevance Feedback. *in* Proceedings of 1st Intl. Conf. on Advance Multimedia Content Processing, pages: 145-160.
- T. S. Chua, C. X. Chu and M. KanKanhalli (1999). Relevance Feedback Techniques for Image Retrieval Using Multiple Attributes. *in* Proceedings of IEEE Intl. Conf. on Multimedia Computing and Systems, pages: 890-894.
- T. S. Chua, S. Y. Neo, H. K. Goh, M. Zhao, Y. Xiao and G. Wang (2005). TRECVID 2005 by NUS PRIS. [Online] Available: <http://www-nlpir.nist.gov/projects/tvpubs/tv5.papers/nus.pdf>.
- J. M. Coggins and A. K. Jain (1985). A Spatial Filtering Approach to Texture Analysis. *Pattern Recognition Letters* 3:195-203.
- S. Cox and R. C. Rose (1996). Confidence Measures for the Switchboard Database. *in* Proceedings of IEEE Intl. Conf. on Acoustics, Speech and Signal Processing, pages: 511-514.
- S. R. Dalal and W. J. Hall (1983). Approximating Priors by Mixtures of Natural Conjugate Priors. *Journal of the Royal Statistical Society, Series B* 45(2): 278-286.
- A. P. Dempster, N. M. Laird and D. B. Rubin (1977). Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society, Series B*, 39(1): 1-38.
- R. O. Duda, P. E. Hart and D. G. Stork (2001). *Pattern Classification*. New York: Wiley.
- P. Duygulu, K. Barnard, N. Freitas and D. Forsyth (2002). Object Recognition as Machine Translation: Learning a Lexicon for a Fixed Image Vocabulary. *in* Proceedings of European Conference on Computer Vision, pages: 97-112.
- J. P. Eakins and M. E. Graham (1999). *Content-Based Image Retrieval: A Report to the JISC Technology Applications Programme*. Institute for Image Data Research, University of Northumbria.

- P. Enser and C. Sandom (2003) Towards a Comprehensive Survey of the Semantic Gap in Visual Image Retrieval. *in* Proceedings of Intl. Conf. on Image and Video Retrieval, Lecture Notes in Computer Science (LNCS) 2728, pages: 291-299.
- J. P. Fan, H. Z. Luo and Y. L. Gao (2005a). Learning the Semantics of Images by Using Unlabeled Samples. *in* Proceedings of IEEE Conf. on Computer Vision and Pattern Recognition, vol. 2, pages: 704-710
- J. P. Fan, H. Z. Luo and M. S. Hacid (2005b). Mining Images on Semantics via Statistical Learning. *in* Proceedings of ACM Intl. Conf. on Knowledge Discovery in Data Mining, pages: 22-31.
- H. M. Feng and T. S. Chua (2004). A Learning-Based Approach for Annotating Large On-Line Image Collection. *in* Proceedings of IEEE Intl. Conf. on Multimedia Modeling, pages: 249-256.
- H. M. Feng, R. Shi and T. S. Chua (2004). A Bootstrapping Framework for Annotating and Retrieving WWW Images. *in* Proceedings of ACM Intl. Conf. on Multimedia, pages: 960-967.
- S. L. Feng, R. Manmatha and V. Lavrenko (2004). Multiple Bernoulli Relevance Models for Image and Video Annotation. *in* Proceeding of the IEEE Conf. on Computer Vision and Pattern Recognition, vol. 2, pages: 1002-1009.
- R. Fergus, P. Perona and A. Zisserman (2003). Object Class Recognition by Unsupervised Scale-invariant Learning. *in* Proceedings of IEEE Conf. on Computer Vision and Pattern Recognition, vol. 2, pages: 264-271.
- M. A. T. Figueiredo and A. K. Jain (2002). Unsupervised Learning of Finite Mixture Models. *IEEE Transaction on Pattern Analysis and Machine Intelligence* 24(3): 381-396.
- M. Flickner, H. Sawhney, W. Niblack *et al.* (1995). Query by Image and Video Content: The QBIC System. *IEEE Computer Magazine* 28: 23-32.
- D. Forsyth and M. Fleck (1997). Body Plans. *in* Proceedings of IEEE Conf. on Computer Vision and Pattern Recognition, pages: 678-683.

- B. Furht (1998). *The Handbook of Multimedia Computing: Chapter 11 - Content- Based Image Indexing and Retrieval*. Boca Raton, FL: CRC Press LLC.
- S. Gao, D. H. Wang and C. H. Lee (2006). Automatic Image Annotation through Multi-Topic Text Categorization. *in Proceedings of IEEE Intl. Conf. on Acoustics, Speech and Signal Processing*, pages: 377-380.
- J. L. Gauvain and C. H. Lee (1994). Maximum a Posterior Estimation for Multivariate Gaussian Mixture Observations of Markov Chains. *IEEE Transactions on Speech and Audio Processing* 2(2): 291-298
- A. Gelman, J. B. Carlin, H. S. Stern and D. B. Rubin (2003). *Bayesian Data Analysis* (2nd edition). Boca Raton, FL: Chapman and Hall/CRC Press.
- L. Gillick, Y. Itou and J. Young (1997). A Probabilistic Approach to Confidence Estimation and Evaluation. *in Proceedings of IEEE Intl. Conf. on Acoustics, Speech and Signal Processing*, pages: 879-882.
- N. Haering, Z. Myles and N. Lobo (1997). Locating Deciduous Trees. *in Proceedings of Workshop in Content-Based Access to Image and Video Libraries*, pages: 18-25.
- J. Hafner, H. S. Sawhney, W. Equitz, M. Flickner and W. Niblack (1995). Efficient Color Histogram Indexing for Quadratic Form Distance Functions. *IEEE Transaction on Pattern Analysis and Machine Intelligence* 17(7):729-736.
- R. Hall (1989). *Illumination and Color in Computer Generated Imagery*. New York: Springer-Verlag.
- T. Hastie and R. Tibshirani (1996). Discriminant Analysis by Gaussian Mixtures. *Journal of the Royal Statistical Society* 58:155-176.
- T. Hastie, R. Tibshirani and J. Friedman (2001). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Series in Statistics.
- A. Hauptmann, R. V. Baron, M.-Y Chen *et al.* (2003). Informedia at TRECVID 2003: Analyzing and Searching broadcast news video. [Online] Available: <http://www-nlpir.nist.gov/projects/tvpubs/tvpapers03/cmu.final.paper.pdf>.

- A. Hauptmann, M. Y. Chen, M. Christel, W. H. Lin, R. Yan J. Yang (2006). Multi-Lingual Broadcast News Retrieval. [Online] Available: <http://www-nlpir.nist.gov/projects/tvpubs/tv.pubs.org.html>.
- G. Hinton, P. Dayan and M. Revow (1997). Modeling the Manifolds of Images of Handwritten Digits. *IEEE Transactions on Neural Networks* 8:65-74.
- W. Hsu, T. S. Chua and H. K. Pung (1995). Integrated Color-spatial Approach to Content-based Image Retrieval. *in Proceedings of ACM Intl. Conf. on Multimedia*, pages: 305-313.
- J. Huang, S. R. Kumar, M. Mitra, W. J. Zhu and R. Zabih (1997). Image Indexing Using Color Correlograms. *in Proceeding of IEEE Conf. on Computer Vision and Pattern Recognition*, pages: 762-768.
- J. Huang (2005). Maximum Likelihood Estimation of Dirichlet Distribution Parameters. CMU Technique Report.
- Q. Huo, C. Chan and C. H. Lee (1995). Bayesian Adaptive Learning of the Parameters of Hidden Markov Model for Speech Recognition. *IEEE Transaction on Speech Audio Processing* 3:334-345.
- A. K. Jain and F. Farrokhnia (1991). Unsupervised Texture Segmentation Using Gabor Filters. *Pattern Recognition* 24:1167-1186.
- A. K. Jain, R. P. W. Duin and J. Mao (2000). Statistical Patter Recognition: A Review. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22:4-37.
- R. Jain, R. Kasturi and B. Schunck (1995). *Machine Vision*. New York: MIT Press.
- H. Jiang (2005). Confidence Measures for Speech Recognition: A Survey. *Speech Communication* 45(4): 455-470.
- J. Jeon, V. Lavrenko and R. Manmatha (2003). Automatic Image Annotation and Retrieval Using Cross-Media Relevance Models. *in Proceedings of ACM Conference on Research and Development in Information Retrieval*, pages: 119-126.

- W. J. Jin, R. Shi and T. S. Chua (2004). A Semi-Naïve Bayesian Method Incorporating Clustering with Pair-Wise Constraints for Auto Image Annotation. *in* Proceedings of ACM Intl. Conf. on Multimedia, pages: 336-339.
- T. Kemp and T. Schaaf (1997). Estimating Confidence Using Word Lattices. *in* Proceedings of EuroSpeech, pages: 827-830.
- V. Lavrenko, R. Manmatha and J. Jeon (2003). A Model for Learning the Semantics of Pictures. *in* Proceedings of Neural Information Processing Systems, pages: 408-415.
- C. H. Lee, F. K. Soong and K. K. Paliwal (1996). Automatic Speech and Speaker Recognition: Advanced Topics. Kluwer Academic Press.
- C. H. Lee and Q. Huo (2000). On Adaptive Decision Rules and Decision Parameter Adaptation for Automatic Speech Recognition. *in* Proceedings of the IEEE 88(8):1241-1269.
- C. H. Lee (2001). Statistical Confidence Measures and Their Applications. *in* Proceedings of Intl. Conf. on Signal Processing, pages: 1021-1028.
- Y. Li and L. Shapiro (2002). Consistent Line Clusters for Building Recognition in CBIR. *in* Proceedings of IEEE Intl. Conf. on Pattern Recognition, pages: 952-956.
- J. Liu, B. Wang, M. J. Li, Z. W. Li, W. Y. Ma, H. Q. Lu and S. Ma (2007). Dual Cross-Media Relevance Model for Image Annotation. *in* Proceedings of ACM Intl. Conf. on Multimedia, pages: 605-614.
- D. G. Lowe (2004). Distinctive Image Features from Scale-invariant Keypoints. *International Journal of Computer Vision* 60(2):91-110.
- S. G. Mallat and Z. F. Zhang (1993). Matching Pursuits with Time-frequency Dictionaries. *IEEE Transaction on Signal Processing* 41:3397-3415.
- B. S. Manjunath and W. Y. Ma (1997). Image Indexing Using a Texture Dictionary. *in* Proceedings of SPIE Storage and Retrieval for Image and Video Databases, pages: 288-296.

- B. S. Manjunath, J. R. Ohm, V. V. Vasudevan and A. Yamada (2001). Color and Texture Descriptors. *IEEE Transaction on Circuits and Systems for Video Technology* 11(6): 703-715.
- J. Mao and A. K. Jain (1992). Texture Classification and Segmentation Using Multiresolution Simultaneous Autoregressive Models. *Pattern Recognition* 25:173-188.
- S. Marshall (1989). Review of Shape Coding Techniques. *International Journal of Image and Vision Computing* 7(4):281-294.
- G. McLachlan and T. Krishnan (1997). *The EM Algorithm and Extensions*. New York: Wiley.
- S. Medasani and R. Krishnapuram (1999). A Comparison of Gaussian and Pearson Mixture Modeling for Pattern Recognition and Computer Vision Applications. *Pattern Recognition Letter* 20:305-313.
- B. M. Mehtre, M. S. Kankanhalli and W. F. Lee (1997). Shape Measures for Content-Based Image Retrieval: A Comparison. *International Journal of Information Processing and Management* 33(3):319-337.
- G. A. Miller, R. Beckwith, C. Fellbaum, D. Gross and K. J. Miller (1990). Introduction to WordNet: an On-line Lexical Database. *International Journal of Lexicography* 3:235-244.
- T. Minka (2003). Estimating a Dirichlet Distribution. CMU Technique Report.
- T. P. Minka (2005). A Statistical Learning/Pattern Recognition Glossary. [Online] Available: <http://research.microsoft.com/~minka/statlearn>.
- P. Modi and M. Rahim (1997). Discriminative Utterance Verification Using Multiple Confidence Measures. *in Proceedings of EuroSpeech*, pages: 103-106.
- F. Monay and D. G. Perez (2003). On Image Auto-Annotation with Latent Space Models. *in Proceedings of ACM Intl. Conf. on Multimedia*, pages: 275-278.
- F. Monay and D. G. Perez (2004). PLSA-based Image Auto-Annotation: Constraining the Latent Space. *in Proceedings of ACM Intl. Conf. on Multimedia*, pages: 348-351.

- Y. Mori, H. Takahashi and R. Oka (2000). Image-to-word Transformation Based on Dividing and Vector Quantizing Images with Words. *International Journal of Computer Vision* 40(2):99-121.
- M. R. Naphade, T. Kristjansson, B. Frey and T. S. Huang (1998). Probabilistic Multimedia Objects (Multijects): A Novel Approach to Video Indexing and Retrieval in Multimedia Systems. *in Proceedings of IEEE International Conference on Image Processing*, pages: 536-540.
- J. Novovicova and A. Malik (2002). Application of Multinomial Mixture Model to Text Document Classification. *Pattern Recognition and Image Analysis, Lecture Notes in Computer Science (LNCS) 2652*, pages: 646-653.
- J. Novovicova and A. Malik (2003). Text Document Classification Using Finite Mixtures. *Research Report to UTIA AVCR*.
- L. X. Pan (2003). Image8: An Image Search Engine for the Internet. Honors Year Project Report. School of Computing, National University of Singapore.
- G. Pass, R. Zabih and J. Millar (1996). Comparing Images Using Color Coherence Vectors. *in Proceedings of ACM Intl. Conf. on Multimedia*, pages: 65-73.
- A. Pentland (1984). Fractal-based Description of Natural Scenes. *IEEE Transaction on Circuits and Systems for Video Technology* 9:661-674.
- A. Pentland, R. W. Picard and S. Sclaroff (1996). Photobook: Content-Based Manipulation of Image Databases. *International Journal of Computer Vision* 18: 233-254.
- R. W. Picard and T. P. Minka (1995). Vision Texture for Annotation. *Multimedia Systems* 3(1):3-14.
- H. Raiffa and R. Schlaifer (1961). *Applied Statistical Decision Theory*. Division of Research, Graduate School of Business Administration, Harvard University.
- M. Rautiainen, T. Ojala and T. Seppanen (2004). Analyzing the Performance of Visual, Concept and Text Features in Content-Based Video Retrieval. *in Proceedings of ACM*

- SIGMM International Workshop on Multimedia Information Retrieval, pages: 197-204.
- J. Rissanen (1978). Modeling by Shortest Data Description. *Journal of Automatica* 14: 465-471.
- J. Rissanen (1989). *Stochastic Complexity in Statistical Inquiry Theory*. World Scientific.
- X. G. Rui, M. J. Li, W. Y. Ma and N. H. Yu (2007). Bipartite Graph Reinforcement Model for Web Image Annotation. *in Proceedings of ACM Intl. Conf. on Multimedia*, pages: 585-594.
- Y. Rui, T. S. Huang and S. F. Chang (1999). Image Retrieval: Current Techniques, Promising Directions and Open Issues. *Journal of Visual Communication and Image Representation* 10(1):39-62.
- N. Sebe, M. S. Lew, X. Zhou, T. S. Huang and E. M. Bakker (2003). The State of the Art in Image and Video Retrieval. *in Proceedings of Intl. Conf. on Image and Video Retrieval, Lecture Notes in Computer Science (LNCS) 2728*, pages: 7-12.
- H. T. Shen, B. C. Ooi and K. L. Tan (2000). Giving Meaning to WWW Images. *in Proceedings of ACM Intl. Conf. on Multimedia*, pages: 39-47.
- R. Shi, H. M. Feng, T. S. Chua and C. H. Lee (2004). An Adaptive Image Content Representation and Segmentation Approach to Automatic Image Annotation. *in Proceedings of Intl. Conf. on Image and Video Retrieval, Lecture Notes in Computer Science (LNCS) 3115*, pages: 545-554.
- R. Shi, W. J. Jin and T. S. Chua (2005). A Novel Approach to Auto-Image Annotation Based on Pair-Wise Constrained Clustering and Semi-Naïve Bayesian Model. *in Proceedings of Intl. Conf. on Multimedia Modeling*, pages: 322-327.
- R. Shi, T. S. Chua, C. H. Lee and S. Gao (2006). Bayesian Learning of Hierarchical Multinomial Mixture Models of Concepts for Automatic Image Annotation. *in Proceedings of Intl. Conf. on Image and Video Retrieval, Lecture Notes in Computer Science (LNCS) 4071*, pages: 102-112.

- R. Shi, C. H. Lee and T. S. Chua (2007). Enhancing Image Annotation by Integrating Concept Ontology and Text-based Bayesian Learning Model. *in Proceedings of ACM Intl. Conf. on Multimedia*, pages: 341-344.
- K. Shinoda and C. H. Lee (2001). A Structural Bayes Approach to Speaker Adaptation. *IEEE Transaction on Speech and Audio Processing* 9(3): 276-287.
- A. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain (2000). Content-Based Image Retrieval: The End of the Early Years. *IEEE Transaction on Pattern Analysis and Machine Intelligence* 22(12):1349-1380.
- J. R. Smith and S. F. Chang (1996). VisualSEEK: A Fully Automated Content-Based Image Query System. *in Proceedings of ACM Intl. Conf. on Multimedia*, pages: 87-98.
- J. R. Smith (1997). Integrated Spatial and Feature Image Systems: Retrieval, Analysis and Compression. Ph.D. Dissertation. Graduate School of Arts and Sciences, Columbia University.
- J. R. Smith, M. R. Naphade and A. P. Natsev (2003). Multimedia Semantic Indexing Using Model Vectors. *in Proceedings of Intl. Conf. Multimedia and Expo*, vol. 2, pages: 445-448.
- M. Srikanth, J. Varner, M. Bowden and D. Moldovan (2005). Exploiting Ontologies for Automatic Image Annotation. *in Proceedings of ACM Conference on Research and Development in Information Retrieval*, pages: 552-558.
- M. A. Stricker and M. Orengo (1995). Similarity of Color Images. *in Proceedings of SPIE Storage and Retrieval for Image and Video Databases III*, pages: 381-392.
- M. Szummer and R. W. Picard (1998). Indoor-Outdoor Image Classification. *in Proceedings of IEEE Intl. Workshop on Content-based Access of Image and Video Databases*, pages: 42-51.
- H. H. Tong, J. R. He, M. J. Li, C. S. Zhang and W. Y. Ma (2005). Graph Based Multi-Modality Learning. *in Proceedings of ACM Intl. Conf. on Multimedia*, pages: 862-871.
- S. Tong and E. Chang (2001). Support Vector Machine Active Learning for Image Retrieval. *in Proceedings of Intl. Conf. on Multimedia*, pages: 107-118.

- M. Tuceryan and A. K. Jain (1990). Texture Segmentation Using Voronoi Polygons. *IEEE Transaction on Pattern Analysis and Machine Intelligence* 12:211-216.
- M. Tuceryan and A. K. Jain (1993). Texture Analysis. *in Handbook of Pattern Recognition and Computer Vision*. World Scientific Publishing Company.
- A. Vailaya, M. Figueiredo, A. Jain and H. J. Zhang (1999). Content-Based Hierarchical Classification of Vacation Images. *in Proceedings of IEEE Intl. Conf. on Multimedia Computing and Systems*, pages: 518-523.
- A. Vailaya, A. K. Jain and H. J. Zhang (1998). On Image Classification: City Images vs. landscapes. *Pattern Recognition* 31:1921-1936.
- V. Vapnik (1995). *The Nature of Statistical Learning Theory*. New York: Springer.
- V. Vapnik (1998). *Statistical Learning Theory*. New York: Wiley.
- N. Vasconcelos and A. Lippman (1997). Library-Based Coding: A Representation for Efficient Video Compression and Retrieval. *in Proceedings of IEEE Intl. Conf. on Data Compression*, pages: 121-130.
- N. Vasconcelos (2004). Minimum Probability of Error Image Retrieval. *IEEE Transaction on Signal Processing* 52(8):2322-2336.
- K. Wagstaff, C. Cardie, S. Rogers and S. Schroedl (2001). Constrained K-Means Clustering with Background Knowledge. *in Proceedings of Intl. Conf. on Machine Learning*, pages: 577-584.
- J. Z. Wang and J. Li (2002). Learning-Based Linguistic Indexing of Pictures with 2-D MHHMs. *in Proceedings of ACM Intl. Conf. on Multimedia*, pages: 436-445.
- X. Yang, T. S. Chua and C. H. Lee (2007). Fusion of Region and Image-Based Techniques for Automatic Image Annotation. *in Proceedings of ACM Intl. Conf. on Multimedia Modeling, Lecture Notes in Computer Science (LNCS) 4351*, pages: 247-258.
- R. Yan, A. Hauptmann and R. Jin (2003). Multimedia Search with Pseudo-Relevance Feedback. *in Proceedings of Intl. Conf. on Image and Video Retrieval, Lecture Notes in Computer Science (LNCS) 2728*, pages: 238-247.

- R. Yan and A. Hauptmann (2004). A Discriminative Learning Framework with Pair-wise Constraints for Video Object Classification. *in* Proceedings of IEEE Conf. on Computer Vision and Pattern Recognition, vol. 2, pages: 284-291.
- C. X. Zhai and J. Lafferty (2001). A Study of Smoothing Methods for Language Models Applied to Ad Hoc Information Retrieval. *in* Proceedings of ACM Conference on Research and Development in Information Retrieval, pages: 334-342.
- C. Zhang and T. Chen (2002). An Active Learning Framework for Content-Based Information Retrieval. *IEEE Transactions on Multimedia* 4:260-268.
- J. Zhang, Z. Ghahramani and Y. Yang (2004). A Probabilistic Model for Online Document Clustering with Application to Novelty Detection. *in* Proceedings of Neural Information Processing Systems, pages: 1617-1624.

AUTHOR BIOGRAPHY



RUI SHI is a Ph.D. candidate in the Department of Computer Science, School of Computing, National University of Singapore. His research interests include applying statistical models and novel image processing/computer vision techniques to tackle the problems related to pattern recognition, multimedia processing, semantic analysis of image/video contents, content-based image/video retrieval, and the applications on information retrieval and web search.

EDUCATION BACKGROUND

- Jul. 2001 – Present **Ph.D. Candidate of Computer Science**
National University of Singapore (NUS), School of Computing (SOC)
- Sep. 1998 – Jul. 2001 **M.Sc., Computer Engineering**
Institute of Computing Technology (ICT), Chinese Academy of Sciences (CAS), Beijing, China
- Sep. 1993 – Jul. 1997 **B.Sc. with Honors, Dept. of Computer Science and Engineering**
Harbin Institute of Technology (HIT), Harbin, Heilongjiang Province, China

PUBLICATIONS

1. Rui Shi, Chin-Hui Lee and Tat-Seng Chua. Enhancing Image Annotations by Integrating Concept Ontology and Text-based Bayesian Learning Model. In *the Proceedings of ACM Multimedia (ACM MM 2007)*. Pages: 341-344. September 2007, Augsburg, Germany.

-
2. Rui Shi, Tat-Seng Chua, Chin-Hui Lee and Sheng Gao. Bayesian Learning of Hierarchical Multinomial Mixture Models of Concepts for Automatic Image Annotation. In *the Proceedings of Conference on Image and video Retrieval (CIVR 2006)*, LNCS 4071. Pages: 102 – 112. Arizona, United States, 2006.
 3. Rui Shi, Wan-Jun Jin and Tat-Seng Chua. A Novel Approach to Auto Image Annotation Based on Pair-Wise Constrained Clustering and Semi-Naïve Bayesian Model. In *the Proceedings of Multimedia Modeling (MMM 2005)*. Pages: 322-327. Melbourne, Australia, 2005.
 4. Hua-Min Feng, Rui Shi and Tat-Seng Chua. A Bootstrapping Framework for Annotating and Retrieving WWW Images. In *the Proceedings of ACM Multimedia (ACM MM 2004)*. Pages: 960-967. New York, United States, 2004.
 5. Wan-Jun Jin, Rui Shi and Tat-Seng Chua. A Semi-Naïve Bayesian Method Incorporating Clustering with Pair-Wise Constraints for Auto Image Annotation. In *the Proceedings of ACM Multimedia (ACM MM 2004)*. Pages: 336-339, New York, United States, 2004.
 6. Tat-Seng Chua, Shi-Yong Neo, Ke-Ya Li, Gang Wang, Rui Shi, Ming Zhao and Hua-Xin Xu. TRECVID 2004 Search and Feature Extraction Task by NUS PRIS. *Technical Report for TRECVID'04, NIST*. Gaithersburg, Maryland, USA, 2004.
 7. Rui Shi, Hua-Min Feng, Tat-Seng Chua and Chin-Hui Lee. An Adaptive Image Content Representation and Segmentation Approach to Automatic Image Annotation. In *the Proceedings of International Conference on Image and video Retrieval (CIVR 2004)*, LNCS 3115. Pages: 545-554, Dublin, Ireland, 2004.