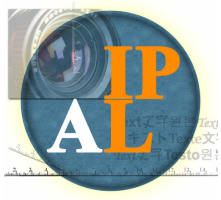
UNIVERSITE PIERRE ET MARIE CURIE
MASTER IAD

Master Thesis

Automatic Image Annotation:
Towards a Fusion of Region-based and Saliency-based Models

Trong-Tôn PHAM

Prepared at the Laboratory of
Image Perception, Annotation and Language



Under the guidance of

Dr. Joo Hwee LIM — IPAL Lab - Singapore, Institute for Infocomm Research(I^2R)
Dr. Nicolas Eric MAILLOT
Prof. Isabelle BLOCH — Département TSI, Telecom Paris (ENST)

September 2006

Acknowledgements

I would like to thank my supervisors at IPAL lab, Dr. Nicolas Maillot and Dr. Lim Joo Hwee, for their energy, patience and for the helpful discussions during the course of this work.

Special thanks to my co-supervisor in France, Prof. Isabelle Bloch ($ENST$), for her constructive comments and valuable suggestions which allowed me complete my Master study.

Furthermore, I would like to thank Jean-Pierre Chevallet, Caroline Lacoste and other staffs at Institute for Infocomm Research (I^2R) for their help and support during my internship at IPAL Lab, Singapore.

Also, I would like to thank the other students of the IPAL lab - Vlad Valea, Clement Fleury and Rabih Kassab - for their technical help and all the interesting discussions during the implementation of my project.

Last but not least, I would like to express my gratitude to my family and my friends for their love and their encouragement through this whole process.

Singapore, September 1^{st} 2006.

Abstract

This thesis addresses the problems of *automatic image annotation* (AIA) for the purpose of image indexing & retrieval in an Annotation Based Image Retrieval (ABIR) system. Specifically, we study different models of image representation in the AIA area. Up to our knowledge, nobody has tried to combine the following approaches for image representation: region-based approach and saliency-based approach. We think this combination will give a model which captures at the same time the global information and the details of objects. The proposed approach is composed of three main stages.

In the first part, the image processing stage consists of building an image representation and extracting the visual features from the image entities. Image presentation is driven by the segmentation process and the keypoint detection algorithm. Each region or keypoint is associated with a set of low-level features (color, histogram, texture, spatial location and invariant local features).

The second stage consists of leaning the relationship between semantics and visual features in image. Machine learning algorithm has been used to cluster the visual features into visterms which is an intermediate representation between high-level semantics and low-level features. The learning phase results in a co-occurrence matrix of words and visterms. The fusion of different models is expressed by the fusion of its corresponding word-by-term matrix. The high dimension of the resulting matrix makes the matching process more expensive. This can be reduced by using dimensionality reduction methods (i.e. Latent Semantic Analysis).

The last stage consists of the automatic annotation propagation scheme for a new image. The latter will be quantized in term of the visterm frequency. The propagated words list is then ranked based on the cosine similarity between visterms extracted in the image and visterms associated with words.

Experiments are conducted on Corel image datasets containing 5000 images and show good annotation performance and demonstrate the improvement of the fusion models compared to the Translation model.

keywords: Automatic Image Annotation, Machine Learning, Image Processing.

Résumé (in French)

Ce travail se place dans le context du problème de l'annotation d'images et plus généralement dans celui de l'indexation et recherche d'images au niveau sémantique. Differents modèles de représentation de l'images ont été utilisés dans le domaine de l'annotation d'images (par exemple modèle à base de régions, modèle à base de graphes et modèles à base de points de saillence). Cependant, la combinaison de deux modèles basés sur des régions et des points de saillence n'a pas été étudiée par des travaux précédent.

L'objectif de ce rapport est de comparer la signification de chaque modèle et de mesurer l'amélioration apportée par la fusion du modèle à base de régions et du modèle à base de points de sailence, face à l'utilisation seule d'une représentation. Nous tachêrons de démontrer l'éfficacité du modèle de fusion directe et du modèle de fusion par la sémantique latente pour la modélisation des informations textuelles et visuelles contenues dans des images. A notre connaissance, cette étude est la première à s'intéresser au rôle de la répresentation d'images à base de région et à base de points de saillence.

Les expérimentations conduites sur la base de Corel contenant 5000 d'images montrent la bonne performance des modèles construits. En plus, nous comparons les résulats obtenus par les modèles de fusion avec d'autres approaches, le modèle de co-occurrence et le modèle de traduction.

Contents

Acknowledgements **i**

Abstract **iii**

1 Introduction **1**
1.1 Motivation 1
1.2 Problem Statement 3
1.3 Organization 4

2 State of the Art on Automatic Image Annotation **7**
2.1 Definitions 7
2.2 Co-occurrence Model 8
2.3 Translation Model 10
2.4 Hierarchical Classification Model 10
2.5 2D Hidden Markov Model 11
2.6 Relevance Model 11
2.7 Conclusion 12

3 Framework Overview **13**
3.1 Proposed Approach 13
3.2 Image Processing 15
3.2.1 Region Segmentation 15
3.2.2 Grid Partition 16
3.2.3 Saliency Point Detection 17
3.3 Feature Extraction 17
3.3.1 Color Histogram 17
3.3.2 Texture 18
3.3.3 Local Feature 19
3.4 Semantic Learning 21
3.5 Annotation Scheme 22

4 Learning Image Semantics for Automatic Image Annotation **23**
4.1 Training 24
4.1.1 Unsupervised Learning with k-Means Clustering . . . 24

4.1.2 Image Vector Quantization 26
4.1.3 Probability Estimation of Words and Visterms 27
4.2 Fusion Model 27
4.2.1 Multi-modal Fusion 27
4.2.2 Dimension Reduction using LSA 28
4.3 Annotation 29
4.3.1 Image Vector Quantization 29
4.3.2 Projection into Latent Semantic Space 29
4.3.3 Measuring Similarity 29

5 Experimental Results **31**
5.1 Image Dataset 32
5.2 Evaluation Method 32
5.2.1 Normalised Score 32
5.2.2 Precision/Recall of Word 33
5.3 Results 33
5.3.1 Experimental Protocol 33
5.3.2 Annotation Results 34
5.3.3 Illustrative Examples 38
5.4 Discussion 39

6 Conclusion **43**
6.1 Contributions 43
6.2 Future works 44

A Latent Semantic Analysis **45**
A.1 Mathematical preliminaries 45
A.2 Latent Semantic Indexing 47
A.3 Indexing using LSI 47
A.4 Retrieval using LSI 49

Bibliography **51**

List of Figures

2.1 Example of images annotation from Corel database 7
2.2 Example of visterm extracted from the Corel image database. 8
2.3 General architecture of an AIA system 9

3.1 Principal components in our AIA system. 14
3.2 Image segmentation using the Mean-shift algorithm 16
3.3 Regular grid partitioning 16
3.4 Construction of the scale space pyramid. 20
3.5 Pipeline for learning semantic of images 22
3.6 Pipeline for annotation scheme 22

4.1 Example of visual vocabulary created from region clusters. . . 25
4.2 Image content is described by a set of visterms. 26
4.3 Co-occurence matrix of images and visterms 26

5.1 Example of images annotated with "*sun*" from the Corel dataset. 32
5.2 Comparison of the results measuring with normalised score . 35
5.3 Retrieval performance of words for each model. 37
5.4 Example of annotation results with the region-based approach. 39
5.5 Example annotation results with saliency-based approach. . . 40
5.6 The effect of choosing different number of clusters 41

A.1 Singular value decomposition 49

List of Tables

3.1 Summary of visual features used in our implementation. . . . 21

5.1 Description of studied models 31
5.2 Performance evaluation with mean precision/recall of words . 35
5.3 Comparing results of different models 38
5.4 Example annotation results with four models. 40

Chapter 1

Introduction

1.1 Motivation

Automatic image annotation (AIA) has been studied extensively for a several years. As defined by wikipedia: *"Automatic image annotation is the process by which a computer system automatically assigns metadata in the form of text description or keywords to a digital image. This application of computer vision techniques is used in image retrieval systems to organize and locate images of interest from a database. This method can be regarded as a type of multi-class image classification with a very large number of classes - as large as the vocabulary size."*. Therefore, AIA can be considered as a multi-class object recognition problem which is a extremely challenging task and still remains an open problem in computer vision.

IPAL[1] lab is a joint research laboratory between CNRS of French and I^2R of Singapore. Its goals is to promote the collaboration of French Singaporean researchers on the field of Information Technology. The main research theme of the laboratory is about multimedia (i.e. text and image) information retrieval. My project is focused on the image annotation task to help the image indexing and retrieval task. The work of the team is applied to indexing and retrieval of general images and medical images. IPAL has obtained good results in the ImageCLEF[2] competition in both medical and photographic retrieval tasks.

Why we need an automatic image annotation system?

Nowadays, the number of digital images is growing rapidly because of the increasing number of digital cameras. According to www.bizreport.com, 48 million digital cameras were sold in 2003 and this number has been increasing in the recent years. As a result, there is a huge amount of digital pictures

[1] Image Perception, Access and Language (http://ipal.imag.fr/index.htm) was officially created in March 2000

[2] http://www.clef-campaign.org/

generated each year. Thus there is a need for an efficient image management system that is able to support the end-user for fast searching, browsing by topic (e.g. Google picasa [3]) or tagging images ([flikr][4]). Content-based Image Retrieval (CBIR) has been studied for several years but the accuracy of current CBIR systems is still not sufficient for real-world applications.

Searching images by content only is an extremely hard and very challenging task for researchers in the CBIR field. A text retrieval system often helps finding rapidly related documents from a vast amount of documents containing keywords. For example, Google image search[5] offers the possibility to search for images using surrounding text and file name. Basically, Google image search is still a text retrieval system and ignores the content of the image. That is why sometimes the search performed does not lead to satisfactory results. Researcher are looking for another way to search for images. One possible approach is to derive a textual description from the image and use then use text retrieval for searching. Another idea is to fuse two modalities (i.e. visual features and text) when indexing images. This is a very promising research direction in multimedia retrieval. Image retrieval based on text is sometimes called Annotation Based Image Retrieval(ABIR) [Inoue, 2004].

However, ABIR systems have also some draw-backs. Arguing on the draw-backs of ABIR systems, researcher working in CBIR pointed out two limitations. First, ABIR requires manual image annotation that is time consuming and costly. Second, human annotation is subjective and sometimes it is difficult to describe image contents by words. For the first question, the AIA system would be the response. For the second, it is a general question and a unsolved problem for computer vision (conf. section 1.2 for more details). Imagine how could we teach a machine to understand completely the spirit of Mona Lisa's picture drawn by Leonard da Vinci? Even if we put real humans in front of this picture, it is likely that two persons will not have the same idea about the picture. In [Inoue, 2004], the author gives more interesting discussions about the need of ABIR systems.

"A picture says more than a thousand words". As this old saying, images are a important part of our daily communication. On the other hand, we could say that *"A word worths more than a thousand images"*. In fact, text contains a rich information that can be used to describe completely one image. For instance, to describe the meaning of the word "apple", one can show many pictures (e.g. 1,660,000 results returned by Google image) with different appearances such as color, size, shape, or even an iPod from the Apple company. It shows that annotations and image content are complementary for reflecting the visual content of images, disambiguating the

[3] http://picasa.google.com/
[4] http://www.flickr.com/
[5] http://images.google.com/

context in which image is involved in and for enabling the end-user with querying by abstract notions represented by keywords.

In conclusion, text is an important additional modality to understand image content. AIA systems can help image retrieval engine without requiring the manual annotation task, and in the meantime improve the performance of the retrievial task. Automatic annotation is also a difficulty task.

Why automatic image annotation is so difficult?

As mentioned earlier, automatic image annotation given a large image dataset is a challenging task. AIA is on the frontier of different fields: image analysis, machine learning, media understanding and information retrieval. Typically, image analysis is based on feature vectors and the training of annotation words is based on machine learning techniques. After the learning process, automatic annotation of new images is possible. To extract the semantics from data, general object recognition and scene understanding is required. This is an extremely hard task. As far as we known, the accuracy of the state of the art face detection systems is 85% and face is one of the easiest object to detect. Furthermore, AIA systems have to detect a least a few hundred objects at the same time from a large image database.

Another difficult known problem in the image retrieval area is the *semantic gap*. This problem is described in section 1.2.

Can we do more?

The next desirable goal in AIA research is to have a machine to tell story (i.e text description). Suppose that the the machine has been given a set of related images associated with text description. Using machine learning and other contextual information (e.g. time, location, personal information), the idea is to generate a short story concerning the image content. This is one example from many that we can imagine about the evolution of an AIA system.

1.2 Problem Statement

AIA works with both visual information and textual information. Our goal is to compare different image representations and to combine two popular approaches: region segmentation and saliency-based models.

In fact, one of the main challenges in this area is to bridge the *semantic gap* between high level semantic concepts and low level features [Lim and Jin, 2005, Smeulders et al., 2000]. Low-level features can be easily extracted from images but they are not completely descriptive for image content. High-level semantic information is meaningful and effective for image retrieval.

The semantic gap is due to at least two main problems. One problem is

how to extract the semantic regions from image data. Current object recognition techniques do not cover completely this problem. This is the *semantic extraction problem*. The other problem is the complexity , ambiguity and subjectivity in user interpretation, called *semantic interpretation problem*.

In working with the AIA literature, we recognize here some questions that an AIA systems have to deal with.

- Which image representation is appropriate to describe image semantics? The object in images are usually ill-posed, occluded and cluttered and appear in poor lighting, focus and exposure. Robust object segmentation for such noisy images is still a unsolved problem.
- Which image features can be extracted to characterize the visual semantics? This question lays on the image processing issues where each feature is represented by a numerical feature vector.
- How to correlate the visual semantics with associated text? It requires a method for modeling multi-modal datasets. This question can be posed as a machine learning problem where the joint distribution of semantic regions and words is exploited.

For the first question, many approaches can be inherited from computer vision: region-based approach, rectangular grid sampling, graph-based method or saliency-based approach. The region-based approach captures the global information of object while saliency-based extracts details of object. Up to our knowledge, nobody has tried to combine these two different approaches for image representation: region-based approach and saliency-based approach. We think this combined use will give a model with some promising properties which has at the same time the global information and the details of objects.

To address the issue of high-level semantic representation, different models such as Co-occurrence Models [Mori et al., 1999], Machine Translation Model [Duygulu et al., 2002] and Relevance Model [Jeon et al., 2003] have been proposed for learning the relationship between images and keywords.

As our focus is to compare the benefit of different approaches in image representation, we have chosen to use the co-occurrence model for its simplicity and efficiency.

1.3 Organization

This thesis is organized as follows:

Chapter 2 presents briefly the state of the art on current models proposed for automatic image annotation.

In **Chapter 3**, we give an overview on the principal components of an image annotation system. This chapter also presents a review on the underlying image processing techniques used in annotation systems.

1.3 Organization

Chapter 4 details techniques used to learn the semantics from images for annotating purposes. The important problem which will be discussed in the first section is about a semantic learning algorithm based on k-Means clustering and the construction of a word/visterm co-occurrence matrix. The second section presents our method for fusing different image representations. The last section presents the annotation scheme which consists of propagating the most probable words to a new image.

Chapter 5 presents our experimental results on the Corel image dataset. We present here a method to evaluate the annotation performance as well as some illustrative examples generated by our annotation system.

Finally, we conclude and propose further research in **Chapter 6**.

Chapter 2

State of the Art on Automatic Image Annotation

AIA has been an intensely studied topic for several years. In this chapter, we present some major contributions of researchers in the AIA field. A number of machine learning approaches have been explored for the this problem.

2.1 Definitions

We give here some brief explanations of some key words used in papers.

Annotation is a set of keywords (or text) associated with an image (fig. 2.1). The process of assigning automatically some text description to a unknown image (or partly unknown) is called *automatic image annotation* (AIA) or *image annotation propagation*.

Concept contains a group of keywords that have a related meaning and describe different aspects of a concept. For example, the concept *Paris/France* can be described as "Eiffeil, European, historical building, beach, lanscape, water".

Visual Concept is a set of images (sub-images) that represent the visual content of a concept. For instance, to "visualize" the concept "building", one can give sample images of different buildings, blocks, apartments taken

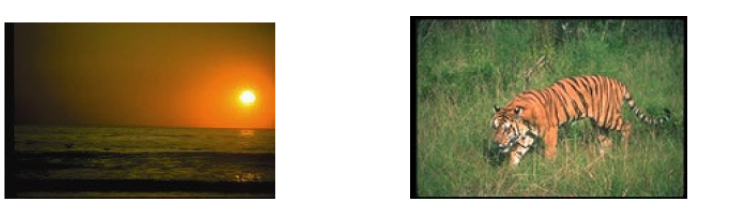


Figure 2.1: Example of images annotation from Corel database

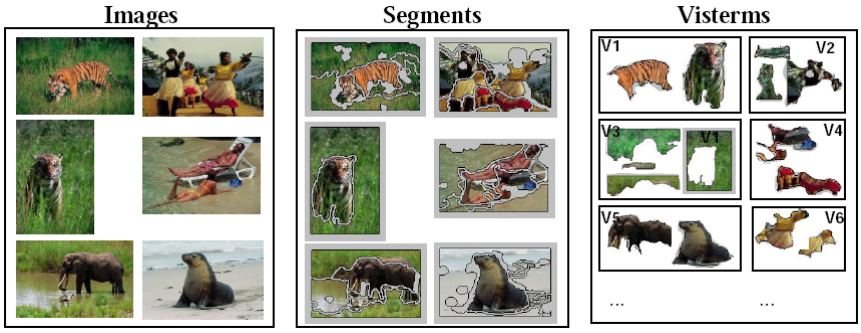


Figure 2.2: Example of visterm extracted from the Corel image database.

in different context. Visual concept are also defined as intermediate-level features between low-level feature and high-level semantics. The latter was proposed in order to solve the *semantic gap* problem (see section 1.2).

Keypoints refer to a set of local features detected in images such Harris corner points. Numerical feature vectors can be extracted from these keypoints (SIFT). These feature vectors are well adapted for characterizing small details and have some invariant properties to image transformations.

Visterms obtained by segmenting images into regions using segmentation algorithms (Normalized-cuts algorithm, Mean-shift algorithm). Segments are then clustered in different clusters. Each cluster corresponds to a visterm (figure 2.2). The same can be applied to saliency key-points detected in images.

Blob as used in [Barnard et al., 2002] has the same meaning as visterm.

Image vocabulary is a set of visterms or blobs.

2.2 Co-occurrence Model

Among the first people dealing the automatic image annotation problem, Mori et al. [Mori et al., 1999] have proposed the co-occurrence method to capture the relationship between images and words. In their co-occurrence model, the authors adopt two processes to construct the learning system. First, they use the grid-based segmentation algorithm to uniformly divide each image into sub-images. Second, the voting probability of each word for a set of segmented images is estimated by the process of a vector quantization of the feature vector of sub-images.

This approach enables the AIA system to transfer a set of words assigned to the whole image to each image part. Another original aspect of this method is the probability of each word for a set of image parts is estimated by using a vector quantization of the feature vectors of the sub-images. However as pointed out in [Feng et al., 2004], the system tends to annotate unknown images with high frequency words and require large number of training sam-

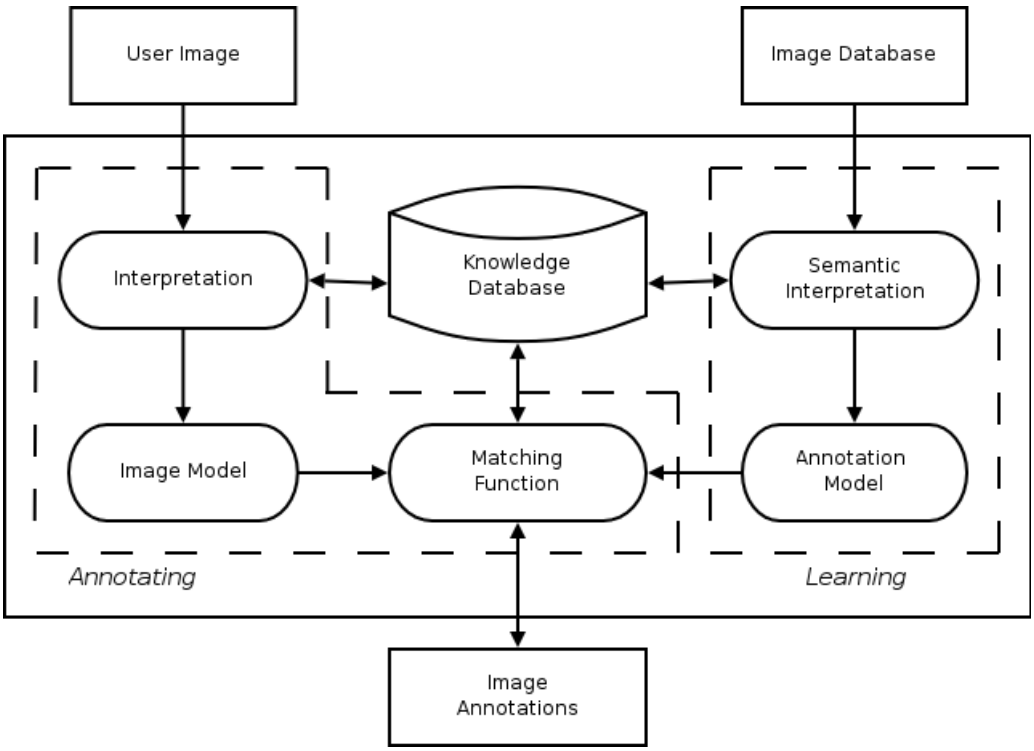


Figure 2.3: General architecture of an AIA system

ples to estimate the correct probabilities. This method has been inherited by many other current approaches [Duygulu et al., 2002, Carson et al., 1999] in capturing the relationships between words and images.

2.3 Translation Model

In [Duygulu et al., 2002], Duygulu et al. propose a model of object recognition as machine translation. The authors model image annotation as the translation of the visual representation of an image to a textual representation. The visual vocabulary is generated by clustering the image regions segmented using N-cut segmentation algorithm. A clustered region correponds which to a visual term, is called *blob*[1]. Then a mapping between blobs and keywords associated with the images is learned using a method based on the EM[2] algorithm. This process is analogous with learning a lexicon from an aligned vocabulary.

The translation model is a substantial improvement from the co-occurrence model. The classical IBM machine translation models was used to translate vocabulary of blobs in to vocabulary of words. Duygulu et al. model can achieve a better accuracy than the co-occurence models. However, the translation model can capture the correlation between blobs and words but the dependencies between them are not captured. Moreover, the training stage with the EM algorithm requires a high computational effort and is very time consuming when dealing with large datasets.

2.4 Hierarchical Classification Model

Motivated by the statistical clustering models proposed by Hoffmann and Puzicha [Hofmann and Puzicha, 1998] for co-occurrence data, Barnard and Forsyth, in [Barnard and Forsyth, 2000], adapt the hierarchical aspect clustering model for image annotation. The hierarchy of models for generating word and image segments is derived from clustered images in the training set. The clusters capture contextual similarities while the nodes capture generality of concepts. Words and blobs are then represented as a distribution over the nodes of the hierarchy. Therefore, words and blobs with similar distribution can be considered correlated. The hierarchies induced by image clusters provide some semantic interpretation for the models.

The proposed hierarchical classification model for image annotation uses an hierarchy derived from a knowledge source that is based on concept organization in natural language. Inspired from the idea of hierarchical classification of visual concepts, the authors of [Srikanth et al., 2005] use WordNet[3]

[1] Blobworld: http://elib.cs.berkeley.edu/blobworld/
[2] Expectation-Maximization
[3] http://wordnet.princeton.edu/

to explore ontologies of annotation words. The effect of using the hierarchy in generating is demonstrated by improvements of the annotation system.

2.5 2D Hidden Markov Model

Li and Wang [Li and Wang, 2003] introduced a statistical modeling approach to the problem of *automatic linguistic indexing*[4] of pictures. A two-dimensional multiresolution hidden Markov models (2D MHMMs) is used to model the stochastic process of associating an image with the textual description of a concept. First of all, each image is summarized by a collection of features vector extracted and spatially arranged on a pyramid grid. The 2D MHMM fitted to each image category plays a role of extracting representive information about the category: clusters of feature vectors at multiresolution and the spatial relation between the clusters.

For a test image, feature vectors are extracted from the pyramid grid. The likelihood of the feature vectors being generated by each profiling 2D MHMM is computed. To annotate the image, words are selected from those in the text description of the categories yielding highest likelihoods. This method have shown good accuracy and has a high potential for image annotation.

2.6 Relevance Model

Recently, relevance models proposed by Jeon et al. [Jeon et al., 2003] for image annotation have shown significant performance inprovements. Relevance Models originally introduced for text retrieval and cross-lingual retrieval [Lavrenko and Croft, 2001] are used. By computing the joint probability of images and words, the latter approach can introduce the context in image without having to do so explicity. For example, in the image context, "tigers" are more often associated with "grass, water, tree or sky" and less often with man-made objects such as "car" or "computer" and the relevance model will take advantage of this context.

Unlike the co-occurrence models [Mori et al., 1999] predict the probability P(w|I) of a word w given an image I, relevance model is interested for generating a set of annotation words w.

$$P(w|I) \approx P(w|v_1...v_m) = \frac{P(w, v_1...v_m)}{\sum_w P(w, v_1...v_m)}$$

Then, the joint distribution computed as an expectation over image training set J

$$P(w|v_1...v_m) = \sum_J P(w, v_1...v_m|J)$$

[4] ALIP system: http://wang.ist.psu.edu/IMAGE/alip.html

And since the events are independent, the joint distribution can be expressed as:

$$P(w|v_1...v_m) = \sum_J P(J)P(w|J)\prod_{i=1}^{m} P(v_i|J))$$

The smoothed maximum likelihood is used to estimate the distribution of words w and visterms v over the training set. The Continous Relevance Model (CRM) is unsuitable when annotation length varies widely beacause of the estimation of multinomial distributions of words. An extended version of this method uses the multiple Bernoulli distribution for modeling image annotation [Feng et al., 2004]. The results show a slight improvement in performance of the annotation task. Compared to the Translation Model, the Relevance Model performs much better on the same dataset (with the same training and test images and the same features).

2.7 Conclusion

We have presented some state of the art models in the domain of AIA. For the purpose of understanding the problem of image annotation and demonstrating the improvement in combining different approaches, we do not compare directly our results with this *state of the art* models. We will show that our method outperforms approaches such as the Co-occurence Model and the Translation Model. The annotated words will be assigned to the entire images and not to specific blobs as in [Duygulu et al., 2002].

Chapter 3

Framework Overview

3.1 Proposed Approach

We present in this section an overview of our proposed approach for the AIA problem.Figure 3.1 illustrates three main stages of an AIA system.

1. **Image processing** consists of extracting image data (i.e. region segmentation, saliency point detection) from the image. It also consists of computing the numerical feature vector (e.g color histogram, texture and geometric information) associated with each region or point.

2. **Semantic learning** consists of two process. First, similar extracted image data will be grouped into clusters using the unsupervised clustering algorithm (e.g. k-Means clustering). The second process consists of modeling the correlation between the visterm (clustered data) and the high-level textual information.

3. **Annotation scheme** for new image consists of an image processing module and an image vector quantization module. The propagated word list will be obtained by ranked $k - best$ similarity words.

These three phrases are clearly distinct from each other. They can be associated with the three layers of a classical paradigm in machine vision of Marr[Marr, 1983]: the image processing layer (1), the mapping layer(2), the high-level interpretation layer(3). Our contributions are mainly related to the semantic learning problem and to the annotation scheme. In the learning phrase, we compare different existing approaches of image presentation such as the region-based approach and the saliency-based approach. After that, we propose a method for fusing these approaches based on direct matrix fusion and the LSA[1] method that was initially proposed in the information retrieval community.

[1]Latent Semantic Analysis

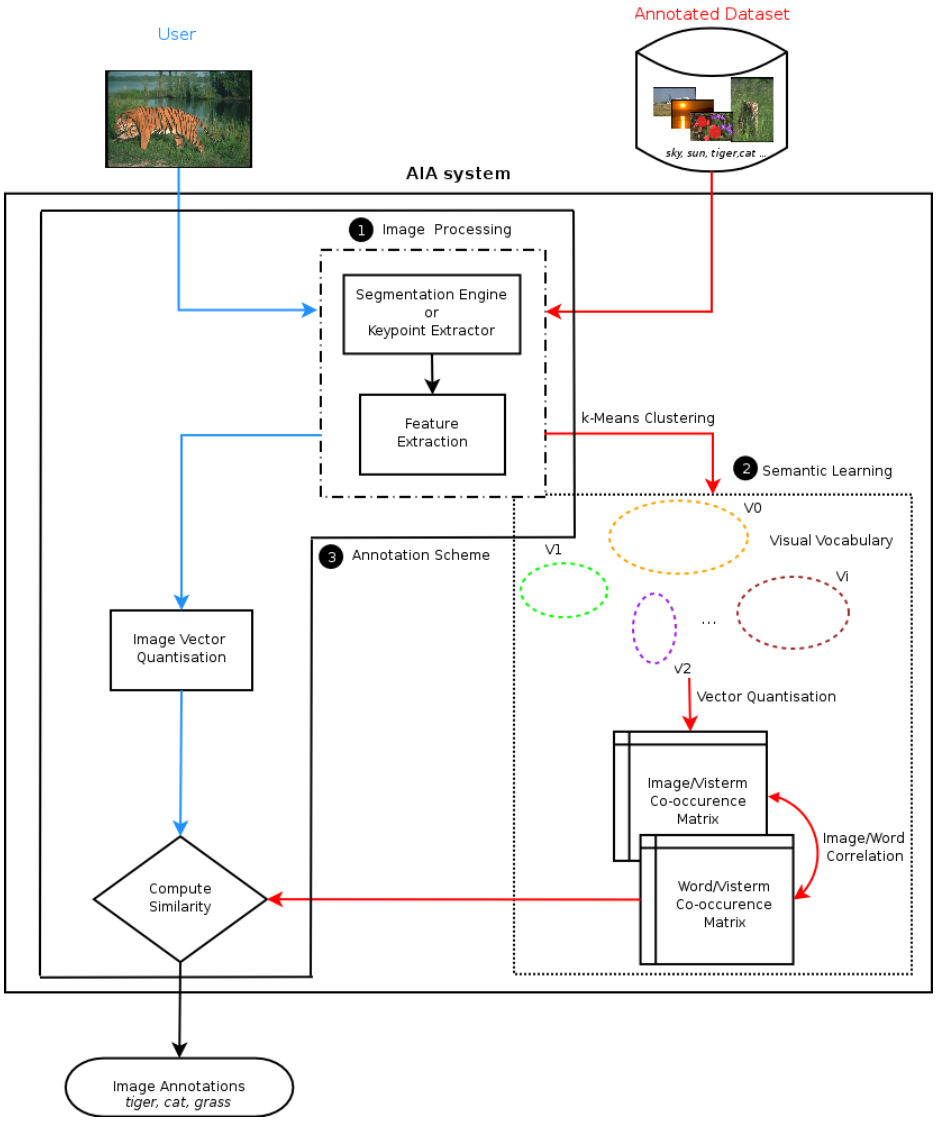


Figure 3.1: Principal components in our AIA system.

3.2 Image Processing

Finding a good image representation for automatic image annotation is difficult. We propose to combine a region-based approach with a saliency-based approach.

3.2.1 Region Segmentation

Image segmentation into regions may help to find out the semantic relation between words and objects contained in image. Image segmentation avoids considering every pixels of the image but rather some groups of pixels that contain more information. As defined in [Smeulders et al., 2000], there are two types of image segmentation:

- **Strong segmentation** is a division of the image data into regions in such a way that region T contains the pixels of the object O.

- **Weak segmentation** is a grouping of the image data in conspicuous regions T internally homogeneous according to some criterion, hopefully with T a subset of O.

These algorithms were based on some homogeneity criterion in each region such as color and texture. It is also difficult to obtain a strong segmentation so that each region contains an object. The weak segmentation helps to eliminate this problem and sometimes helps to identify better an object in image.

Many algorithms have been proposed for region segmentation. A graph-cut algorithm has been used to find minimum normalized-Cut (N-cut) in a graph pixel of image [Shi et al., 1998]. Normalized-cut algorithms gives bad results with cluttered background as they use only color as homogeneous criterion. The computational time of normalized-Cut algorithm is also high. The Blobworld system [Carson et al., 1999] has used this famous algorithm for building image tokens.

Mean-shift segmentation algorithm [Comaniciu and Meer, 2002a] search for a higher density of data distribution in images. The Mean-shift segmentation algorithm was recognized as a very flexible algorithm (user can choose different parameters: window size, filter kernel, region threshold, etc...) and perhaps the best segmenting technique to date. For this reason, we use this algorithm for segmenting images in our framework. Even if few people have selected this algorithm for their annotation systems, we think that it tends to segment objects. This is an interesting property for building our model.

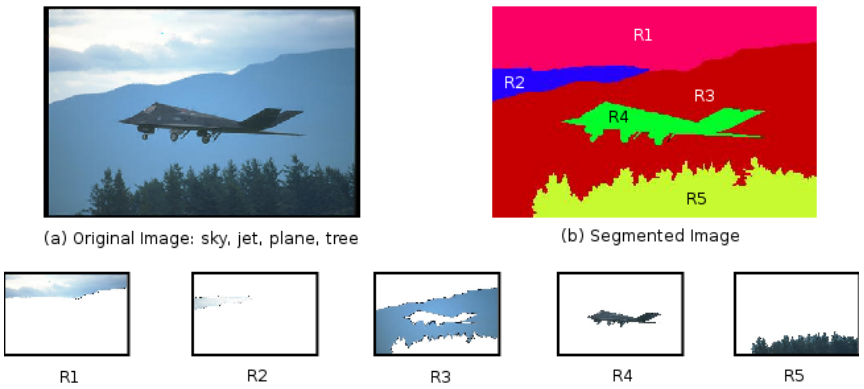


Figure 3.2: Example of image segmentation using the Mean-shift algorithm.

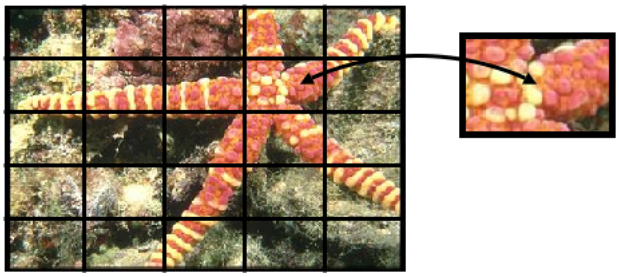


Figure 3.3: An image decomposed into 5x5 sub-images using regular grid partitioning

3.2.2 Grid Partition

This is the simple method for segmenting an image. A rectangular grid with fixed-size [Feng et al., 2004] slides over (can be overlap) the image. For each rectangular grid, a feature vector is extracted. The rectangular size can be variable to make a multi-scale version [Lim and Jin, 2005] of grid partitioning. Combining overlapping and multi-scale partitioning enables to cope with changes in object positions and image scale changes.

Using grid provides a number of advantages. The performance of rectangular grid as pointed out in [Feng et al., 2004] is better than the method based on region segmentation in annotation tasks. In addition, there is a significant reduction in the computational time required for segmenting the image. Using grid partitioning (with more regions than produced by the segmentation algorithm) allows the model to learn how to associate words with images using a much larger set of training samples.

In our implementation, we apply a regular grid partitioning resulting in 5x5 sub-windows. This value yields 25 rectangular grids for each image which is apparently a good trade off between computational requirements and feature vectors significance.

3.2.3 Saliency Point Detection

Saliency-based models have been experimented firstly in the object recognition field [Lowe, 1999] then has been studied extensively in the CBIR [2] [Schmid and Mohr, 1997] and AIA [Hare and Lewis, 2005] fields for several years. Image content is extracted using local descriptors such as SIFT[3] detector [Lowe, 1999], Harris corner detector [Harris and Stephens, 1988] or affine invariant point detector [Mikolajczyk and Schmid, 2002]. These points are localized in the zones of the image which contain rich information and some invariant properties to image transformations (e.g. affine, scale, rotation)

The saliency-based model has shown good performance in object recognition problems with very high accuracy on some limited object databases and with certain kind of objects (building, car, bicycle ...). However, dealing with more general object and with a large dataset, the performance of saliency-based decreases surprisingly.

Our goal is to compare two most important current approaches in the AIA field : region-based and saliency-based approaches. For that purpose we have chosen to implement the saliency-based model based on SIFT descriptors developed by D. G. Lowe in University of British Columbia.

3.3 Feature Extraction

3.3.1 Color Histogram

Color is an important feature for object recognition and for matching image. We present here some popular color models used in the AIA area.

- **RGB**[4] is the fundamental representation of color in computer. RGB uses an additive model in which red, green and blue are combined in various ways to reproduce other colors. This color model is simple. It is also sensitive to illumination changes.

  Nevertheless, this color model in widely used in object recognition [Duffy and Crowley, 2000] and in image annotation systems (Blobworld [Carson et al., 1999]).

- **HSV**[5] Artists sometimes prefer to use the HSV color model over alternative models such as RGB or CMYK, because of its similarities to the way humans tend to perceive color. HSV encapsulates information about a color in terms that are more familiar to humans.

[2] Content-based Image Retrieval
[3] Scale-invariant Feature Transform
[4] Red Green Blue
[5] Hue Saturation Value

Using this this color model in object representation has shown its efficiently and it independence to illumination changes.

- **L*a*b** The CIE 1976 L*a*b color model, defined by the International Commission on Illumination (Commission Internationale d'Eclairage, hence its CIE initialism), is the most complete color model used conventionally to describe all the colors visible to the human eye. The three parameters in the model represent the lightness of the color (L), its position between magenta and green (a*) and its position between yellow and blue (b*).

Color Histogram Considering a three-dimensional color space (x, y, z), quantized on each component to a finite set of colors which correspond to the number of bins N_x, N_y, N_z, the color of the image I is the joint probability of the intensities of the three color channels. Let $i \in [1, N_x]$, $j \in [1, N_y]$ and $k \in [1, N_z]$. Then, $h(i, j, k) = Card\{p \in I \mid color(p) = (i, j, k)\}$. The color histogram H of image I is then defined as the vector $H(I) = (..., h(i, j, k), ...)$.

In our system, we use the RGBL color model, a combination of RGB and L*a*b color representation, to give more robustness to the illumination changes in images. This simple feature consists on four independently calculated histograms, one for each color component R, G and B and the luminance L, defined as $L = (min(R, G, B) + max(R, G, B))/2$. Each histogram contains the number of cells given in the parameters, which means that the resulting feature vector has a dimensionality of 4*cells.

Although, only using color information is not enough to discriminate two different class. For instance, two images may have the same dominating color (e.g. blue) but may present different objects (e.g. sky, water, beach). We need another feature to construct the visual features that are more discriminant and robust to image transformations.

3.3.2 Texture

We use the Gabor Wavelet Transforms as suggested in [Maillot, 2005]. In addition to good performances in texture discrimination and segmentation, the justification for Gabor filters is also supported through psychophysical experiments. Texture analyzers implemented using 2-D Gabor functions produce a strong correlation with actual human segmentation [Reed and Wechsler, 1990]. Gabor functions are Gaussian modulated by complex sinusoids. In two dimensions they take the form:

$$g(x, y) = \frac{1}{2\pi\sigma_x\sigma_y} \exp\left(-\frac{1}{2}\left(\frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2}\right) + 2\pi jWx\right)$$

A dictionary of filters can be obtained by appropriate dilatations and rotations of $g(x, y)$ through the generating function:

$$g_{mn}(x,y) = a^{-m} g(x',y'),\ m = 0,1,...,S-1$$
$$x' = a^{-m}(xcos\ \theta + ysin\ \theta),\ y' = (-xsin\ \theta + ycos\ \theta)$$

where $\theta = n\pi/K$, K the number of orientations, S the number scales in the multiresolution, and $a = (U_h/U_l)^{-1/S-1}$ with U_l and U_h the lower and upper center frequencies of interest. A compact representation needs to be derived for learning and classification purposes. Given an image $I(x,y)$, its Gabor wavelet transform is then defined as:

$$W_{mn}(x,y) = \int I(x,y) g_{mn} * (x - x_1, y - y_1)\, dx_1 dy_1$$

where $*$ represents the complex conjugate. The mean μ_{mn} and the standard deviation σ_{mn} of the magnitude of the transform coefficients are used to represent the image.

$$\mu_{mn} = \int\int |W_{mn}(x,y)|\, dx\, dy$$
$$and\ \sigma_{mn} = \sqrt{\int\int (|W_{mn}(x,y)| - \mu_{mn})^2\, dx\, dy}$$

A feature vector is then constructed using μ_{mn} and σ_{mn} as feature components:

$$\mathbf{f} = [\mu_{00}\ \sigma_{00}\ \mu_{01}\ \sigma_{01}\ ...\ \mu_{mn}\ \sigma_{mn}]$$

As result, we obtain a numerical vector of 30 dimensions for 6 orientations and 5 scales changes. Also note that, the texture feature is computed only for rectangular grid as it is difficult to compute the texture vector for one arbitrary region.

3.3.3 Local Feature

Scale-Invariant Feature Transform (SIFT) features have been introduced in [Lowe, 1999]. These features belong to the class of local image features. The are well adapted for characterizing small details. They are invariant to image *scaling*, image *translation*, and partially invariant to *illumination changes* and *affine* for 3D projection. First, features are detected through a staged filtering approach that identifies stable points in scale space. The result of this detection, is a set of key local regions. Then, given a stable location, scale, and orientation for each key point, it is possible to describe the local image regions in a manner invariant to these transformations.

Key locations are selected at maxima and minima of a difference of Gaussians applied in scale space. The input image I is first convolved with the

Gaussians function to give an image A. This is then repeated a second time with a further incremental smoothing to give a new image B. The difference of Gaussians function is obtained by subtracting image B from A. This difference of Gaussians is formally expressed as:

$$D(x, y, \sigma) = (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y)$$

with k corresponding to the strength of smoothing and

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} \exp -(x^2 + y^2)/2\sigma^2$$

This differentiation process is repeated with different values of k.

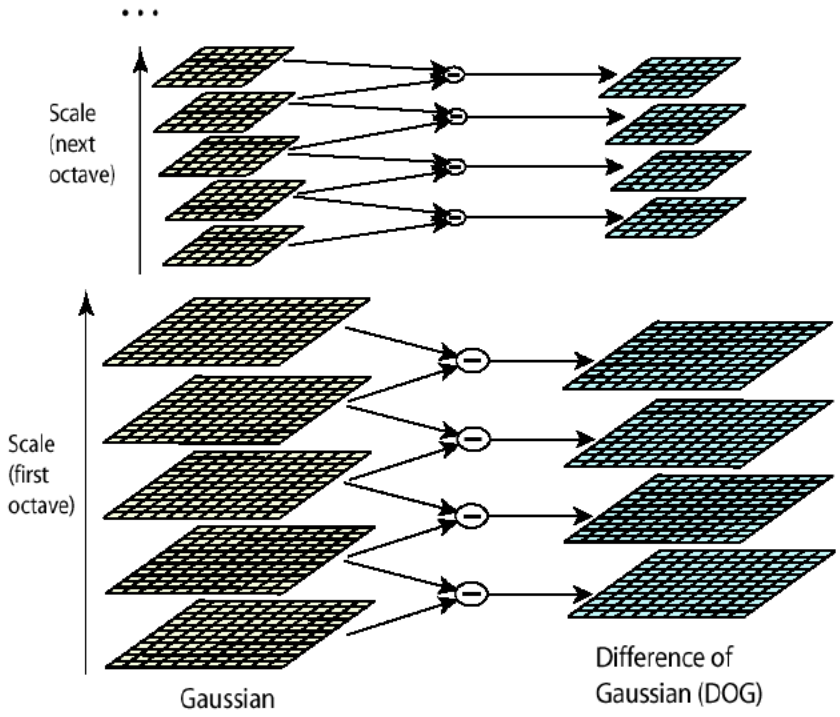


Figure 3.4: Construction of the scale space pyramid.

A change of scale consists of sampling the smoothed images by using a bilinear interpolation. The combination of scaling and smoothing produces a scale space pyramid. An overview of the scale/space construction is shown in fig. 3.4.

Minima and extrema detection of $D(x, y, \sigma)$ uses this scale space pyramid and is achieved by comparing each sample point to its neighbors in the current image and 9 neighbors in the scale above and below. It is selected only if it is larger than all its neighbors or smaller than all its neighbors.

The result of this selection is a set of key-points which are assigned a location, a scale and an orientation (i.e. obtained by gradient orientation computation).

The last step consists of assigning a numerical vector to each keypoint. The 16×16 neighborhood around the key location is divided into 16 sub-regions. Each sub-region is used to compute an orientation histogram. Each bin of a given histogram corresponds to the sum of the gradient magnitude of the pixels in the sub-region. The final numerical vector f associated with a keypoint is of dimension 128.

An important information to represent the image structure is the location of regions or keypoints extracted from images. Thus for each feature vector, we append two coordinates x_c and y_c of the location in the end. Table 3.1 summarizes the features used in our implementation and its dimensional vector. For the fusion model,the final feature vector is the combination of different feature types and give a quite big vector. For this reason, we need a method to reduce the dimension of this vector and also to reduce the computational. Latent Semantic Analysis technique has been used to analyse the pricipal factor of data thus to reduce the dimentsion of high dimension datasets.

FEATURE	Vector Quantization	Dimensions
RGBL histogram	16 cells $\times$ 4 channels	64
Texture	6 orientations $\times$ 5 scales	30
SIFT descriptor	$4 \times 4 \times 8$ orientations	128
Location	(x_c, y_c)	2

Table 3.1: Summary of visual features used in our implementation.

3.4 Semantic Learning

The role of image semantic learning is to fill the semantic gap between high-level text description and low-level image features. This stage consists of finding a model to represent the joint distribution of visterms and words. This means that each segmented region (or keypoint) is associated with one or several keywords. This process can be done by manual assigning or using machine learning to train the system. In our approach we use the unsupervised learning algorithm k-Means. As shown in figure 3.5 the learning stage is composed of three main tasks:

1. **Unsupervised learning** with k-Means clustering group the similar feature vector from extracted data (e.g. region or saliency point) to

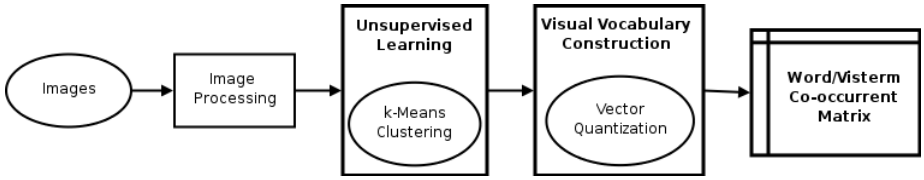


Figure 3.5: Pipeline for learning semantic of images

the same cluster. These clusters are called the *visterms*. Clustering transforms the continuous feature space into a discrete number of clusters.

2. **Visual vocabulary construction** consists of quantizating each image by a numerical vector.

3. Construction of **word/visterm co-occurrent matrix** takes into account the textual information and the visual features.

We introduce more details of this stage in chapter 4.

3.5 Annotation Scheme

Given a new image, the role of the annotation task is to predict the most probable words to be associated with the new image. First, regions and saliency points are extracted from the image and the associated feature vectors are computed. Then, the annotation scheme consists of two main tasks (figure 3.6):

1. **The image vector quantization** process assigns each image region (or saliency point) to a nearest cluster. The visterms frequencies are then computed. Finally, the resulting vector is normalized with the total number of visterms containing in the image.

2. **Measuring similarity** consists of computing the distance of a new image to the quantized words. Different metric distances can be employed to measure the similarity.

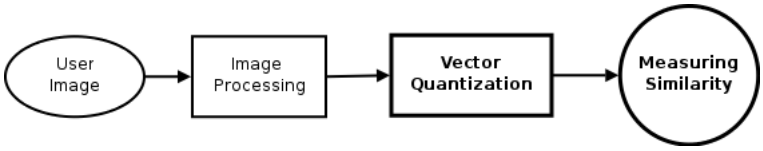


Figure 3.6: Pipeline for annotation scheme

Section 4.3 gives more detail about this stage.

Chapter 4

Learning Image Semantics for Automatic Image Annotation

In the previous chapter, we have presented the image processing stage that leads to a set of feature vectors (e.g. RGBL color, texture and SIFT descriptors) describing the visual contents of images. The goal of this chapter is to show how the resulting feature vectors are involded in a learning process which aims at modeling the distribution of visterms and words. The resulting model can then be used to propagate the appropriate annotations to a new image.

Machine learning techniques have been used to cluster regions and keypoints into clusters, that we call visterms. As a result, the joint distribution of words and visterms can be estimated. To fuse different models, two methods are proposed. First, the direct fusion of the two matrices obtained from two different models (i.e. regions and keypoints). The second approach uses Latent Semantic Indexing (LSA) to reduce the dimension of the fused matrix.

Below is the notations used in this chapter:

$$\begin{cases} \mathcal{I} : \text{set of image} \\ \mathcal{W} : \text{set of keywords associated with images} \\ \mathcal{V} : \text{set of visterms (visual vocabulary)} \\ \mathcal{R} : \text{set of image regions} \\ \mathcal{L} : \text{set of keypoints} \\ \mathcal{F} : \text{set of features associated with regions or keypoints} \mathcal{V} \\ |.| : \text{cardinality of set or vector} \end{cases}$$

This chapter is structured in three main sections. Section 4.1 presents how a set of training images are used to train an annotation model. Section

4.2 presents a method for fusing different approaches. Section 4.3 shows how a new image is automatically annotated with the trained model.

4.1 Training

Given a set of features $\mathcal{F}$ extracted from regions $\mathcal{R}$ or keypoints $\mathcal{L}$, the goal of the training stage is to classify these feature vectors into the homogenous groups that can be represented by a visual concept. For this purpose, we apply the k-mean clustering algorithm to the pool of feature set and clustered them with the provided number of cluster. The next section will introduce the principle of this algorithm.

4.1.1 Unsupervised Learning with k-Means Clustering

k-Means Clustering

The k-means algorithm is a very popular technique to partition data in machine learning. The goals is to find k mean vectors $\mu_1, ..., \mu_k$, one for each cluster. The basic idea of this interactive algorithm is to assign the feature vector to the cluster such that the sum of squared error E is minimum

$$E = \sum_{i=1}^{k} \sum_{j=1}^{N_j} ||x_{ij} - \mu_i||^2$$

where x_{ij} is the jth point in the ith cluster, μ_i is the mean vector of ith cluster and N_j is the number of pattern in the ith cluster.

In general, the k-means clustering algorithm works as follows:

1. Select an initial mean vector for each of k clusters.
2. Partition data into k-clusters by assigning each pattern x_n to its closest cluster centroid μ_i.
3. Compute new mean clusters $\mu_1, ..., \mu_k$ as the centroids of k clusters.
4. Repeat step 2 and 3 until the cluster criterion is reached.

In step 1, the initial mean vectors can be chosen randomly from k seed points in the data. The partitioning is then performed from these initial points. In the second step, to measure the distance between two patterns, different metric distances (e.g Hamming distance, Euclidean distance, Mahalanobis distance, etc...) can be applied. Usually, the Euclidean distance is good enough to measure the distance between two vectors in the feature space. In step 3 the centroid μ_i for each cluster is re-estimated by computing the mean of cluster members. The number of iterations can be used in the last step as a convergence criterion.

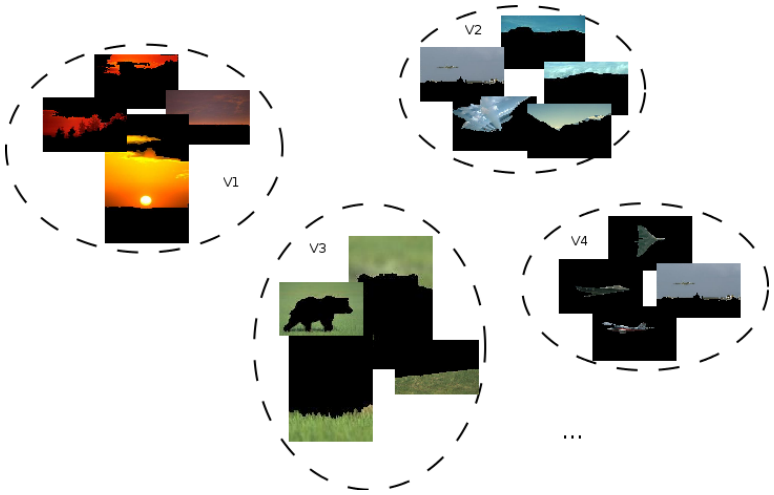


Figure 4.1: Example of visual vocabulary created from region clusters. Each cluster corresponds to a visterm in the visual vocabulary. The number of clusters is also the size of vocabulary.

The k-means algorithm is simple and easy to implement. It has a time complexity of $\mathcal{O}(nk)$ for each iteration. This algorithm is provided in the LTI-lib computer vision library [1] and the only parameter which needs to be fixed is the number of clusters.

Visual vocabulary construction

The clustering algorithm is applied on set of feature vectors x_j. The result is a set of numerical label v_i associated with each region R_j or keypoint L_j (figure 4.1). The number of clusters is the number of visterms contained in the visual vocabulary.

In fact, the choice of the number of clusters is still an issue and affects the quality of image annotation. In [Duygulu et al., 2002], the number of blob-tokens generated is set at 500 on the Corel dataset. Therefore, we chose the same number of cluster in order to compare directly our results with this approach. Moreover, we also provide experimental results with different size for the visual vocabulary in section 5.4 to confirm our choice.

Hence, for each image I_n, we obtain a set of couples (R_j, v_i) for regions or a set of couples (L_j, v_i) for keypoints. Figure 4.2 shows an image of a plane which has been segmented into 5 regions. Each region is associated with a label which correspond to a visterm v_i taken from the visual vocabulary $\mathcal{V}$. In this case, the set of visterms $\{v_2, v_6, v_2, v_4, v_9\}$ characterize the visual description of the image content.

[1] http://ltilib.sourceforge.net/

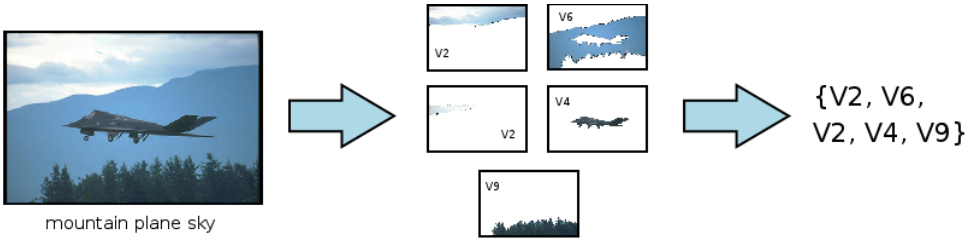


Figure 4.2: Image content is described by a set of visterms.

4.1.2 Image Vector Quantization

Inspired by the idea of information retrieval area where documents are weighted by the term frequency contained in each document. Given a set of image $I = \{I_1, I_2, ..., I_n\}$, each image is weighted by the visterm frequency appearing in the image.

We note :

$\forall k \in \mathcal{V}, I_n \in \mathcal{I} : a_k^{I_n}$ is the number of occurence of the visterm v_k in image I_n

For example, image I2 in figure 4.3 contains three visterms v2, v3 and v500 and the number of each visterm is respectively 1, 11 and 2. So the corresponding vector for image I2 is : $a_{500}^{I2} = \{0, 1, 11, ..., 2\}$

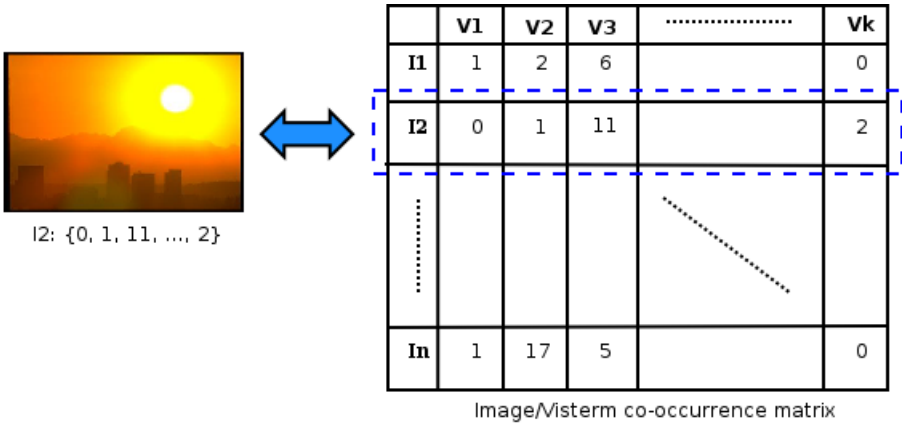


Figure 4.3: Co-occurence matrix of images and visterms. Each line of matrix represents the term frequency for each image in the training set. A column corresponds visterms which belong to the visual vocabulary.

Finally, the entire training images can be represented by a visterm-by-image matrix M^I of the size $|\mathcal{V}| \times |\mathcal{I}|$

$$M_k^n = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1k} \\ a_{21} & a_{22} & \dots & a_{2k} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nk} \end{pmatrix}$$

4.1.3 Probability Estimation of Words and Visterms

Once the visterm-by-image is constructed, we need to estimate the distribution of visterm and word from annotated image to represent the relation between textual information and visual information. Using the same way to compute the number of occurence of visterms in an image, the number of occurence of visterm (v_k) for the word (w_i) is defined as the number of visterm v_k appearing in any image I which is annotated by word w_i. In our case, the number of words is 374 taken from 4500 annotated images. This relation is captured by a word-by-visterm matrix M^W of the size $|\mathcal{V}| \times |\mathcal{W}|$.

$$M_k^m = \begin{pmatrix} b_{11} & b_{12} & \dots & b_{1k} \\ b_{21} & b_{22} & \dots & b_{2k} \\ \dots & \dots & \dots & \dots \\ b_{m1} & b_{m2} & \dots & b_{mk} \end{pmatrix}$$

Matrix Normalisation

The word-by-visterm matrix need to be normalized. **Zero-mean normalization** has been used to normalize each line of matrix M^W. There are m lines in the matrix and the dimension of the vector is k. In order to perform the above operation we have to compute the mean μ_i and standard deviation σ_i for each vector:

$$\mu_i = \frac{1}{k}\sum_{i=1}^{k} b_{mi}, \quad \sigma_i = \sqrt{\sum_{i=1}^{k}(b_{mi} - \mu_i)^2}$$

and then normalize it using the following equation:

$$b_{mi} = \frac{b_{mi} - \mu_i}{\sigma_i}$$

4.2 Fusion Model

4.2.1 Multi-modal Fusion

From now two matrices for the region-based approach (noted by $M_{\mathcal{R}}$) and for the saliency-based approach (noted by $M_{\mathcal{L}}$) have been computed. Our goal to fuse these two approaches to take advantages of the different approaches.

To do that, the two matrices are concatenated together to form a bigger matrix $M_{\mathcal{RL}}$ of size $|W| \times (|V^{\mathcal{R}}| + |V^{\mathcal{L}}|)$.

$$M^m_{k+l} = \begin{pmatrix} b_{11} & b_{12} & \dots & b_{1k} & b_{1(k+1)} & b_{1(k+2)} & \dots & b_{1(k+l)} \\ b_{21} & b_{22} & \dots & b_{2k} & b_{2(k+1)} & b_{2(k+2)} & \dots & b_{2(k+l)} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ b_{m1} & b_{m2} & \dots & b_{mk} & b_{m(k+1)} & b_{m(k+2)} & \dots & b_{m(k+l)} \end{pmatrix}$$

In our case, we choose the equal number of visterm (i.e. 500 visterms) for region-based approach and saliency-based approach. It means that $|V_{\mathcal{R}}| = |V_{\mathcal{L}}|$. Finally, the fusion matrix has a large dimensions (i.e. 4500x(500+500) = 4 500 000 elements) and occupies a lot of memory. Obviously, a common computer can not handle the big matrix and causes the crashing or memory fault when loading a big matrix. Therefore we need a technique to reduce the dimension of matrix to the lower dimension. We propose to use the LSA technique in order to solve this problem.

4.2.2 Dimension Reduction using LSA

The LSA (or LSI[2]) technique was originally using in the information retrieval field for text retrieval. This technique is similar with the popular technique called PCA[3] in data analysis. It helps to analyse the document-by-term matrix by mapping the original matrix into lower dimensional space. The other advantage of using LSA is reduce the noise in documents by keeping only the most important part from the term-by-document matrix.

Detail about the LSA technique will be given in appendix A. We only give here a brief idea on how the LSA isused in text retrieval.

Given a term-by-document matrix M rank r, M is decomposed into 3 matrices using SVD [4] as follows:

$$M = U\Sigma V^t$$

where

$$\begin{cases} U : \text{is the matrix of eigenvectors derived from } MM^t \\ V^t : \text{is the matrix of eigenvectors derived from } M^tM \\ \Sigma : \text{is an } r \times r \text{ diagonal matrix of singular values } \sigma. \\ \sigma : \text{are the positive square roots of the eigen-values of } MM^t \text{ or } M^tM \end{cases}$$

In fact, this transformation divides matrix M into two parts. One is related to the documents and the second related to the terms. By selecting

[2]Latent Semantic Indexing
[3]Principal Component Analysis
[4]Singular Value Decomposion

only k largest values from matrix Σ and keep the corresponding column in U and V, the reduced matrix M_k is given by

$$M_k = U_k \Sigma_k V_k^t$$

where $k < r$ is the dimensionality of the concept space. Indeed, the choosing of parameter k is not obvious and depends on the working database. It should be large enough to allow fitting the characteristics of the data. Otherwise, it must be small enough to filter out the non-relevant representation details. Our goal is not finding a value k optimal for our image dataset, so we fix the new dimensionality of word-by-visterm matrix at 100, which is not optimal but allow us to compare with different models.

4.3 Annotation

In this preliminary work, we only look at a very simple method of annotation propagation based on the similarity rank of matching words with images. The idea is simple. Images that have similar distributions of visterms with words should describe similar semantic content.

4.3.1 Image Vector Quantization

For the new image, we first apply the image processing step as described in section 3.2 to perform the image segmentation and keypoint extraction. Feature vectors will be computed from the corresponding regions and keypoints (section 3.3). For each region or keypoint, the corresponding visterm is computed by assigning its feature vector to the closest cluster. Then, image vector is constructed by estimating the number of occurence of each visterm and is normalised using the zero-mean normalization algorithm. We denote the image vector by V_R for region-based model and V_L for saliency-based approach.

4.3.2 Projection into Latent Semantic Space

For the fusion method, we concatenate two image vector come from two models (V_R and V_L). The vector combination can be expressed as V^I with dimension $|V| = |V^R| + |V^L|$. After that, we need to project this vector into the latent semantic space to obtain a pseudo-vector with the dimension reduced.

$$V_k = V^t U_k$$

4.3.3 Measuring Similarity

Given an image vector V and a matrix word-by-visterm M constructed previously, we want to measure the similarity of this vector with words contained in the word-by-visterm matrix.

Cosine Similarity

Most text retrieval system represents a document and query as a set of terms. These terms are represented as axes in a vector space using weighted term frequency and thus the distance to an axis corresponds to a term. Two documents, V_q and V_d can be considered to be similar if the angle between their vectors is small. The normalised scalar product is used to measure similarity:

$$cos(\theta) = \frac{V_q \bullet V_d}{|V_q||V_d|}$$

A cosine similarity of 1 implies that two documents are identical, an a similarity of 0 implies they are unrelated. We use this measure to compute the similarity between an image vector and each vector extracted from word-by-visterm matrix. The obtained results also show that the cosinus similarity works slightly better than the normal Euclidean distance. The results are then ranked using the similarity values. We choose to propagate only the k most probable words (e.g. top 5 or top 10). After that, an evaluation method will be used to measure the precision of system.

Next chapter shows how the proposed approach has been applied to an image dataset. An evaluation method and the overall results obtained are also presented.

Chapter 5

Experimental Results

This chapter presents the results from the testing on a commonly used image dataset (i.e. Corel dataset). The algorithm was implemented in C/C++ language using LTI lib [1] for the image processing operations. The system runs under Linux operating system on a Pentium4 3.0 GHz with 1GB memory.

In this preliminary experiments, our goal is to compare different approaches of current image representation using in the AIA area. Moreover, we propose a method for fusing different existing models. In the limit of time, the grid-based model will not be implemented but is mentioned as one approach of the region-based model. The details of our implemented models is given in table 5.1.

MODEL	Region	Grid	SIFT
REGION-BASED (RB)	$\surd$	$\times$	$\times$
GRID-BASED (GB)	$\times$	$\times$	$\times$
SALIENCY-BASED (SB)	$\times$	$\times$	$\surd$
DIRECT FUSION (DF)	$\surd$	$\times$	$\surd$
LSA (LSA)	$\surd$	$\times$	$\surd$

Table 5.1: Description of studied models

We also compare our results with the co-occurrence model [Mori et al., 1999] and the machine translation model from [Duygulu et al., 2002] as the baseline for comparison. Finally, we provide some discussion on the obtained results.

[1] http://ltilib.sourceforge.net/doc/homepage/index.shtml

5.1 Image Dataset

The Corel dataset[2] consists of 5000 images from 50 Corel Stock Photo CDs. Each CD includes 100 images on the same theme. The collection covers a variety of subjects ranging from urban scenes to nature scenes and from artificial objects to animals. The image layout are 384x256 pixels and 256x384 pixels. Overall there are 374 keywords in the dataset and each image is associated with 1-5 keywords.

In our experiments, we divide the image collection into 2 parts: a training set of 4500 images and a test set of 500 images (with the annotation ground-truth). This dataset is used widely in AIA area [Barnard and Forsyth, 2000, Carson et al., 1999, Feng et al., 2004] because of the professional quality of the photos shot and the simplicity and the pertinence of annotations.

Figure 5.1: Example of images annotated with "*sun*" from the Corel dataset.

5.2 Evaluation Method

5.2.1 Normalised Score

In [Barnard and Forsyth, 2000], Barnard et al. suggest the use of the normalised score measure

$$E_{NS}^{(model)} = \frac{r}{n} - \frac{w}{N-n}$$

where r is the number of correctly predicted words, n the actual number of keywords in the ground-truth, w is the number of wrongly predicted words, and N denotes the size of vocabulary (i.e. 374 words). The score gives a value of 1 if the testing image is annotated exactly, a value of 0 for predicting both everything or nothing, and -1 if the exact complement of the actual words set is predicted.

However, the use of normalised score has also some limitations. The impact of incorrectly predicted words is weaker than the impact of correctly

[2]Available at: http://www.cs.arizona.edu/people/kobus/research/data/eccv_2002

annotated words. As a consequence, one can obtain a good normalised score if their annotation algorithms generate enough words to cover the ground-truth set. For example, Monay and Gatica-Parez [Monay and Gatica-Perez, 2004] report that in their test database, the normalised score is maximised when the system return about 40 keywords per image. Thus, if the algorithm is selecting all of the correct labels (but also more incorrect ones), it can lead to a very noisy annotation. Nevertheless, we also include this measurement for comparing the performance of different models as reference values.

5.2.2 Precision/Recall of Word

As adviced in [Feng et al., 2004], the performance of an AIA system can be measured by the *precision* and *recall* of each word. This performance evaluation method is more interesting in the sense of expressing the automatic annotation task as a text retrieval process. The performance evaluation of AIA sytem is considered as the retrieval performance of an information retrieval system.

Let A be the number of images automatically annotated with a given word, B is the number of images correctly annotated with that word, C is the number of images having that word in ground-truth annotation. Then the *precision* (P) and *recall*(R) is computed as follows:

$$R = \frac{B}{C}, P = \frac{B}{A}$$

To evaluate the system performance, recall and precision values are averaged over the testing words. Unlike in classical image retrieval, the aim is to get high precision for all value of recall. In annotation, the aim is to get both high precision (high proportion of correctly annotated images to the number of relevant images) and high recall (high overall proportion of correct annotated images). Hence, we choose to use this performance measurement to compare our results with other models such as the co-occurrence model and the machine translation in the next section.

5.3 Results

5.3.1 Experimental Protocol

Training involves 4500 images from testing set using the method presented in section 4.1. For the region-based approach, images are segmented using the Mean-shift segmentation algorithm [Comaniciu and Meer, 2002b]. Each image contains about 5-10 regions. For the saliency-based approach, we detected the keypoints using SIFT detector and append to each keypoint its location. We fix the parameter of Lowe's algorithm so that 10-50 keypoints will be extracted from a image of size 384x256 pixels.

The next step consists of extracting for each region or keypoint its corresponding features as detailed in Table 3.1. Feature vectors are then clustered into 500 visterms using k-Means clustering algorithm. We emphasize that the selection of the number of cluster (or visual vocabulary size) affects the accuracy of the annotation process. We have chosen 500 visterms for the visual vocabulary as advised in [Duygulu et al., 2002] for the comparison purpose but it can be refine depend on the image database.

Once the visual vocabulary is constructed, each image is labeled with the corresponding visterms. The correlation between images and visterms is captured by the image-by-visterm matrix. From here the word-by-visterm co-occurrence matrix is computed. It has to be noted that only 371 words have been learnt from the 374 word in the vocabulary. The fusion of different approaches is expressed by the concatenation of different image-by-term matrices. The fusion matrix usually has a very high dimension (i.e. 4500×1000 matrix), thus we need a method to reduce the matrix to lower dimension. LSA technique has been employed to reduce the dimension of fusion to 100 only.

Finally, we test our algorithm using 500 images from a held-out test set to evaluate the performance. The k-nearest words (we used 10 most similar words from the propagation list) to image is propagated. We compare the automatic annotated words with the ground-truth annotations provided along with 500 test images. Considering the annotating problem as the problem of retrieving an image from the test set using keywords from vocabulary, we measure the performance of annotation using the *recall/precision* for each word and the normalised score for a whole test set.

5.3.2 Annotation Results

Given a list of images files associated with the predicted annotations, each image is predicted with the 10 best words ordered by the similarity value. We measure the performance of our annotation system with the normalised score and the retrieval performance by precision and recall of words as presented in section 5.2.

Figure 5.2 shows four curves of the normalised score values. The curve was generated by increasing the number of propagated words from 1 to 10. When the number of predicted words increases, the values of normalised score will also increase. As seen in this figure, the direct fusion method yields the best performance over other methods. This also gives us an initial point of view of the relative performance for each model. This observation will be later confirmed by the performance evaluation with precision and recall. This is discussed further below.

Table 5.2 resumes the number of predictable words and the mean per-

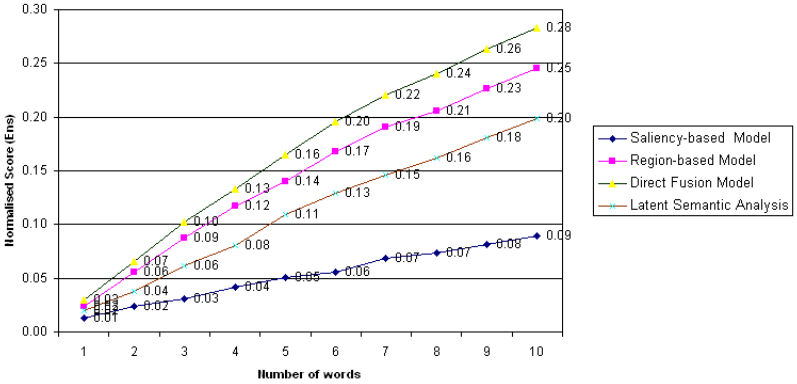


Figure 5.2: Comparison of the results measuring with normalised score (E_{NS}).

word precision/recall obtained by each model. There are 260 words occurred in the test set but not all of them can be predicted by the annotation system. Let say, one word is predictable if its average recall value is strictly greater than 0, otherwise, word is considered unpredictable. Based on this convention, there are 87 words predicted by SB model, 81 words predicted by DF model and 75 words predicted by LSA nodel. But the SB model can predict only 36 words which is surprisingly low.

Model	SB	RB	LSA	DF
Number of Predictable Words				
#words with recall > 0	36	87	75	81
Performance Evaluation				
Mean per-word Recall	0.32	0.35	0.31	0.36
Mean per-word Precision	0.21	0.23	0.24	0.26

Table 5.2: Performance evaluation with mean precision/recall of words

As reported in this table, the direct fusion model provides the best overall performance on mean precision (0.36) and mean recall (0.26). The region based model has 0.35 mean precision and 0.23 mean recall. The saliency based model acquired 0.32 mean precision and 0.21 mean recall but computed with only 36 words. In fact, the combination of two methods gives a better mean precision (+13% in comparing with the second best model) and a slightly improvement of mean recall (+3% better than the second best model).

The reason that the mean recall value is not improved so much in the fusion model may be due to the fact that the number of predicted words by saliency-based model is too low. When two models are combined together

these unpredictable words will drop in terms of the overall mean recall value in the fusion model.

In Figure 5.3, we present some best retrieved words for each model. These words are sorted in the decreasing order of the precision value. As shown in this figure, the SB model and DF model have a quite regular distribution. The value of precision and recall is similar. On the other hand, the SB model predicts annotations with a high fluctuation distribution yielding noisy annotations. For example, "smoke" and "sky" obtained a high value of precision (1.0) but the recall values are very low 0.2 and 0.01 respectively . On the contrary, "desert", "buddist", "petals" and "ocean" have a high recall but low precision. It means that, these words can be easily predict but with the very low accuracy.

We also observes that some words have both high precision and high recall in every models such as "sun", "sunset", "jet", "plane", "sky". It means that these words could be easily predicted with a high probability that it is a true annotation. This comment suggests that some keywords are more related to the visual appearance than others. For instance, the visual terms of "sun" and "sky" are more easy to recognize (i.e. color and texture) because of more homogeneity than other terms (e.g. "tree", "tiger", "flower").

Finally, we give the comparison results of our models with others approaches. We compare the annotation performance of the four models: Co-occurrence Model [Mori et al., 1999], Translation Model [Duygulu et al., 2002] and the fusion models proposed in this work (Direct Fusion and method using LSA). We report the results on two set of words as experimented in [Lavrenko et al., 2003]:

- subset of 49 best words which was used by [Duygulu et al., 2002]
- complete set of all 260 words that occur in the testing set.

Table 5.3 show the mean averaged *precision/recall* of each model on both word sets. The results clearly shows that our fusion model is better than the Translation Model. Specifically, for the subset of 49 words we obtained +32% improvement in recall and +80% improvement in precision. For the biggest dataset with a total of 260 words, the percentage of improvement are respectively +150% and +33% for recall and precision. The LSA model also provides the better results than the Translation Model but still slightly less performance than the DF model. On the overall, the DF model provides the best performance.

5.3 Results

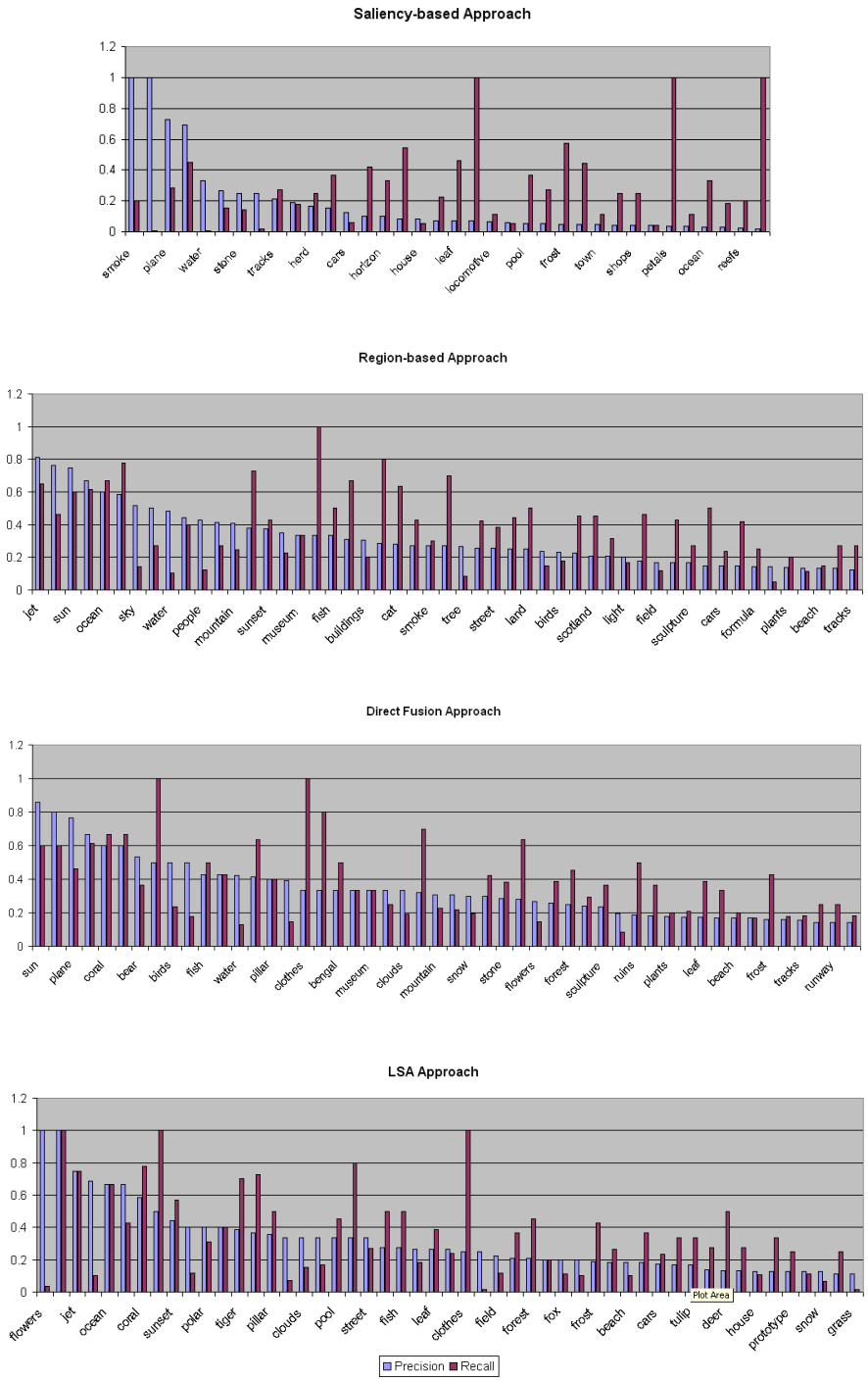


Figure 5.3: Retrieval performance of words for each model.

Models	Co-occurence	Translation	LSA	**DF**
#words with recall ≥ 0	19	49	224	**226**
Results on 49 best words, as in [Duygulu et al., 2002]				
Mean per-word Recall	-	0.34	0.43	**0.45**
Mean per-word Precision	-	0.20	0.33	**0.36**
Results on all 260 words				
Mean per-word Recall	0.02	0.04	0.09	**0.10**
Mean per-word Precision	0.03	0.06	0.07	**0.08**

Table 5.3: Comparing results of different models on the task of automatic image annotation.

5.3.3 Illustrative Examples

This section shows some illustrative examples of the annotations generated by our models.

Figure 5.4 shows that images 108007, 119064, and 34065 are successfully annotated by the region-based model. While image 161079 is wrongly predicted ("light" and "shadows" with "statue") and image 267027 assumes "cloud" and "valley" for "water". With the human judgement, these incorrect annotations are not completely wrong because its meaning is relatively close to the image content. However, as we mentioned earlier, the manual annotation task is subjective. Our goal to find a AIA system that can find a best deal between annotating "nothing" or "everything".

Figure 5.5 shows the limit of the saliency based approach on the Corel image database. Some important annotations are missed and some images are annotated with uncorrelated annotations (e.g. "nest", "herb" and "birds" for "cars", "railroad"). Even if the SIFT descriptor is very successfull for recognizing rigid objects, the performance of the saliency model on more general object classes is not satisfactory.

It confirms that, in a more general object database, the global information (e.g. region-based, grid-based) is more important than the local appearance. Because the global representation capture more important information of images (e.g. color, texture and shape) than the local features. In this case, the invariant feature is not so important as the images have a wider range of image transformations than scale changes, translation and orientation.

Table 5.4 gives a more comprehensive results of the annotation generated by the four models. We note that the direct fusion model has a better annotation in the first image and the last image. It also gives a more relevant and meaningful annotation to the image. The automatically annotated

Image	Human Annotation	Machine Predicted
108007.jpeg	bengal cat forest tiger	**bengal cat tiger forest** head
119064.jpeg	buildings skyline street	**street** gate **buildings skyline** tower
161079.jpeg	pillar sculpture statue stone	**pillar sculpture** light **stone** shadows
276027.jpeg	mountain sky snow water	**mountain** clouds **sky snow** valley
34065.jpeg	jet bear polar snow	**polar jet** face **bear snow**

Figure 5.4: Example of annotation results with the region-based approach.

words by SB model have no overlap with the annual annotation.

5.4 Discussion

We have compared a variety of methods for predicting annotations from images. The experiment shown that the fusion method for region-based model and saliency-based model offers some improvements (higher precision and better recall) from the stand-alone model. The comparison result with the state-of-the-art Translation model also demonstrated that the fusion model outperforms the base line.

This section discusses some questions raised when we implement the system.

Importance of choosing segmentation algorithm

Segmentation is a difficult problem in computer vision. Segmenting im-

Image	Human Annotation	Machine Predicted
10075.jpeg	jet plane sky smoke	**plane jet smoke** tracks prototype
143093.jpeg	desert sand sky	**desert sand** city scotland tiger
182022.jpeg	cars locomotive railroad train	**train** nest herd **locomotive** birds
276029.jpeg	mountain sky tree wate	ice **mountain** frost **water** hills

Figure 5.5: Example annotation results with saliency-based approach.

Image			
Human	beach people sunset water	coral ocean reefs	cars formula tracks wall
SB	petals swimmers leaf black pool	sphinx man girl statue woman	frost arch ice house coral
RB	**sunset** sea sunrise shadows tables	**reefs coral ocean** fan bridge	**formula** bengal log **tracks** head
LSA	light **sunset** sun reflection clouds	**reefs coral ocean** fish bridge	forest log cat tiger **tracks**
DF	**sunset** island palm sunrise **beach**	**coral ocean reefs** fish fan	**formula tracks** arch bridge **cars**

Table 5.4: Example annotation results with four models.

age into individual objects is nearly impossible for computers. The segmentation is done on a per image basis. However, if we start from a rectangular partition, the number of visterms will increase which leads to a better chance

of associating regular regions with the correct words. We believe that the using of rectangular grid will improve the performance of the current region-based system.

The effect of choosing number of clusters

The number of clusters in k-Means clustering is always an issue in machine learning. There are several methods proposed for automatic selection of this number (i.e. adaptive k-Means) but they also increase the computational complexity of the algorithm. We have tested our system with two numbers of clusters, k=374 which is exactly the number of words in the vocabulary and k=500 as implemented in [Duygulu et al., 2002]. Figure 5.6 shows the normalised score curve of the region-based model as an evidence. The results are better when we cluster with k=500.

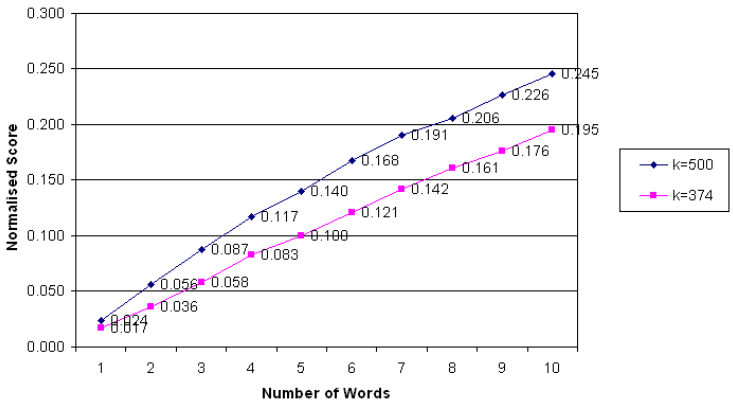


Figure 5.6: The effect of choosing different number of clusters on the region-based model measured by normalised score.

The effect of text clustering

We also did the image clustering based on the annotation contained in each image. There are some very nice clusters which have strong semantics and visual relations like (sun, sunset, sunrise, beach, sky), (skyline, jet, plane, cloud, smoke) or (coral, ocean, reefs). So one promising direction for the better quality of clustering is the combination of semantic representation with the visual representation. It might help the clustering algorithm better weighting the importance of visual part and semantic part of image. This would be considered for our future research.

Chapter 6

Conclusion

6.1 Contributions

In this thesis, we have investigated the problem of *automatic image annotation* for an image retrieval system. Our approach is based on image processing algorithms combined with machine learning techniques. The proposed approach is composed of three main phases:

- An **image processing** phase consists of segmenting images into regions and extracting keypoints located in these images. Feature vectors are then computed based on the color, texture, orientation histogram and the location of corresponding regions and keypoints.
- A **semantic learning** phase consists of constructing an intermediate level of semantics (the visual terms). The relation between words and visterms is captured by the word-by-visterm matrix. This stage helps to fill the *semantic gap* between high-level image semantic and low-level image features extracted by image processing algorithm.
- An **annotation** phase that uses the image processing stage to extract the visual features and the quantisation process to compute the occurence of visterm in the new image. The propagated words are based on the ranked similarity values of image-by-visterm vector with each vector extracted from the word-by-visterm matrix.

Two main questions that occupied our attention when dealing with the AIA problem are: (i) which image representation is more appropriate for characterizing the semantic entities in images? and (ii) can we combine different approaches to make a more reliable system?

Based on the above system sketch, we have constructed three different annotation models: Region-based Model, Saliency-based Model, and Direct Fusion Model. Furthermore, inspired by the idea of using Latent Semantic Analysis in text retrieval area to analyze the meaning of terms in documents,

we applied this method to analyze the word-by-visterms matrix and set up a fourth model: LSA Model. Two contributions have been made in this internship. First, we compared two main approaches in AIA fields (Region-based Model and Saliency-based Model) and provided a discussion on the significance of each model. Second, we have proposed a method for fusion of different models to take advantage of the strong points of each model.

The conducted experiments have shown that the fusion models obtained the better result than other stand alone methods. The Direct Fusion Model has given the best overall results during the experimental process but lack of efficiency and of memory performance. The LSA Model has also produced interesting results. Indeed, its major advantages is to reduce the computational complexity of the annotation process. In addition, the experimental results have shown that our fusion approach outperformed the base-line Translation model.

6.2 Future works

Even if we have used a simple method for modelling multimodal information (i.e. visual information and textual information), the fusion model is very promising. The combination of different approaches could bring a better performance to an AIA system. However, some further extensions of current system could be considered to improve the annotation performance.

In the short-term vision, our goal is to extend the current framework on other kind of image representations, for example, regular grid partioning and graph-based model. Future work will also involve an automatic image features selection process for each category of image.

As discussed above, the k-Means clustering lacks of efficiency and it is difficult to choose the number of clusters. One promising solution is to use the one-class Support Vector Machine (SVM) to estimate the parameter for each word.A probabilistic model for multimodal modelling is also interesting. The current state-of-the-art annotation systems [Feng et al., 2004] use statistical generative models and estimate the joint probability of image/text using a Bernoulli distribution. We hope that an intensive study of the correlation of word and visual term could lead to a model which is better for representing of this relation.

For a long-term perspective, automatic video annotation is an interesting issue that currently requires for a lot more research efforts. The annual TRECVID conference draws many attentions of research community in content-based retrieval of digital videos. One of the major objectives of this forum is to benchmark state-of-the-art algorithms for multimodal analysis of digital media targeting semantic retrieval. We think that our framework could be extended and tested to evaluate its performance and its robustness on this type of dataset.

Appendix A

Latent Semantic Analysis

This chapter will first present the mathematical preliminaries needed to understand how LSA works. Then we will have an overview on LSA in its original form as employed on text retrieval.

A.1 Mathematical preliminaries

Definition 3.1.1 1 *The singular value decomposition(SVD) of any matrix* $A^{m\times n}$ *of rank* $r \leqslant q = min(m, n)$*, denoted by SVD(A) is defined as:*

$$A = U\Sigma V^* \tag{A.1}$$

where $U^{m\times q}$ *and* $V^{q\times n}$ *are unitary matrices*$(U^*U = V^*V = I_q)$*-orthogonal, if* A *is real-,* $\Sigma = diag(\sigma_1, \sigma_2, \sigma_q)$ *and* $\sigma_i > 0$ *for* $i \leqslant r$ *and* $\sigma_i = 0$ *for* $i > r$*. The frst* r *columns of* U *and* V *are called the left, respectively right, singular vectors and are the eigenvectors of* AA^* *and* A^*A *respectively. The elements* σ_i *are the nonnegative square roots of the* n *eigenvalues of of* AA^* *or* A^*A

From now on, we consider only real matrices, so the adjoint of a matrix is equal to its transpose and unitary means orthogonal. In the remainder of this section we will present the most interesting properties of SVD from the point of view of usage in information retrieval.

Theorem 3.1.1 1 *The SVD(A) as defined by Def. 3.1.1 is unique except for:*

1. *Exchanging column* i *with column* j *in both* U *and* V *and exchanging* σ_i *with* σ_j*.*
2. *Replacing column* i *in both* U *and* V *with a unitary linear combination of columns* $l_1, l_2, \ldots l_h$ *if and only if*

$$\sigma_{l_1} = \ldots = \sigma_{l_h} = \sigma_i.$$

Theorem 3.1.2 1 *Let the SVD of A be given by Def 3.1.1 with*

$$\sigma_1 > \sigma_2 > \ldots = \sigma_{r+1} = \ldots = \sigma_q = 0$$

which is possible according to Thm. 3.1.1 and let, without loss of generality $m \geq n$. *Let R(A) and N(A) denote the range respectively the null-space of A. Then*

1. *rank property:* $rank(A) = r$, $N(A) \equiv span\{v_{r+1}, \ldots, v_n\}$ *and* $R(A) \equiv span\{u_1, \ldots, u_r\}$, *where* $U = [u_1 \ldots u_m]$ *and* $V = [v_1 \ldots v_n]$.

2. *dyadic decomposition:* $A = \sum_{i=1}^{r} u_i \cdot \sigma_i \cdot v_i^T$.

3. *norms:* $\|A\|_F^2 = \sum_{i=1}^{r} \sigma_i^2$ *and* $\|A\|_2^2 = \sigma_1^2$.

Theorem 3.1.2 presents several properties of the singular value decomposition of a matrix. The last two are especially interesting because they allow us to construct an approximation of A and estimate its quality. This is expressed by the following theorem:

Theorem 3.1.3 1 *(Eckart and Young)*
Let the SVD of A be given by Def 3.1.1 with

$$\sigma_1 > \sigma_2 > \ldots = \sigma_{r+1} = \ldots = \sigma_q = 0$$

and define the truncated SVD approximation A_k *of A as*

$$A_k = \sum_{i=1}^{k} u_i \cdot \sigma_i \cdot v_i^T = U_k \sigma_k V_k^T \tag{A.2}$$

then

$$min_{rank(B)=k} \|A - B\|_F^2 = \|A - A_k\|_F^2 = \sum_{i=k+1}^{q} \sigma_i^2$$

Theorem 3.1.3 states that the best rank k approximation of A with respect to the Frobenius norm is A_k as defined by equation A.2. We can even say that A_k is the best rank k approximation of A with respect to any unitarily invariant norm and hence:

$$min_{rank(B)=k} \|A - B\|^2 = \|A - A_k\|^2 = \sigma_{k+1}$$

A.2 Latent Semantic Indexing

Latent semantic indexing (LSI) was first introduced in information retrieval as a text retrieval technique. This section will offer an overview on how LSI works in that area.

LSI is a high-dimensional linear associative model that embodies no human knowledge beyond its general learning mechanism, to analyze a large corpus of natural text and generate a representation that captures the similarity of words and text passages. It uses statistical techniques for the purpose of infering and extracting relations of contextual usage of words in in passages of discourse. No use of humanly constructed dictionaries, knowledge bases, semantic networks, grammars, syntactic parsers, or morphologies is made. As a input it takes only raw text wich containes words separated into passages wich have a semantic.

The birth of LSI was motivated by a a set problems that existed in textual retrieval techiniques. The fundamental problem was that users wanted to retrieve documents on the basis of their conceptual meaning, and individual terms provide little reliability about the conceptual meaning of a document. This issue has two faces: synonymy and polysemy.

Synonymy is used to describe the fact that there are many ways to refer to the same object. Users in different contexts, or with different needs, knowledge, or linguistic habits will describe the same concept using different terms.

By *polysemy* we refer to the fact that most words have more than one distinct meaning. In different contexts or when used by different people, the same term takes on a varying referential significance. Thus, the use of a term in a query may not necessarily mean that a document containing the same term is relevant at all.

LSI is said to overcome this deficiencies because of the very way it associates meaning to words and groups of words according to mutual constrains set by the context in wich they appear. So for synonymy, it is likely that words wich have a similar meaning from a human point of view will appear in similar contexts and thus will be given a similar representation by LSI. The non-beneficial efect of polysemy is reduced by LSI because the meaning of a word can be conditioned not only by other words in the document but by other appropriate words in the query not used by the author of a particular relevant document.

A.3 Indexing using LSI

The first step of LSI is to compute the term-by-document matrix. In this matrix each row stands for a term and each column for a document. So if the

matrix is $A = [a_{ij}]$ then the element a_{ij} represents number of occurrences of word i in document j. This matrix is usually very sparse because not every word appears in a certain document.

Most of the times weight is added to the entries of the matrix A to give an expression of the word's importance in a certain document and in the whole domain in general. Mathematicaly this can be expressed as

$$a_{ij} = L(i,j) \times G(i) \tag{A.3}$$

where $L(i,j)$ is the local weight of term i in document j and $G(i)$ is the global weight for term i.

Next the term-by-document matrix is decomposed through SVD see equation A.1. SVD is the general method for linear decomposition of a matrix into independent principal components . It finds a representation of all the intercorrelations between a set of variables in terms of a new set of abstract variables, each of which is unrelated to any other but which can be combined to regenerate the original data. In other words, SVD is a form of eigenvalue-eigenvector analysis or principal components decomposition and, in a more general sense, of two-way, two-mode multidimensional scaling. The output of the SVD is a product of three matrices in wich the term-by-document matrix has been decomposed. One matrix describes the original row entities as vectors of derived orthogonal factor values, another describes the original column entities in the same way, and the third is a diagonal matrix containing scaling values such that when the three components are multiplied, the original matrix is reconstructed.

An aproximation of the initial matrix is constructed A_k see equation A.2 taking into account only the k largest singular values. This representation has major advantages over the original term-by-document matrix. In the reconstructed matrix a word's vector is computed by SVD as a linear combination of data from every cell in the matrix and not only the information about the word's own occurrences across documents, as represented in its vector in the original matrix. Rather, SVD uses everything it can to induce word vectors that best predict all and only those text samples in which the word occurs. This expresses a belief that a representation that captures much of how words are used in natural context captures much of what we mean by meaning. The derived matrix captures most of the important underlying structure in the association of terms and documents, yet at the same time removes the noise or variability in word usage wich means that minor differences in terminology will be ignored.

A graphical representation of the SVD and the way the derived A_k is constructed is presented in figure A.1, page 49.

The number of dimensions retained in LSA is an empirical issue. Because the underlying principle is that the original data should not be perfectly regenerated but, rather, an optimal dimensionality should be found that will cause correct induction of underlying relations, the factor-analytic approach

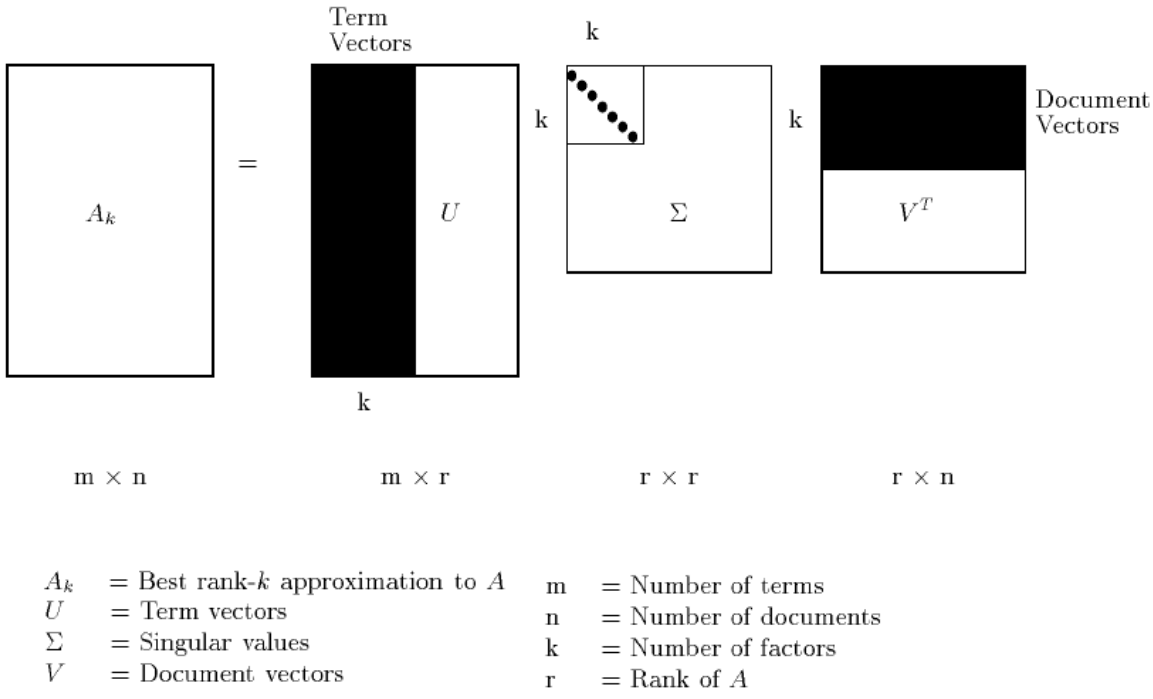


Figure A.1: Singular value decomposition

of choosing a dimensionality that most represent the true variance of the original data is not appropriate. Instead some external criterion of validity is sought, such as the performance on a synonym test or prediction of the missing words in passages if some portion are deleted in forming the initial matrix.

A.4 Retrieval using LSI

For the retrieval part of this technique the essential step is to represent the user's query in the reduced dimensionality latent space. To do this the raw query wich is a pseudo-document itself must be projected in the k-dimensional space using the factors of the SVD decomposition. This is done through the following transformation

$$q_p = q^T U_k \Sigma_k^{-1} \tag{A.4}$$

where q is the vector representation of the users query that is a sequence of words, properly weighted using equation A.3, U_k is the truncated term vectors matrix, and Σ_k is a the diagonal matrix containing the k-largest singular values. Once projected into the latent semantic space the query q_p can be compared to the indexed documents and all the documents can be ranked according to the similarity with the query. As a measure for similarity the cosine distance is usually used(see equation 4.3.3). This measure of similarity is used here because cosines can be interpreted as representing the direction or quality of a meaning rather than its magnitude. For a text segment, that is roughly like what its topic is rather than the quantity of information it contains.

Bibliography

[Barnard et al., 2002] Barnard, K., Duygulu, P., Forsyth, D., de Freitas, N., Blei, D., and Jordan, M. (2002). Matching words and pictures. *JMLR, 2002*.

[Barnard and Forsyth, 2000] Barnard, K. and Forsyth, D. (2000). Learning the semantics of words and pictures.

[Carson et al., 1999] Carson, C., Thomas, M., Belongie, S., Hellerstein, J. M., and Malik, J. (1999). Blobworld: A system for region-based image indexing and retrieval. In *Third International Conference on Visual Information Systems*. Springer.

[Comaniciu and Meer, 2002a] Comaniciu, D. and Meer, P. (2002a). Mean shift: A robust approach toward feature space analysis. *IEEE Trans. Pattern Anal. Mach. Intell.*, 24(5):603–619.

[Comaniciu and Meer, 2002b] Comaniciu, D. and Meer, P. (2002b). Mean shift: A robust approach toward feature space analysis. *PAMI*, 24(5):603–619.

[Duffy and Crowley, 2000] Duffy, N. and Crowley, J. L. (2000). Object detection using colour. In *ICPR 2000*, Barcelona.

[Duygulu et al., 2002] Duygulu, P., Barnard, K., de Freitas, J. F. G., and Forsyth, D. A. (2002). Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary. In *Proceedings of the 7th European Conference on Computer Vision-Part IV*, pages 97–112. Springer-Verlag.

[Feng et al., 2004] Feng, S., Lavrenko, V., and Manmatha, R. (2004). Multiple bernoulli relevance models for image and video annotation. *CVPR'04*.

[Hare and Lewis, 2005] Hare, J. S. and Lewis, P. H. (2005). Saliency-based models of image content and their application to auto-annotation by semantic propagation (speech). *Proceedings of Multimedia and the Semantic Web*.

[Harris and Stephens, 1988] Harris, C. and Stephens, M. (1988). A combined corner and edge detector. In *Proceedings of the 4th Alvey Vision Conference*, pages 147–151.

[Hofmann and Puzicha, 1998] Hofmann, T. and Puzicha, J. (1998). Statistical models for co-occurrence data. *A.I. Memo No. 1625*.

[Inoue, 2004] Inoue, M. (2004). On the need for annotation-based image retrieval. *Workshop on Information Retrieval in Context*.

[Jeon et al., 2003] Jeon, J., Lavrenko, V., and Manmatha, R. (2003). Automatic image annotation and retrieval using cross-media relevance models. *ACM SIGIR 2003*.

[Lavrenko and Croft, 2001] Lavrenko, V. and Croft, W. B. (2001). Relevance-based language models. In *Research and Development in Information Retrieval*, pages 120–127.

[Lavrenko et al., 2003] Lavrenko, V., Manmatha, R., and Jeon, J. (2003). A model for learning the semantics of pictures. *Proceedings of the 16th Conference on Advances in Neural Information Processing Systems NIPS*.

[Li and Wang, 2003] Li, J. and Wang, J. Z. (2003). Automatic linguistic indexing of pictures by a statistical modeling approach. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(9):1075–1088.

[Lim and Jin, 2005] Lim, J.-H. and Jin, J. S. (2005). A structured learning framework for content-based image indexing and visual query. In *Multimedia System*, volume 10, pages 317–331. Springer-Verlag.

[Lowe, 1999] Lowe, D. G. (1999). Object recognition from local scale-invariant features. In *Proc. of the International Conference on Computer Vision ICCV, Corfu*, pages 1150–1157.

[Maillot, 2005] Maillot, N. E. (2005). *Ontology Based Object Learning and Recognition*. PhD thesis, Universite de Nice - Sophia Antipolis.

[Marr, 1983] Marr, D. (1983). *Vision: A computational investigation into the human representation and processing of visual information*. W. H. Freeman.

[Mikolajczyk and Schmid, 2002] Mikolajczyk, K. and Schmid, C. (2002). An affine invariant interest point detector. In *ECCV*, pages 128–142. Springer. Copenhagen.

[Monay and Gatica-Perez, 2004] Monay, F. and Gatica-Perez, D. (2004). Plsa-based image auto-annotation: Constraining the latent space. ACM, Proc. ACM Int. Conf. on Multimedia (ACM MM).

BIBLIOGRAPHY

[Mori et al., 1999] Mori, Y., Takahashi, H., and Oka, R. (1999). Image-to-word transformation based on dividing and vector quantizing images with words.

[Reed and Wechsler, 1990] Reed, T. R. and Wechsler, H. (1990). Segmentation of textured images and gestalt organization using spatial/spatial-frequency representations. *IEEE Trans. Pattern Anal. Mach. Intell.*, 12(1):1–12.

[Schmid and Mohr, 1997] Schmid, C. and Mohr, R. (1997). Local greyvalue invariants for image retrieval. *Pattern Analysis and Machine Intelligence*, 19(5).

[Shi et al., 1998] Shi, J., Belongie, S., Leung, T., and Malik, J. (1998). Image and video segmentation: The normalized cut framework. *IEEE Int'l Conf on Image Processing*, pages 943–947.

[Smeulders et al., 2000] Smeulders, A. W. M., Worring, M., Santini, S., Gupta, A., and Jain, R. (2000). Content-based image retrieval at the end of the early years. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(12):1349–1380.

[Srikanth et al., 2005] Srikanth, M., Varner, J., Bowden, M., and Moldovan, D. (2005). Exploiting ontologies for automatic image annotation. In *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 552–558, New York, NY, USA. ACM Press.