IDIAP RESEARCH REPORT

Learning the structure of image collections with latent aspect models

Florent Monay [a]

IDIAP–RR 07-06

March 21, 2007

published in
École polytechnique Fédérale de lausanne

[a] IDIAP Research Institute, CP 592, 1920 Martigny, Switzerland, florent.monay@idiap.ch

IDIAP Research Institute **www.idiap.ch**
Rue du Simplon 4 P.O. Box 592 1920 Martigny – Switzerland
Tel: +41 27 721 77 11 Fax: +41 27 721 77 12 Email: info@idiap.ch

IDIAP Research Report 07-06

Learning the structure of image collections with latent aspect models

Florent Monay

March 21, 2007

published in
École polytechnique Fédérale de lausanne

Abstract. The approach to indexing an image collection depends on the type of data to organize. Satellite images are likely to be searched with latitude and longitude coordinates, medical images are often searched with an image example that serves as a visual query, and personal image collections are generally browsed by event. A more general retrieval scenario is based on the use of textual keywords to search for images containing a specific object, or representing a given scene type. This requires the manual annotation of each image in the collection to allow for the retrieval of relevant visual information based on a text query. This time-consuming and subjective process is the current price to pay for a reliable and convenient text-based image search.

This dissertation investigates the use of probabilistic models to assist the automatic organization of image collections, attempting to link the visual content of digital images with a potential textual description. Relying on robust, patch-based image representations that have proven to capture a variety of visual content, our work proposes to model images as mixtures of *latent aspects*. These latent aspects are defined by multinomial distributions that capture patch co-occurrence information observed in the collection. An image is not represented by the direct count of its constituting elements, but as a mixture of latent aspects that can be estimated with principled, generative unsupervised learning methods. An aspect-based image representation therefore incorporates contextual information from the whole collection that can be exploited. This emerging concept is explored for several fundamental tasks related to image retrieval - namely classification, clustering, segmentation, and annotation - in what represents one of the first coherent and comprehensive study of the subject.

We first investigate the possibility of classifying images based on their estimated aspect mixture weights, interpreting latent aspect modeling as an unsupervised feature extraction process. Several image categorization tasks are considered, where images are classified based on the present objects or according to their global scene type. We demonstrate that the concept of latent aspects allows to take advantage of non-labeled data to infer a robust image representation that achieves a higher classification performance than the original patch-based representation. Secondly, further exploring the concept, we show that aspects can correspond to an interesting soft clustering of an image collection that can serve as a browsing structure. Images can be ranked given an aspect, illustrating the corresponding co-occurrence context visually. In the third place, we derive a principled method that relies on latent aspects to classify image patches into different categories. This produces an image segmentation based on the resulting spatial class-densities. We finally propose to model images and their caption with a single aspect model, merging the co-occurrence contexts of the visual and the textual modalities in different ways. Once a model has been learned, the distribution of words given an unseen image is inferred based on its visual representation, and serves as textual indexing. Overall, we demonstrate with extensive experiments that the co-occurrence context captured by latent aspects is suitable for the above mentioned tasks, making it a promising approach for multimedia indexing.

Acknowledgments

L'essentiel est invisible pour les yeux.
Antoine de Saint-Exupéry, Le Petit Prince

The work presented in this dissertation was made possible with the precious help and support from a number people. They have all contributed in different ways to make my research and my life easier during the last four years. I will try to mention all of them, but I will certainly not be exhaustive. Please, do not be offended if you think that you should be mentioned in the following lines and you are not.

First, I would like to thank my supervisor, Daniel, for sharing his expertise in this line of research with me. Without his help and guidance, I certainly would have become lost in the darkness of the first year of research. Daniel accomplished real miracles, turning my first paper drafts into something readable, and eventually acceptable for publication. His patience and perseverance helped me to meet deadlines that would have been impossible without his support.

A fruitful collaboration with Pedro and Jean-Marc gave birth to the idea of combining state-of-the-art image representations with the concept of latent aspect models for images, opening up new perspectives to long-standing vision problems. A number of interesting meetings and informal discussions made this outcome possible, and certainly brought a lot of value to this dissertation. I am very grateful to them for their collaboration. I also thank Samy for correcting my thesis proposal, David for making my name appear in machine learning oriented publications, and Kevin and Marc for their precious English support.

The most important moment of the last four years was not related to my research. I had the great pleasure of marrying Valérie two years ago, who was not discouraged by the evenings and nights that I have sometimes spent writing about baseline experiments or tweaking LaTeXfiles. I sincerely thank her for her patience and understanding. Finally, thanks to all of my family, who contributed to make my life more comfortable. I was very fortunate to benefit from their support throughout my (seemingly endless) studies.

Contents

Chapter 1

Introduction

1.1 Retrieving images with keywords

The value of a media collection relies equally on the quality and the accessibility of its content. If the billions of existing webpages were not continuously indexed by efficient search engines, a lifetime would not suffice to browse through this unorganized information, and relevant webpages would be unreachable. Off-line webpage indexing systems such as Google's PageRank allow the formulation of intuitive text-based queries, and make the retrieval of relevant documents practically independent of the number of documents to search.

With the production of large digital image collections, favored by increasingly cheap digital recording and storage devices, there is a similar need for efficient indexing and retrieval systems for images. Digital photo collections constantly grow in size, and this information needs to be consequently organized to be accessible. To illustrate the size of current photo collections, we have computed basic statistics from the Flickr (www.flickr.com) photo-sharing website in July 2006. Flickr is a popular photo-sharing service that allows its registered users to upload and share their images on the World Wide Web. According to Newsweek magazine, Flickr was a 2.5 million-member community in April 2006 [38], which gives an estimate of 1.2 billion photos. The histogram of the number of public photos from $21'000$ users is shown in Figure 1.1, for an average number of 480 photos per collection. This photo-sharing website is emblematic of current and also future online image collections. They consist of a huge number of photos that have to be organized and indexed in order to be accessible. The ideal retrieval system should allow for intuitive search for the user, and require a minimal amount of human interaction to be applicable to large collections. Two approaches, based on distinct query types, coexist in the multimedia information retrieval literature.

One is based on the *query-by-example* (QBE) paradigm [60, 80, 84, 13, 79]. In QBE systems, various low-level visual features are preliminarly extracted from the data set and stored as image indices. The query is an image example that is indexed by its low-level visual features, and images are ranked with respect to their similarity to this query index. Given that indices are directly derived from the image content, this process requires no semantic labeling, thus no human interaction. The QBE paradigm is therefore an interesting solution for particular image retrieval tasks such as medical imaging [19], satellite images [39], or personal photo collections [50, 27]. These data sources tend however to be specific, as the corresponding QBE solutions are too.

In general, however, no image example is available at the time of querying, and the formulation of a more intuitive textual query is widely preferred. Commercial image collections such as Getty images (www.gettyimages.com) and Corbis (www.corbis.com) are for instance designed to be searched with text-based queries. Their customers want to retrieve photos related to a given event, location, or concept, and these queries are naturally expressed by short textual queries. Despite the development of systems and tools to assist it, manual annotation involves a substantial amount of work, and often results in heavy costs. One solution, when applicable, is a collaborative annotation system. For

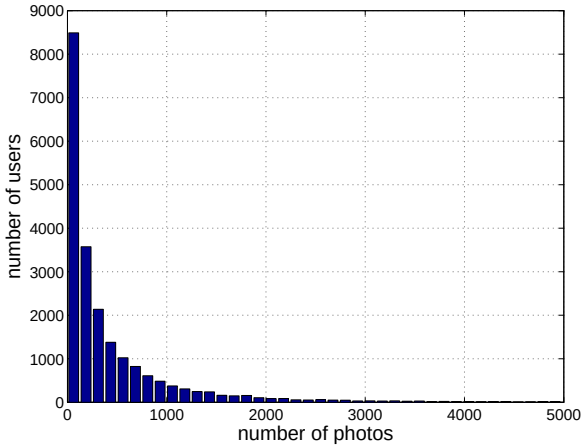
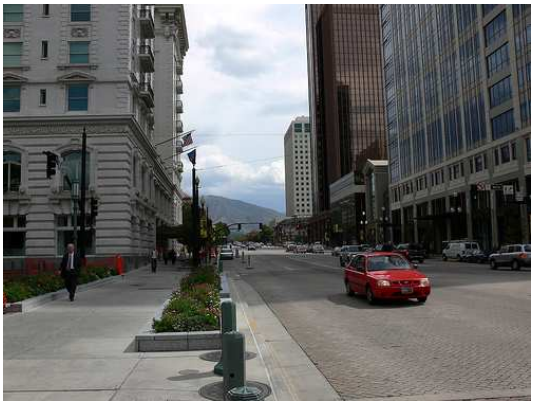


Figure 1.1: Number of users vs. number of photos in collection for 21′000 users from the *Flickr.com* website. This sample corresponds to an average of 480 photos per user that need to be organized for accessibility.

example, Flickr takes advantage of its community members as potential annotators. Each user is invited to attach tags to his own photos to organize his collection, simultaneously allowing other users to access this collection.

Image annotation is however a subjective process: an image represents a large number of potential words from which the annotator has to choose. Without any constraint, caption words can be sampled from many semantic levels, as shown with the four annotated examples from Flickr on Figure 1.2: they can refer to the global scene type (e.g. *city street*, *kitchen*, *ocean*), name particular entities in the image (e.g. *mountains*, *tree*, *leaf*), specify geographic locations (e.g. *Salt Lake City*, *Mexico*), describe the main color distribution (e.g.*yellow*), describe the related event (e.g. *birthday*), or even refer to objects that are not present, but only suggested by the image (e.g. *cake*). Depending on their personal visual interpretation, an image will likely be annotated differently by two individuals: their respective annotation depends on a set of subjective decisions. And given that queries are unknown at the time of annotation, relevant images can be missed because the resulting subjective caption is non-exhaustive. The words *sky*, *buildings*, or *car* are not in the caption of the image (a) on Figure 1.2, although the image contains these instances. Similarly, the word *rock* is not part of the tags attached to the image (b), because the tag *yellow* has been chosen instead. Other high-level descriptions could be considered: images (a)-(b) could be labeled as *outdoor*, and image (d) as *indoor*.

We see on Figure 1.2 that some form of correspondence implicitly exists between the visual content of images and the words that were chosen to describe it. This correspondence is however only partially defined, because words are not explicitly attached to a given region in the image. For instance, the word *tree* in image (c) only relates to an unspecified image region. Similarly, the word *mountains* only describes a small region in the center of image (a). Caption words can also relate to the whole image: *city street* for image (a), and *kitchen* for image (d). If some form of mapping between words and local image characteristics could be learned, the prediction of a word caption given an unannotated image would be possible. Learning this relationship is however a real challenge given the variety of concepts to consider, and the loose correspondence provided between the textual description and the visual content.



(a) Utah, Salt Lake City, mountains, city street

(b) cabo, San, Lucas, Mexico, sky, ocean, landscape, yellow

(c) Japan, Kamogawa, kitten, tree, leaf

(d) messy kitchen, kitchen, cake, birthday

Figure 1.2: Four images and their annotation tags from the Flickr website. Tags are attached by the image owner, resulting in a subjective description of the image content. A significant number of foreground objects, for instance, are not necessarily mentioned.

1.2 Problems addressed

In this dissertation, we investigate the use of local image representations and probabilistic models to assist the automatic organization of image collections, attempting to link the visual content of digital images with a textual description. Relying on patch-based, discrete image representations that have proven to capture a variety of visual content, our work proposes to tackle the problem with the help of *latent aspect models*. An image is not represented by the direct count of its constituting elements, but as a mixture of latent aspects - modeled with multinomial distributions - that capture co-occurrence of patches in the collection. An aspect-based image representation therefore incorporates contextual information from the whole collection that we exploit for several fundamental tasks related to image retrieval.

- Using the aspect mixture weights, estimated for each image, as an image representation for image classification. Relying on a histogram of quantized image regions as initial image representation, a latent aspect model allows to identify distinctive patterns of co-occurrence from unlabeled data, improving the classification of images in scene and object categories.
- Ranking the images based on the probability of the images in a dataset given each aspect. This

produces a soft clustering of an image collection based on the co-occurrence patterns of quantized patches identified by the aspect model, that can correspond to concrete visual categories in the collection.

- Classifying image patches in images to obtain a form of sparse image segmentation, taking advantage of the co-occurrence context identified by the aspect model. This further illustrates the possible match between latent aspects and concepts in images in the case of image segmentation.
- Predicting a word distribution given an unannotated image for textual indexing, relying on the aspect distribution inferred from the visual representation. Three ways of learning a mixture of aspects from both the image description and the discrete local image representation are investigated.

These lines of research are successively addressed throughout the thesis, investigating new implications of the concept of mixture of aspects for images in each chapter. We build the analysis sequentially, relying on the results from the previous investigations to tackle the next task.

1.3 Contributions and organization of the thesis

Our research has addressed each of the points listed in the previous Section and resulted in a number of contributions. Each of them is presented in a chapter form, as follows [1].

Latent aspect models for text and images. Chapter 2 has two goals: explaining the motivations behind the *latent aspect models* family [29, 7, 10], and providing an intuition of their choice to model visual information. The use of latent aspect models for images is put in perspective with the concept of a patch-based image representation.

Aspect-based image classification and clustering. Representing an image as a mixture of latent aspects can be seen as a second-level feature extraction process. We explore this idea in Chapter 3, proposing an approach to classify objects [54] and scenes [68] based on their aspect mixture weights. This results in state-of-the-art performance, and we demonstrate that the aspect model allows to infer a more effective image representation from non-labeled, auxiliary data. Using the aspects as soft clusters, we also prove that the aspects can serve as a browsing structure, because they can correspond to semantic concepts.

Mapping latent aspects and image regions. We show in Chapter 4 that, although no spatial information about the position of local patches is used, latent aspects can correspond to particular image regions defined by patch co-occurrences [55]. We explore this idea on a natural vs. man-made scene segmentation task, deriving a principled method to classify the local image patches based on the latent aspect model. The results show that the co-occurrence context captured by the aspects allows to improve the classification of the quantized patches. Also, the more standard spatial context information, generally exploited for image segmentation, can conveniently be combined to the co-occurrence context.

Modeling semantic aspects for cross-media image indexing. In Chapter 5, we propose a number of principled approaches to learn a latent aspect model from the patch-based representation of an image and its text caption. Different ways of modeling the co-occurrence between text and images are possible. Given that keywords have context among themselves - the meaning of a keyword depends on the context it appears in - there should be a correlation between these

[1] The work in Chapters 3 and 4 was done in collaboration with Pedro Quelhas. I was initially focusing on latent aspect modeling, and Pedro brought his expertise on local descriptors. The resulting ideas and work presented in these two chapters was divided equally between the two of us. A number of experiments and discussions for the evaluation of different combinations of point detectors and descriptors discussed in [67] are not presented in Chapter 3, as these results represent a contribution from Pedro's own work.

feature-based and keyword-based co-occurrence contexts. This approach is used to infer annotation for unseen images, and compared with other state-of-the-art methods [53]. The results show that if the aspect mixtures are estimated based on the textual information in the training phase, the inferred annotations allow an improved retrieval performance.

Chapter 2

Latent aspect models for text and images

Within this chapter, we motivate and detail the probabilistic latent aspect model proposed by Hofmann [29]. Our discussion first focuses on the aspect model in the context of text collections, which is its original purpose. We then extend the use of this model to the case of image collections, showing that the concept of latent aspect is also adequate in the case of visual information. This image model will be used throughout our work.

2.1 Limitations of the vector space model for text

The *vector space model* [71, 70] is the widely agreed representation method for text collections. It summarizes a document into a N-dimensional vector that encodes the count of each vocabulary word w in this document. This is equivalent to treating a text document as a *bag-of-words*, where each word has been sampled from a vocabulary of size N. Discarding the punctuation information and the word ordering, the bag-of-words assumption is an important simplification of the original text document that casts documents of different lengths into a vectorial form. Only word stems are usually considered to build the vocabulary, because variations due to plural forms and verb conjugation are not considered as informative. Another common processing consists in discarding words from a *stop word* list, as they do not contain discriminative information (e.g. *above*, *again*, *where*). A text document d_i is represented by the count of its constituting word stems w_j, expressed by $n(w_j, d_i)$.

The intuition behind the vector space model is simple: texts about different topics have a distinct vectorial representation, because different words are used to express these topics. The notion of similarity between texts, essential for any information retrieval tasks, is implicitly defined on a word basis; texts sharing many words are more similar than texts with only a few words in common. The similarity between a document d_i and a query q is computed as the cosine of the angle between their respective bag-of-words representations, usually weighted by:

$$sim_{cos}(q, d_i) = \frac{\sum_j^N n(w_j, q) n(w_j, d_i)}{\sqrt{\sum_j^N n(w_j, q)^2} \sqrt{\sum_j^N n(w_j, d_i)^2}}, \tag{2.1}$$

Please note that the term frequency and inverse document frequency weighting $tfidf$ generally replaces the word count to prevent a bias towards long text documents, and to minimize the influence of very frequent words in the text corpus. The term frequency tf is defined as the count of the word w_j in the document d_i devided by the total number of words in this document, and the inverse document frequency idf is defined as the logarithm of the number of documents in the corpus M divided by the

number of documents in which the word w_j appears:

$$tfidf(w_j, d_i) = \underbrace{\frac{n(w_j, d_i)}{\sum_l n(w_l, d_i)}}_{tf} \underbrace{log \frac{M}{|n(w_j, d) > 0|}}_{idf} .$$

With or without this $tfidf$ weighting, basic information retrieval tasks are efficiently implemented as simple linear algebra operations in a vector space, what explains the popularity of this approach [4]. Documents from a collection can be ranked based on their similarity with a text query, and documents can be clustered based on their similarity.

The simplicity of the vector space model has limitations. Word-based similarity, although efficient in general, can result in ambiguities [29]. Texts should ideally be compared at their topic level, and not based on the specific words that were chosen to express these topics. For instance, the fact that the term *space* appears many times in the previous paragraphs should not make this chapter closer to documents about space research. This word should be interpreted in its context to avoid this ambiguity: the term *space* co-occurs with the words *vector*, *model*, *linear* and *algebra* in the previous discussion, thus defining a related *linear algebra* topic. Models that attempt to capture this type of co-occurrence information from the bag-of-words representation have therefore attracted interest in the recent information retrieval literature.

2.2 Modeling text documents as mixtures of latent aspects

Various probabilistic models [29, 7, 10] have been proposed to model topics in a text collection, moving beyond the standard bag-of-words approach. These models are based on the same idea of capturing patterns of word co-occurrences given latent variable with multinomial distributions. A document is not represented as an isolated set of word occurrences, but modeled as a mixture of word co-occurrences given a latent aspect observed within the collection. These word co-occurrences, modeled with multinomial distributions over the vocabulary words, can be interpreted as *topics* existing in the text collection [29, 7, 10, 49, 82]. The interpretation of multinomial word distributions as topics gives a good intuition of the structure identified in the text collection, and this family of probabilistic models is, therefore, often referred to as *topic models* [82]. Other names have been used (e.g. *discrete component analysis* [10]), but we prefer the denomination of *aspect models* [49]. The word *aspect* has the advantage of not being text specific, and can refer to multinomial over words and any other type of discrete data.

Modeling texts with mixtures of aspects is a promising approach to handle the *synonymy* and the *polysemy* issues that penalize the performance of bag-of-words text retrieval systems. On one side, a text query will not necessarily match the words of a relevant document whenever different words have been chosen to express the same idea. Documents that do not contain the exact same words as the query, but synonyms, will not be retrieved. Conversely, polysemy makes irrelevant documents similar to a text query if the same terms are used to express different concepts. Aspect models therefore aim at identifying a disambiguated representation of text documents learned from their bag-of-words representation.

We illustrate the concept of latent aspects for text documents in Figure 2.1, showing the decomposition of a document into a mixture of 300 latent aspects. The text document is taken from the Reuters-21578 text dataset (www.daviddlewis.com/resources/testcollections/reuters21578/), a collection of documents from the 1987 Reuters newswire. The original text document is shown on the top-left of Figure 2.1. Its corresponding bag-of-words representation, after word stemming and stopping, is shown on the middle-left as a word histogram over a vocabulary of approximately 18000 word stems. The mixture of aspects for this document, identified with the aspect model presented later in this chapter in Section 2.4, is shown on the bottom-left of the figure. From this mixture, the three aspects with the higher weights are shown on the right of Figure 2.1, characterized by their multinomial distributions over words. Observing the multinomial distributions and their most probable words gives

The United States and Japan are not involved in a trade war, despite U.S. sanctions announced last week against Japanese semiconductors, U.S. Trade Representative Clayton Yeutter said. "In my judgement, we're not even close to a trade war," Yeutter told a House Agriculture Committee hearing. Yeutter said if Japan takes action to honor its agreement with the U.S. on semiconductor trade, "Then the retaliatory response will last a relatively short period." Yeutter said Japan must stop dumping chips in third countries and buy more American computer chips.

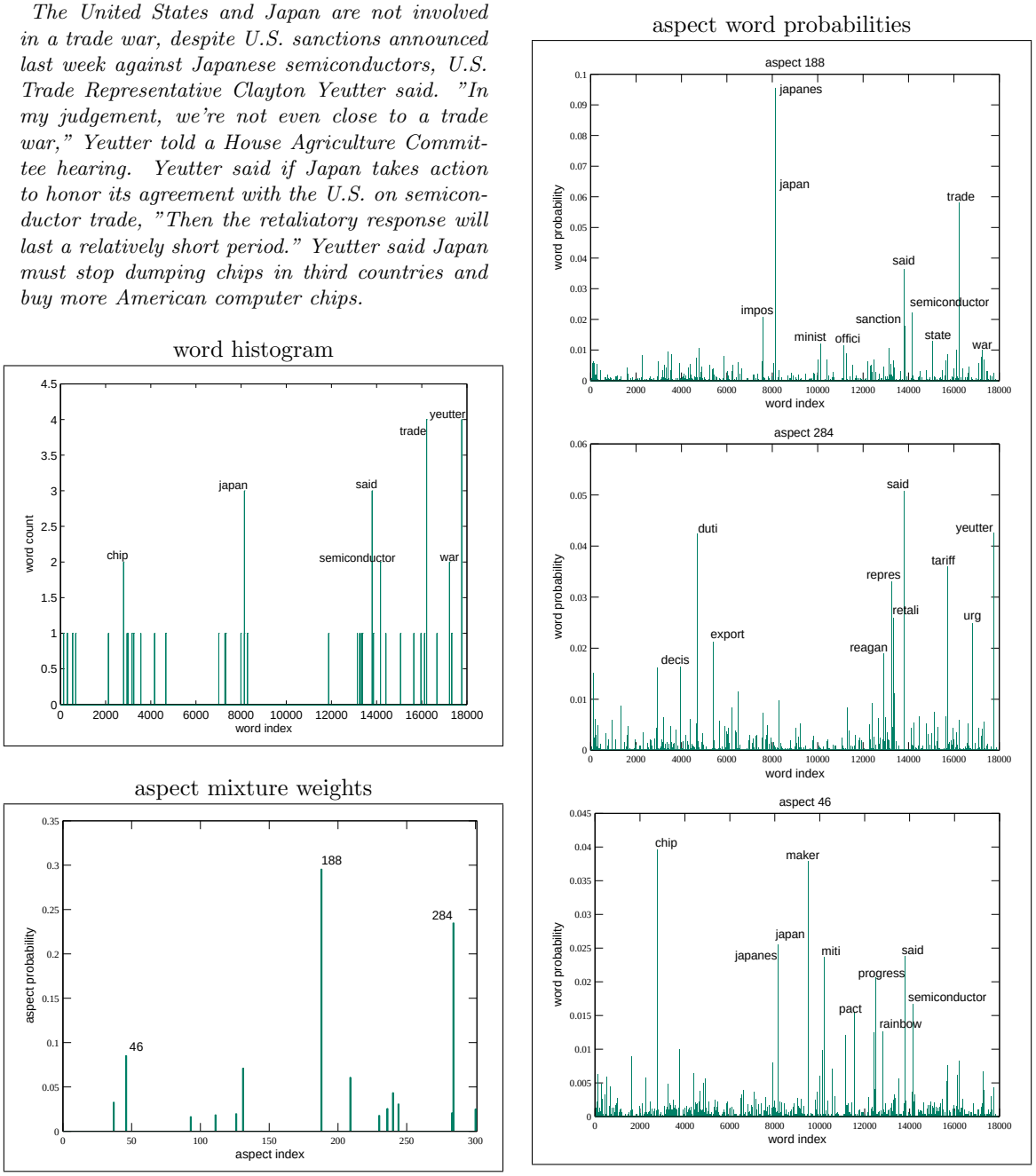


Figure 2.1: Representation of a text from the Reuters-21578 dataset as a mixture of aspects. The original document is shown on the top-left, the corresponding bag-of-words histogram is shown on the middle-left, the aspect decomposition is shown on the bottom-left, and the multinomial distributions over words of the three most probable aspects are shown on the right. As can be interpreted based on their most probable words, these three aspects are related to specific topics: aspects number 188 seems related to issues regarding economical relationships with Japan; aspect number 284 contains words related to international United States trade (Clayton Yeutter was the US trade representative at the time the document was published); and aspect number 46 seems related to the Japanese semiconductor industry.

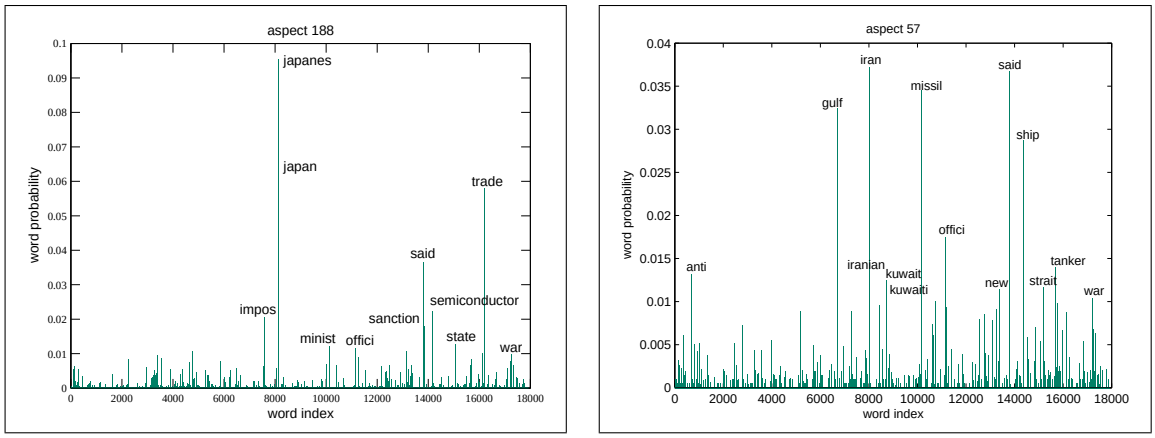


Figure 2.2: Two aspects learned on the Reuters-21578 text collection that include *war* as a highly probable word. Aspect 188 (left) seems related to US economical relationships with Japan, while aspect 57 (right) contains terms related to military tension in the Arabian Gulf.

an intuition of what the aspect thematics are: aspect number 188 seems related to issues regarding economical relationships with Japan; aspect number 284 contains words related to international United States trade (Clayton Yeutter was the US trade representative at the time the document was published); and aspect number 46 seems related to the Japanese semiconductor industry.

Under the aspect model assumption, words are not attributed to one aspect exclusively, but assigned to each aspect with a given probability in the corresponding multinomial distribution. For instance, the word *said* is highly probable for all three aspects on the right of Figure 2.1, and the terms *japanes* and *japan* are in the top-ten words of both aspects 188 and 46. The latent aspect approach is therefore very different from basic word clustering, because it allows terms to belong to several aspects with a given probability. For instance, the term *war* is used in a very specific context in the document (trade), but other documents in the collection contain the word *war* used in another context (military conflict). Two aspects learned from the text collection, and capturing different contexts of the word *war* are shown on Figure 2.2 to illustrate how aspect models can take this into account. Aspect 188, as already seen, is related to the topic of economical relationships between the US and Japan. Aspect 57 corresponds to a different contextual use of the word *war*, defined by the highly probable terms *iran*, *missile*, *gulf*, *ship*, *kuwait*, etc. The soft clustering of words with respect to aspects, which is the core idea of aspect models, allows to model several co-occurrence contexts for the same word. The bag-of-words representation in our example makes aspect 188 much more likely than aspect 57, what seems a correct interpretation of the word *war* in the document.

Modeling aspects by multinomial word distributions also allows to tackle synonymy ambiguities. For instance, *japan* and *japanes* coexist in our vocabulary of the Reuters-21578 text collection, and can be seen as synonyms, created by a non-optimal stemming algorithm. In other words, a query containing the word *japanese* will not match the bag-of-words representation of a document containing the word *japan*, which is counter-intuitive. Capturing the probability of word co-occurrence with a latent aspect can link two synonyms: on Figure 2.1, *japan* and *japanes* are both highly probable given aspects 188 and 46. A mixture of aspects containing aspect 188 or 46 will implicitly make both the words *japan* and *japanese* probable in the corresponding document, even is only one of them is present.

The structure of aspect models, based on multinomial word distributions, make them suitable to handle polysemy and synonymy ambiguities in text collections as we have seen. Their use is however not restricted to text collections, and they have been investigated for a variety of tasks, from music genre classification [3] to the analysis of voting records [11]. The concept of mixture of aspects is in particular very promising for modeling visual information in image collections, as discussed in the

following section.

2.3 Modeling images as mixtures of latent aspects

An image contains an almost endless amount of visual information that can be derived from its pixel representation. Depending on the task, various types of image representation have been proposed to capture a priori relevant information. Global color histograms, for instance, conveniently represent an image with its color distribution. Being intrinsically robust to a variety of geometric transformations, this compact representation appears to be efficient for a number of image retrieval tasks [69], where color information is discriminant. However, a color histogram will not allow to discriminate between images sharing a similar color distribution. An example of a more elaborate image representation, tailored to a very specific task, is the representation based on *spatial envelope properties* for the categorization of image scenes [61]. The spatial envelope describes the degree of some global properties from the image (naturalness, openness, roughness, ...), that are estimated based on the analysis of the image spatial frequencies. A good discrimination of different scene types is possible with this representation.

To model a heterogeneous image collection, image representations designed for the classification of specific objects are not adequate. We need an image representation that allows to deal with different image scales, while still capturing sufficient information about the various concepts that have to be modeled. We therefore consider a family of image representations that is intended to be more generic, and therefore suitable for information retrieval tasks. These representations have received various names in the literature, such as *blobs* [18], *textons* [30], *parts* [2], *visual words* [76], or *visterms* (for visual term) [68, 32], but are all based on the same principle: the definition and quantization of local image regions or patches. In this dissertation, we use the word *visterm* when referring to a quantized version of image regions. The image regions can be sampled uniformly given a fixed grid layout [57, 32], result from a principled image segmentation algorithm [18], or be identified at different locations and scale using a point detector [2, 68, 30]. Depending on what visual information is a priori relevant for a given task, a variety of descriptors can be considered to depict these regions. Color, texture, shape, or a combination of these features are generally considered as relevant information. The region descriptors are then quantized into a fixed set of possible visterms v_j, that can be seen as a *visual vocabulary*. Mapping similar image regions onto the same visterm simplifies the pixel-based representation of an image d_i into a fixed-size histogram of the visterm count $n(v_j, d_i)$.

To give a first intuition of what type of visual information can be captured by a histogram of visterms, we have constructed the two examples shown on Figure 2.3. We have recourse to these constructed examples to simplify the discussion: the actual visterm construction is described in details in each related chapter. In Figure 2.3, the image regions are sampled from a uniform grid and represented by their color distribution and texture. Visually similar regions are quantized into the same visterm, and images are represented as a count of co-occurring visterms describing their content. The comparison of the two image representations shows the type of information that can be captured by a visterm histogram. Both images contain regions that are quantized into visterm #1, which corresponds to greenish regions covering *grass* and *tree* parts of the images. Visterm #2, #3 and #22 contains blue and white colors corresponding to *sky+clouds* regions occurring in the two images. *Building* regions from the two images are quantized into 7 visterms (#8, #9, #10, #11, #12, #14, and #16). The *building* regions in the first image are mapped onto 6 of these visterms (#9, #10, #11, #12, #14 and #16), while *building* regions in the second image are quantized into 4 visterms (#8, #9, #12, #14). We see that the parts of the second image containing *sea* are quantized into 3 different visterms (visterm #4, #18 and #21), which are almost not represented in the first image. Globally, we see that visterm histograms allow an intuitive comparison of the visual content of their respective images. The two images in Figure 2.3 contain *tree*, *grass*, *sky* and *buildings*, and this similarity in visual content is reflected by the partial overlap of their respective visterm histograms. The absence of *sea* in the first image translates in the empty visterms #4, #18, and #21.

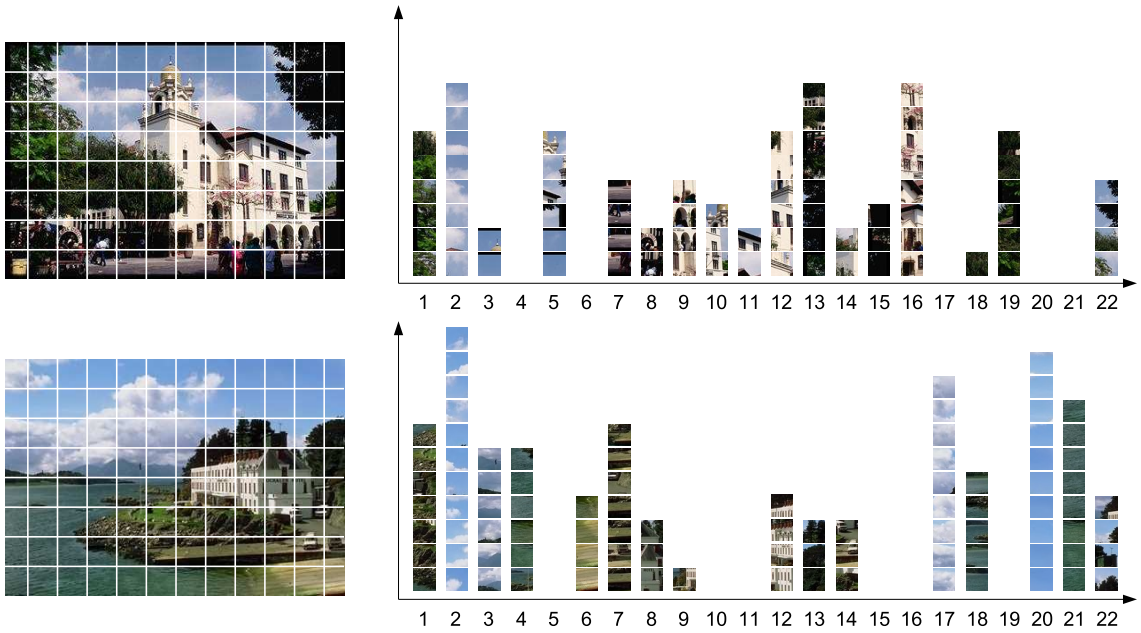


Figure 2.3: Representing an image as a visterm histogram: (left) in this case, image regions are defined by a uniform grid; regions are quantized into a finite set of $N = 22$ visterms; (right) each image is represented by the histogram of its constituting visterms.

A number of similarities exist between the bag-of-words representation for text and the visterm histogram for images. A histogram of visterms discards the spatial layout of the image regions, what is comparable to the bag-of-words simplification of a text document (words are treated as unordered information). This justifies the *bag-of-visterms* denomination used in the rest of the dissertation. Visterm ambiguities, related to the one discussed for words in section 2.2, affect the bag-of-visterms representation. The quantization of the image regions allows to obtain a convenient histogram-based image representation, but at the same time creates ambiguities between the resulting quantized patches. A given visterm can represent a variety of visual contents, what is equivalent to the polysemy issue observed for words. In Figure 2.3, the patch #1 corresponds to both *tree* and *grass* regions in the original images. Similarly, the patch #13 corresponds to both *building* and *tree* parts. Furthermore, several visterms can represent the same concept: in the example, the *sky* regions are clearly splitted into 6 visterms (#2, #3, #5, #17, #20 and #22), depending on the respective amount of *cloud* they contain. Similarly, *building* parts are mapped onto various visterms depending on their respective texture and color distribution (#8, #9, #10, #12, #14 and #16).

The decomposition of the bag-of-visterms representation into a mixture of visterm distributions, similarly to what is done in the text case, is a possible approach to solve the intrinsic visterm ambiguities. Within a dataset of images containing *buildings*, *sky*, *sea*, or *vegetation* regions, interesting visterm co-occurrence patterns related to these concepts are likely to exist. These patterns can be modeled by multinomial distributions over visterms, characterizing a visual latent aspect of the collection. As an illustration, in Figure 2.4 we show four visterm distributions that could be identified in such a dataset. Four multinomial distributions over the same visterm vocabulary as the one used for Figure 2.3 are shown in Figure 2.4, characterizing four visual aspects. Representative image patches are shown on the x-axis to indicate what each of the 22 visterms correspond to. The distribution in Figure 2.4 (a) corresponds to high probabilities for *sky*-related visterms, likely to co-occur throughout an image collection in which *sky* regions exist. Such a multinomial distribution would nicely model the visterm distribution corresponding to *sky* regions in the two images in Figure 2.3. The distributions

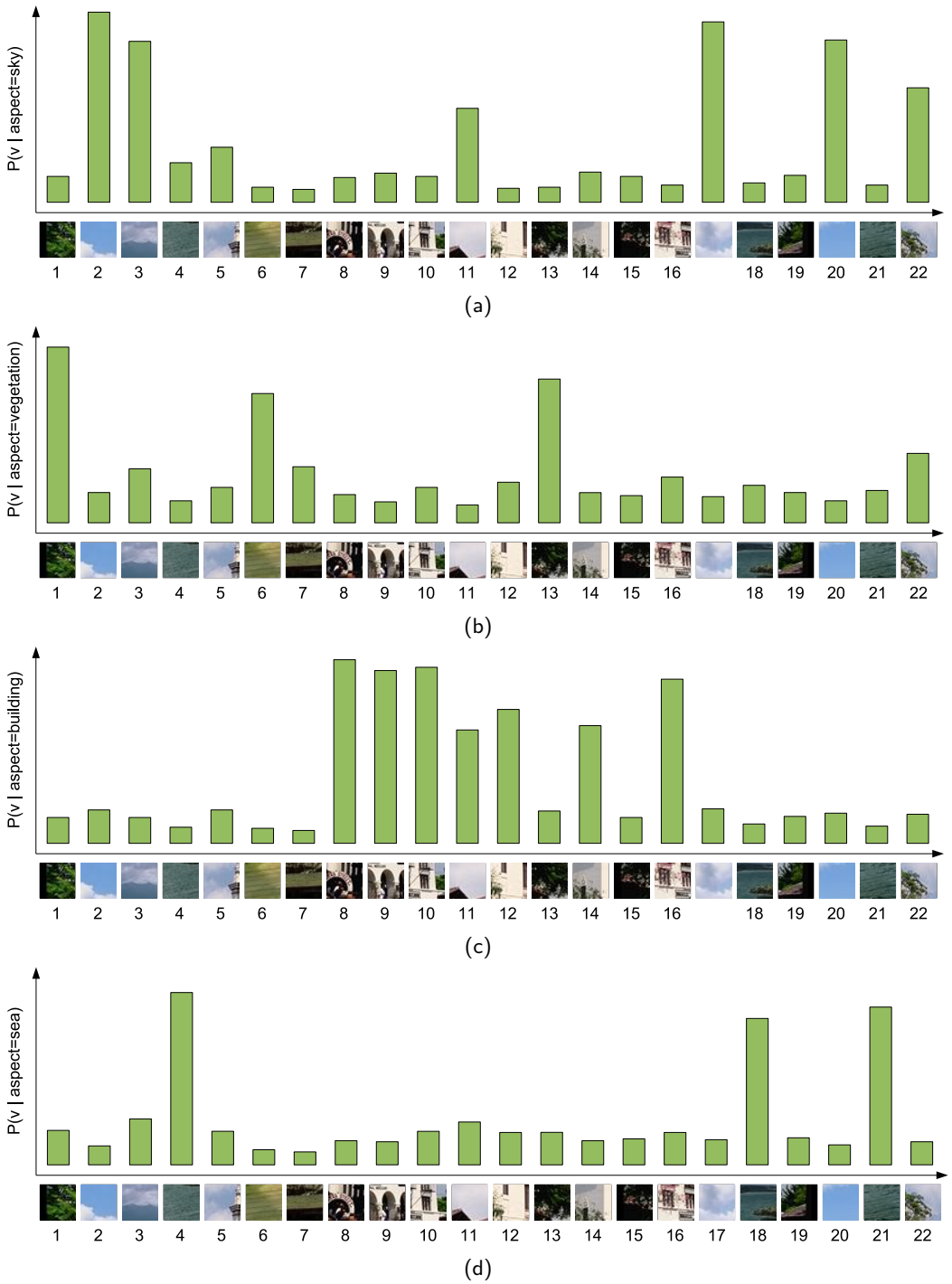


Figure 2.4: Four multinomial distributions over visterms, with higher probabilities for specific types of image regions: (a) *sky*-related visterms (visterms #2, #3, #17, #20, #22); (b) *vegetation*-related visterms (visterms #1, #6, #13); (c) *buildings*-related visterms (visterms #8, #9, #10, #12, #14, #16); (d) *sea*-related visterms (visterms #4, #18, #21). Representative image patches are shown on the x-axis to indicate what the 22 visterms correspond to.

in Figure 2.4 (b) and (c) show a predominance of *vegetation* and *buildings* visterms respectively. The *vegetation* and *buildings* regions from the two images in Figure 2.3, mapped onto their corresponding visterms, could be respectively modeled by these two visual aspects. As for the *sea*-related visterms, only present in the second image, they could be modeled by the visterm distribution in Figure 2.4 (d). The two bag-of-visterms in Figure 2.3 can thus be modeled as a mixture of these four aspects, defined by multinomial distributions over visterms, with different mixture weights reflecting their actual content. The first image in Figure 2.3 is a mixture of the *sky*, *building* and *vegetation* aspects. The second image is a mixture of the same aspects, but also includes the *sea* aspect, defined by the visterm distribution in Figure 2.4 (d).

From this basic example, the potential benefits of decomposing an image into a mixture of latent aspects are clarified. If the number of aspects is chosen to be smaller than the size of the visterm vocabulary, creating an image representation based on aspect mixture weights also represents a dimensionality reduction of the bag-of-visterms representation. At the same time, the aspect-based image representation implicitly incorporates information from other images in the collection, in the form of visterm co-occurrence patterns captured by multinomial distributions over visterms. The obvious question is how to learn an aspect representation from an image collection. An unsupervised approach, as used in other mixture models, is particularly attractive. Note however that an unsupervised learning does not guarantee a coherent visual interpretation of the latent aspects. One model to do this, that will be used throughout this dissertation, is introduced in the next section.

2.4 Probabilistic latent semantic analysis

The Probabilistic Latent Semantic Analysis model (PLSA) [29], the initial element of the *aspect model* family, was proposed by Hofmann as a probabilistic alternative to the linear algebra-based Latent Semantic Analysis (LSA) method [16]. It proposes an interesting probabilistic formulation of the concept of topics in text collections, decomposing a document into a mixture of latent aspects defined by a multinomial distribution over the words in the vocabulary. As we have seen in the previous section, this formulation is not restricted to text collections, and will in particular be considered to model visual information. In the following description of the PLSA model, the term *document* therefore defines sets of discrete elements, referring to either text or image documents. Document elements consequently correspond to either words or visterms, respectively.

Document elements are considered as the observation of a discrete random variable X, that can take the values x_j ($j \in \{1, \ldots, N\}$), where N is the number of elements); x_j ranges over the vocabulary words in the text case, over the different visterms in the image case. Documents are represented by a discrete random variable D, that can take the values d_i ($i \in \{1, \ldots, M\}$), where M is the number of documents. Under the PLSA assumptions, the observation of X is conditionally independent of the document under consideration given a hidden variable Z, referred to as a *latent aspect*. This discrete variable is not observed, and can take the possible values z_k ($k \in \{1, ..., L\}$), where L is the number of aspects. The joint probability of observing x_j and d_i is thus given by the marginalization over all the possible values z_k:

$$P(x_j, d_i) = \sum_{k=1}^{L} P(x_j, z_k, d_i). \tag{2.2}$$

The conditional independence assumption between x_j and d_i given z_k translates into the factorization of the joint probability of x_j, z_k and d_i:

$$\begin{aligned} P(x_j, d_i) &= \sum_{k=1}^{L} P(x_j \mid z_k) P(z_k \mid d_i) P(d_i) \\ &= P(d_i) \sum_{k=1}^{L} P(x_j \mid z_k) P(z_k \mid d_i) \end{aligned} \tag{2.3}$$

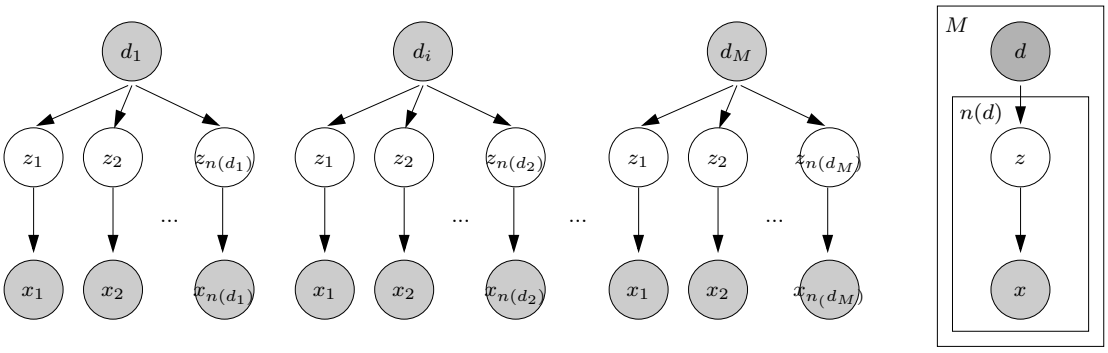


Figure 2.5: Left: unwrapped representation of the PLSA model. Right: compact representation of the same graphical model using the plate notation. Shaded nodes are observed. M is the number of training documents d_i, and $n(d_i) = \sum_j^N n(x_j, d_i)$ is the number of elements in document d_i.

The conditional independence assumption, expressed in Equation 2.3, makes each document d_i a mixture of latent aspects, defined by the multinomial distribution $P(z \mid d_i)$. Each latent aspect z_k is defined by the multinomial distribution $P(x \mid z_k)$, which gives the probability of each element x_j given an aspect z_k. The graphical model shown in Figure 2.5 illustrates the conditional independence assumption of the PLSA model. The unwrapped PLSA model is shown on the left and the compact version is shown on the right, using the plate notation [9]. The variables x and d are observed and represented as shaded nodes; z is not observed, and represented as a white node. The conditional independence assumption between x and d given z is illustrated by the lack of link between the node d and x.

2.4.1 Model parameters

The conditional probability distributions $P(z \mid d)$ and $P(x \mid z)$ are multinomial given that both z and x are discrete random variables. The parameters of these distributions are estimated by the Expectation-Maximization algorithm [29]. For a vocabulary of N different elements, $P(x \mid z)$ is a N-by-L table that stores the parameters of the L multinomial distributions $P(x \mid z_k)$. $P(x \mid z_k)$ characterizes each aspect z_k, and is valid for documents that are not part of the training set. On the contrary, the L-by-M table $P(z \mid d)$ is only relative to the M training documents, as it stores the parameters of the M multinomial distributions $P(z \mid d_i)$ that describes the training document d_i.

To illustrate these conditional probability distributions in the context of image captions, Figure 2.6(c) shows the PLSA decomposition of an image caption in $L = 80$ aspects, where the parameters are learned on the captions of 5188 images (more details will be given in Chapter 5). The PLSA model decomposes the caption into three main aspects, which are represented in Figure 2.6 (d)-(f) by their multinomial distributions over words $P(x \mid z_k)$ and ranked by their probability $P(z \mid d)$. Each distribution $P(x \mid z)$ has also been rearranged showing most probable words first. As can be seen, aspect number 10 (Figure 2.6 (d)) is most likely to generate the word *mountain* (then *valley*); aspect number 3 (Figure 2.6 (e)) generates the words *temple*, *statues*, *sculpture* and *stairs* with high probabilities; aspects 47 (Figure 2.6 (f)) is related to the words *stone*, *ruins*, *sculpture*, *pillars* and *pyramids*. Note that this decomposition of an image caption into a mixture of aspects can now be related to the decomposition of the visual content of an image into a mixture of aspects, that we have introduced before. The approach we have presented here thus offers the possibility to link the two modalities through their aspect decomposition, which we investigate in Chapter 5.

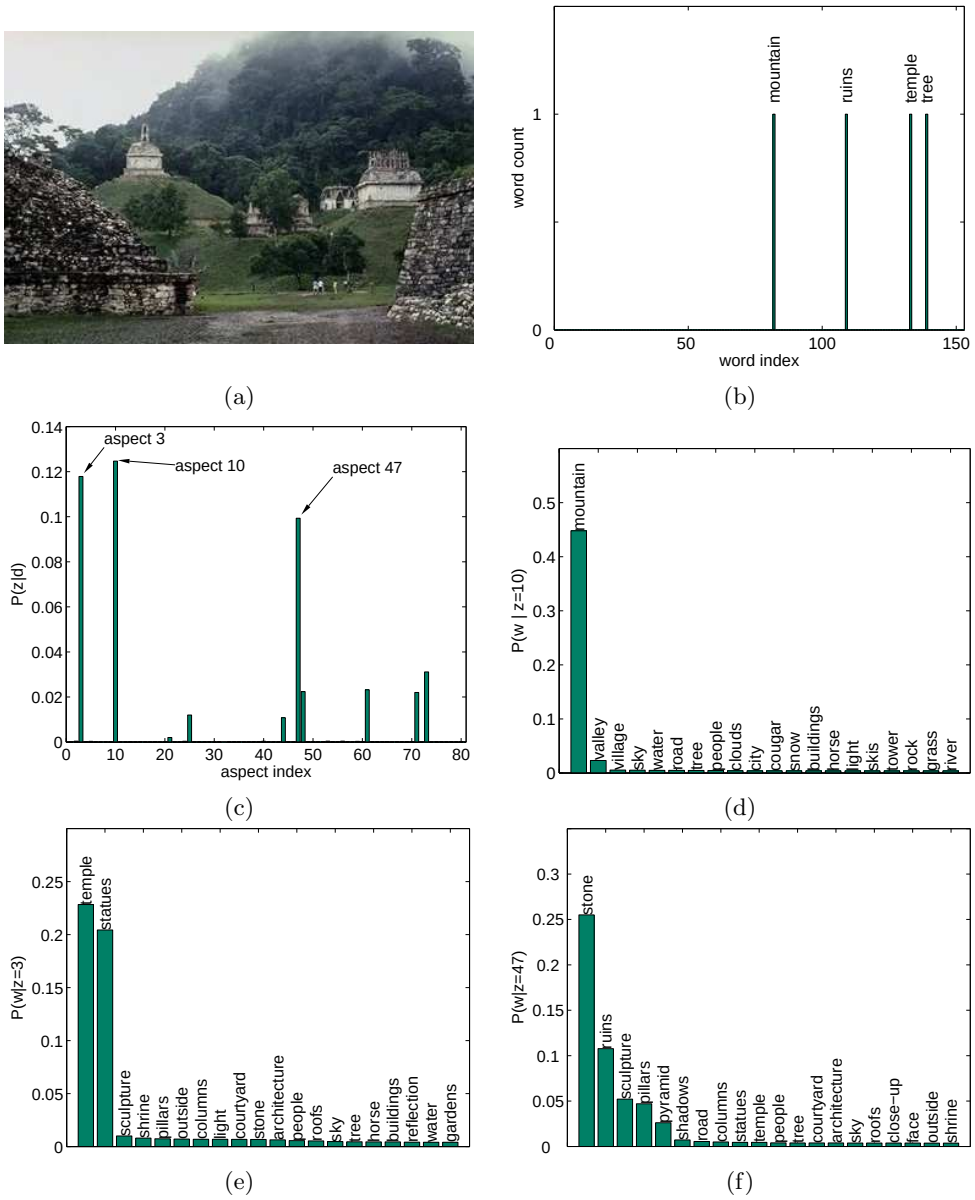


Figure 2.6: Aspect decomposition of the image caption *mountain*, *ruins*, *temple*, and tree for a PLSA model trained with 80 aspects on 5188 image captions. (a) is the considered image d_i, (b) is the word caption histogram $w(d_i)$, (c) is the aspect distribution $P(z \mid d_i)$, and (d-f) are the distributions of the 20 top-ranked words given the three most probable aspects (10, 3 and 47 respectively).

2.4.2 Learning

If each element in each document in a collection was explicitly attributed to its corresponding aspect, the estimation of the per-aspect multinomial distributions over words would only be a matter of counting. Similarly, the distribution of aspects per document could be directly estimated by counting the number of times each aspect is attached to its constituting element. The aspect attribution is however not observed, and an Expectation-Maximization algorithm must therefore be derived from the likelihood of the observed data (Equation 2.4) to estimate the parameters of the distributions $P(x \mid z)$ and $P(z \mid d)$.

$$\mathcal{L} = \prod_i^M \prod_j^N P(d_i) \sum_k^L P(z_k \mid d_i) P(x_j \mid z_k)^{n(x_j, d_i)}, \tag{2.4}$$

where $n(x_j, d_i)$ is the count of element x_j in document d_i. The two steps of the EM algorithm are the following:

E-step : the conditional probability distribution of the latent aspect z_k given the observation pair (d_i, x_j) is computed from the previous estimate of the model parameters.

$$P(z_k \mid d_i, x_j) = \frac{P(x_j \mid z_k) P(z_k \mid d_i)}{\sum_{l=1}^L P(x_j \mid z_l) P(z_l \mid d_i)} \tag{2.5}$$

M-step : The parameters of the multinomial distribution $P(x \mid z)$ and $P(z \mid d)$ are updated with the new expected values $P(z \mid d, x)$.

$$P(x_j \mid z_k) = \frac{\sum_{i=1}^M n(d_i, x_j) P(z_k \mid d_i, x_j)}{\sum_{m=1}^N \sum_{i=1}^M n(d_i, x_m) P(z_k \mid d_i, x_m)} \tag{2.6}$$

$$P(z_k \mid d_i) = \frac{\sum_{j=1}^N n(d_i, x_j) P(z_k \mid d_i, x_j)}{n(d_i)} \tag{2.7}$$

2.4.3 Inference of the aspect mixture weights of a new document

The conditional probability distribution over aspects $P(z \mid d_{new})$ can be inferred for an unseen document d_{new} using the Algorithm 2.1. This *folding-in* method, proposed in [29], maximizes the likelihood of the document d_{new} with a partial version of the EM algorithm described above, where $P(x \mid z)$ is *kept fixed* (i.e., not updated at each M-step). In doing so, $P(z \mid d_{new})$ maximizes the likelihood of the document d_{new} with respect to the previously learned $P(x \mid z)$ probability table.

2.4.4 Overfitting control

We control the overfitting of the model by early stopping, based on the likelihood of a validation set $\mathcal{D}_{valid}$. We consider the *folding-in likelihood*, that allows good performance prediction and overfitting control without the need for a tempered version of the EM algorithm [85]. The probability of aspects given each validation document $P(z \mid d'_i)$ is first estimated using the folding-in method, as described in Section 2.4.3. The folding-in likelihood of the validation set $\mathcal{D}_{valid}$ given the current parameters $P(x \mid z)$ is then defined as:

$$\mathcal{L}(\mathcal{D}_{valid}) = \prod_i^{M_{valid}} \prod_j^N P(x_j \mid d'_i)^{n(x_j, d'_i)} \tag{2.8}$$

$$= \prod_i^{M_{valid}} \prod_j^N \sum_k^L (P(x_j \mid z_k) P(z_k \mid d'_i))^{n(x_j, d'_i)}, \tag{2.9}$$

Algorithm 2.1 Estimation of the $P(z \mid d_{new})$ distribution using the $P(x \mid z)$ probability table

random initialization of the $P(z \mid d_{new})$ distribution

for iter $<$ max_iter **do**
`[E-step]`
for all (d_{new}, x_j) pairs such that $n(d_{new}, x_j) > 0$, and $k \in \{1, \ldots, L\}$ **do**

$$P(z_k \mid d_{new}, x_j) = \frac{P(x_j \mid z_k)P(z_k \mid d_{new})}{\sum_{l=1}^{L} P(x_j \mid z_l)P(z_l \mid d_{new}))}$$

end for

`[Partial M-step]`
for $k \in \{1, \ldots, L\}$ **do**

$$P(z_k \mid d_{new}) = \frac{\sum_{j=1}^{N} n(d_{new}, x_j)P(z_k \mid d_{new}, x_j)}{n(d_{new})}$$

end for
end for

where $P(z_k \mid d'_i)$ is estimated by folding-in (see Algorithm 2.1), and M_{valid} is the number of validation documents d'. The complete learning algorithm of a PLSA model, including the overfitting control based on the folding-in likelihood computation of an evaluation set $\mathcal{D}_{valid}$, is detailed in Algorithm 2.2. At each EM iteration, the folding-in likelihood of the validation set $\mathcal{D}_{valid}$ is computed, and the model parameters corresponding to the highest folding-in likelihood value are kept.

Algorithm 2.2 Learning a PLSA model with overfitting control, using a validation set $\mathcal{D}_{valid}$

random initialization of the $P(z \mid d)$ and $P(x \mid z)$ probability tables

while increase in the likelihood of validation data $\mathcal{L}(\mathcal{D}_{valid}) > T$ **do**

`[E-step]`

for all (d_i, x_j) pairs such that $n(d_i, x_j) > 0$, and $k \in \{1, \ldots, L\}$ **do**

$$P(z_k \mid d_{new}, x_j) \quad = \quad \frac{P(x_j \mid z_k)P(z_k \mid d_i)}{\sum_{l=1}^{L} P(x_j \mid z_l)P(z_l \mid d_i))}$$

end for

`[M-step]`

for $j \in \{1, \ldots, N\}$ and $k \in \{1, \ldots, L\}$ **do**

$$P(x_j \mid z_k) \quad = \quad \frac{\sum_{i=1}^{M} n(d_i, x_j)P(z_k \mid d_i, x_j)}{\sum_{m=1}^{N}\sum_{i=1}^{M} n(d_i, x_m)P(z_k \mid d_i, x_m)}$$

end for

for $k \in \{1, \ldots, L\}$ and $i \in \{1, \ldots, M\}$ **do**

$$P(z_k \mid d_i) \quad = \quad \frac{\sum_{j=1}^{N} n(d_i, x_j)P(z_k \mid d_i, x_j)}{n(d_i)}$$

end for

estimate $P(z \mid d')$, where $d' \in \mathcal{D}_{valid}$ by folding-in

compute folding-in likelihood of the validation data $\mathcal{L}(\mathcal{D}_{valid})$

end while

2.5 Conclusion

Modeling an image as a mixture of hidden aspects, defined by multinomial distributions over visterms, represents a promising approach to model visual information. In this chapter, we introduced the concept of latent aspect and illustrated what visual latent aspects are, in perspective with the concept of latent aspects in a text collection. But can latent aspect models really capture coherent co-occurrence information from a bag-of-visterms representation? Given that aspects are obtained by unsupervised learning, will they match semantic concepts? The concept of mixture of aspect for images is systematically investigated in the following chapters to answer these questions.

The decomposition of an image into a mixture of aspects $P(z \mid d)$ is first used as a novel image representation estimated from the bag-of-visterms, which is evaluated for various image classification tasks in Chapter 3. In the same chapter, an image ranking method is proposed to illustrate what latent aspects are, relying on the probability of documents given an aspect $P(d \mid z)$. The visterms v themselves can be classified into different classes, producing a novel way of performing image segmentation resulting from the class label density. We show in Chapter 4 that the conditional probability of aspects given a document and a visterm, estimated in the E-step (Equation 2.5) $P(z \mid d, v)$ can serve as a basic visterm classifier. We then propose and evaluate two principled methods to learn a visterm classifier based on the image decomposition into aspects. Finally, we propose three alternative learning procedures to merge the textual and visual aspects of annotated image in Chapter 5, for which the aspect mixture weights are estimated from both the textual and visual modalities, the visual modality only or the textual modality only. This allows to predict a word distribution $P(w \mid z)$ based on the bag-of-visterms representation of a new image that can serve as textual indexing.

Chapter 3

Aspect-based image classification and ranking

In this Chapter, we evaluate the relevance of an aspect-based image representation for image classification, and propose a ranking method derived from the aspect mixture weights. The intuition behind the concept of mixture of aspects for images was given in Chapter 2, and a concrete application of the aspect model for images is proposed here in the context of scene and object classification. Three of our publications compose the base of this analysis: the first two [68, 67] investigated the classification of images in different scene types, the third [54] addressed the problem of classifying images based on what object they contain. Both introduced the concept of aspect-based image ranking.

We discuss the problem of scene and object classification, pointing out how the two problems are related in Section 3.1. An overview of the related work, discussing recently proposed approaches for scene and object classification is given in Section 3.2. In Section 3.3, we present the construction of the bag-of-visterms representation - combining a point detector and an invariant region descriptor - that we considered to learn the aspect-based image representation from. A parallel between words and visterms is drawn, confirming the polysemy and synonymy ambiguities that result from the quantization of image regions into visterms. Section 3.4 discusses the classifier that is considered, the different scene and object classification datasets used for the evaluation, and the baseline methods for the scene classification problems. The experimental investigation, contained in Section 3.5, starts with the evaluation of the bag-of-visterms representation for scene and object classification, comparing its performance to existing baseline methods and analyzing the influence of the visterm vocabulary size. The aspect-based image representation is then compared to the bag-of-visterms representation for scene and object classification, showing how the aspect-based image representation can take advantage of unlabeled data. Finally, we propose a method for unsupervised soft clustering of images in Section 3.6, showing the type of structure captured by an aspect model in an image collection.

3.1 Scene and object classification

Scene and object classification are two tightly related problems, which can be difficult to specify independently. A scene is composed of several entities (e.g. car, house, building, face, wall, door, tree, forest, rocks), organized in often unpredictable layouts. If a majority of these entities are correctly recognized, the interpretation of the corresponding scene might become straightforward. Conversely, if the scene in which an object occurs is correctly identified, the recognition of this object might be easier given this scene context. Scene and object classification are therefore two interdependent problems, as illustrated with examples taken from our scene dataset (left column) and the objects dataset (right column) in Figure 3.1. Image (a) is from the *city* scene category, and image (b) is from the *building* object category. The correct identification of a *city* scene context could certainly

(a) city (b) building

(c) indoor (d) book

(e) landscape (f) tree

Figure 3.1: Illustration of the scene and the object classification datasets. From the scene dataset, examples from the categories city (a), indoor (c), and landscape (e) are shown. From the object dataset, examples from the categories buildings (b), book (d), and tree (f) are shown.

help the classification of an image as containing a *building* object. Conversely, an image classified as containing a *building* is more likely to belong to the *city* class.

In the extreme case in which each object category is highly correlated with a single type of background, specific to each object category, the object classification task can be equivalently formulated as a scene classification task. However, if the background regions surrounding an object are only marginally correlated with their categories, the two problems differ. The *indoor* scene category, illustrated by the image (c) in Figure 3.1, does not necessarily imply the presence of a *book* (d) for instance. The *tree* object category, exemplified by image (f), is not necessarily related to the *forest* scene category, illustrated by image (e) if it appears in isolation. Furthermore, in that example, the background of the *tree* object example is actually closer to the *city* category, showing that background information is not sufficient for object classification. Object and scene classification are therefore distinct - though related - problems.

On one hand, images of a given object are usually characterized by the presence of a limited set of specific local image patches, organized into different view-dependent geometrical configurations. On the other hand, the visual content (entities, layout) of a specific scene class exhibits a large variability, characterized by the presence of a large number of different visual descriptors. In view of this, while the specificity of an object strongly relies on the geometrical configuration of a relatively limited number of visual descriptors [76, 25], the specificity of a scene class greatly rests on the particular patterns of co-occurrence of a large number of visual descriptors. Their constitutive components are nevertheless the same: local image patches that either characterize a scene class when distributed over an image, or define an object category when organized in a specific configuration. We propose to use the same representation for the two types of visual content, relying on visterms to capture the relevant constitutive elements.

3.2 Related Work

The problem of scene classification using low-level features has been studied in image and video retrieval for several years [28, 83, 88, 87, 61, 59, 64, 79]. Broadly speaking, the existing methods differ by the definition of the target scene classes, the specific image representations, and the classification method. We focus the discussion on the first two points. With respect to scene definition, most methods have aimed at classifying images into a small number of semantic scene classes, including *indoor/outdoor* [83, 78], *city/landscape* [88], and sets of natural scenes (e.g. *sunset/forest/mountain*) [59]. However, as the number of categories increases, the issue of overlapping between scene classes in images arises. To handle this issue, a continuous organization of scene classes (e.g. from *man-made* to *natural* scenes) has been proposed [61]. Alternatively, the issue of scene class overlap can be addressed by doing scene annotation (e.g. labeling a scene as depicting multiple classes). This approach is followed by Boutell et al. [8], which exploits the output of one-against-all classifiers to derive multiple class labels.

Regarding global image representations for scene classification, the work by Vailaya et al. is regarded as the representative of the literature in the field [88, 87]. This approach relies on a combination of distinct low-level cues for different two-class problems (global edge features for *city/landscape*, and local color features for *indoor/outdoor*). In the work by Oliva and Torralba [61], an intermediate classification step into a set of global image properties (*naturalness*, *openness*, *roughness*, *expansion*, and *ruggedness*) is proposed. Images are manually labeled with these properties, and a Discriminant Spectral Template (DST) is estimated for each property. The DSTs are based on the Discrete Fourier Transform (DFT) extracted from the whole image, or from a four-by-four grid to encode basic spatial information. A new image is represented by the degree of each of the five properties based on the corresponding estimated DST, and this representation is used for the classification into semantic scene categories (*coast*, *country*, *forest*, *mountain*,...). Other approaches to scene classification also rely on an intermediate supervised region classification step [59, 72, 20, 91]. Based on a Bayesian Network formulation, Naphade and Huang defined a number of intermediate regional concepts (e.g. *sky*, *water*,

rocks) in addition to the scene classes [59]. The relations between the regional and the global concepts are specified in the network structure. Serrano et al. [72] propose a two-stage classification of *indoor/outdoor* scenes, where features of individual image blocks from a spatial grid layout are first classified into *indoor* or *outdoor*. These local classification outputs are further combined to create the global scene representation used for the final image classification. Similarly, Vogel and Schiele recently used a spatial grid layout in a two-stage framework to perform scene retrieval [91] and scene classification [92]. The first stage does classification of image blocks into a set of regional classes, different from the scene target labels which extends the set of classes defined in [59] (this is thus different from [72] and requires additional block ground-truth labeling). The second stage performs retrieval or classification based on the occurrence of such regional concepts in query images. Alternatively, Lim and Jin [42] successfully used the soft output of semi-supervised regional concept detectors in an image indexing and retrieval application. In a different formulation, Kumar and Herbert used a conditional random field model to detect and localize man-made scene structures, doing in this way scene segmentation and classification [33]. Overall, a large number of local, regional, and global image representations have been used for scene classification.

The combination of interest point detectors and local descriptors are increasingly popular for object detection, recognition, and classification [43]. The literature in the field is too large to discuss in exhaustively here [76, 25, 21, 17, 62, 77, 94, 37]. For the classification task, recent works include [25, 21, 17, 62, 22, 94]. Most existing works have targeted a relatively small number of object classes (an exception is [22]). Fergus et al. optimized, in a joint unsupervised model, a scale-invariant localized appearance model and a spatial distribution model [25]. Fei-Fei et al. proposed a method to learn object classes from a small number of training examples [21]. The same authors extended their work to an incremental learning procedure, and tested it on a large number of object categories [22]. Dorko and Schmid performed feature selection to identify local descriptors relevant to a particular object class, given weakly labeled training images [17]. Opelt et al. proposed to learn classifiers from a set of visual features, including local invariant ones, via boosting [62]. Although our work shares the use of invariant local descriptors with all these methods, scenes are different than objects in a number of ways, as discussed in the Introduction, and pose specific challenges.

The analogy between invariant local descriptors and words has also been exploited recently [76, 77, 94]. Sivic and Zisserman proposed to cluster and quantize local invariant features into visterms, for object matching in frames of a movie. Such approach allows to reduce noise sensitivity in matching and to search efficiently through a given video for frames containing the *same* visual content (e.g. an object) using inverted files [76, 77]. Csurka et al. extended the use of visterms creating a system for object matching and classification based on a bag-of-words representation built from local invariant features and various classifiers, reporting good results [94]. However, these methods neither investigated the task of scene modeling and classification, nor considered latent aspect models.

In another research direction, a number of works have also relied on the definition of visterms and/or on variations of latent space models to model annotated images, i.e. to link images with (semantic) words [57, 5, 6, 31, 51, 52, 95]. However, all these methods have relied on traditional regional image features without much viewpoint and/or illumination invariance. As an example, R. Zhang and Z. Zhang [95] explored the use of a latent space model to discover semantic concepts for content-based image retrieval. The model is learned from a set of quantized regions per image, and the similarity between images is computed from the estimated posterior probability over aspects. In our work, we characterize an image using local descriptors as visterms, taking into account the problems that exist in the construction of a visterm vocabulary. We use latent space models not to annotate images but to address some limitations of the visterm vocabulary, describing images with a model that explicitly accounts for the importance of visterm co-occurrence. The problem of image annotation is addressed in Chapter 5.

In parallel to our work [68], the joint use of local invariant descriptors and probabilistic latent aspect models has been investigated by Sivic et al. for object clustering in image collections [75], and by Fei-Fei and Perona for scene classification [23]. Although related, these two approaches differ from ours in their assumptions, and do not take advantage of unlabeled data in their experiments.

Sivic et al. [75] investigated the use of both Latent Dirichlet Allocation (LDA) [7] and PLSA for clustering objects in image collections. With the same image representation as ours, they showed that latent aspects closely correlate with object categories from the Caltech object database, though these aspects are learned in an unsupervised manner. The number of aspects was chosen by hand to be equal (or very close) to the number of object categories, so that images are seen as mixtures of one 'background' aspect with one 'object' aspect. This allows for a direct match between object categories and aspects, but at the same time implies a strong coherence of the appearance of objects from the same category: each category is defined by only one multinomial distribution over the quantized local descriptors. Closer to our work, Fei-Fei and Perona [23] proposed two variations of LDA [7] to model scene categories. They tested different region detection processes - ranging from random sampling to fixed-grid segmentation - to build an image representation based on quantized local descriptors. Contrarily to [75], Fei-Fei and Perona [23] propose to model a scene category as a mixtures of aspects, and each aspect is defined by a multinomial distribution over the quantized local descriptors. This is achieved by the introduction of an observed class node in their models [23], which explicitly requires each image example to be labeled during the learning process.

In our approach, we model scene and object images using a probabilistic latent aspect model and quantized local descriptors, but without assuming a one to one correspondence between categories and aspects as in [75], and without learning a single distribution over aspects per scene category as in [23]. Images - not categories - are modeled as mixtures of aspects in a fully unsupervised way, without class information. The distribution over aspects serves as image representation, that is inferred on new images and used for supervised classification in a second step. These differences are crucial, as they allow us to investigate the use of unlabeled data for learning the aspect-based image representation, which is one contribution of this chapter. We also evaluate the performance of the bag-of-visterms representation, learned from different data sources, with other types of image representations.

3.3 Image representation

In this section, we focus on the two image representations that we want to evaluate for image classification: the first one is the bag-of-visterms, built from automatically extracted and quantized local descriptors, the second one is based on aspect mixture weights, as defined in Chapter 2.

3.3.1 Bag-of-visterms from interest points

The construction of the bag-of-visterms (BOV) feature vector $s$from an image d can be summarized in the four steps illustrated in Figure 3.2 (a-d): (b) interest points are automatically detected in the image, (c) local descriptors are computed over those regions, and (d) all the descriptors are quantized into visterms, and counted to build the BOV representation of the image.

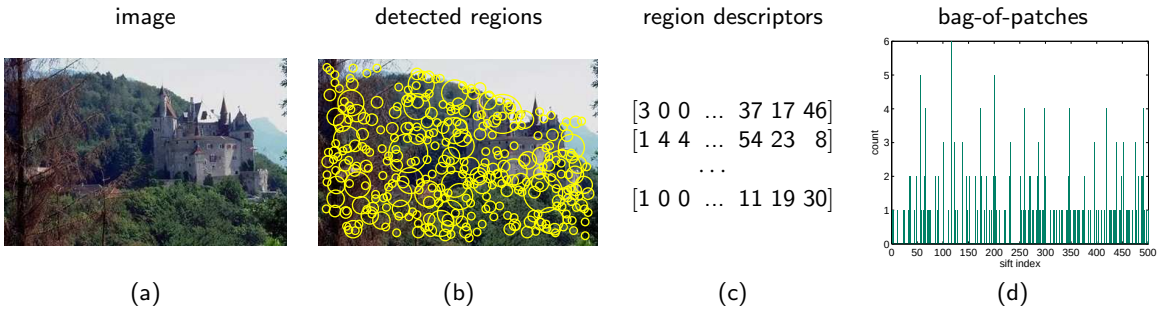


Figure 3.2: Construction of the bag-of-patches representation of an image. Interest points are detected at different locations and scales (b), regions properties are captured by local descriptors (c), and these descriptors are quantized into patches, creating a bag-of-patches representation (d).

A variety of point detectors and descriptors have been proposed in the literature, and thorough evaluations of their combinations have been conducted for *wide baseline matching* tasks [46, 48, 47]. Their performance is therefore well established in this context, but the extrapolation of these results to our case scenario is not obvious. We want to exploit the combination of point detectors and descriptors to build an image representation that can discriminate between scene and object classes. This non-standard application does not necessarily require the same invariance properties that are crucial for matching the same points of a given object between two transformed images. The evaluation of different detector and descriptor combinations should therefore ideally be reconducted to find the best performing solution for each classification task, that could require a particular combination depending of the specific classes that are considered. We do not discuss these experiments here. The results and discussions can be found in [67], and they justify the choice made in this chapter.

In the following, we provide a description of the interest point detector and the local descriptor used in this work, justifying our choice based on other studies.

Interest point detector

The goal of the interest point detector is to automatically extract characteristic points -and more generally regions- from the image, which are invariant to some geometric and photometric transformations. This invariance property is interesting, as it ensures that given an image and its transformed version, the same image points will be extracted from both and hence, the same image representation will be obtained.

Several interest point detectors exist in the literature. They vary mostly by the amount of invariance they theoretically ensure, the image property they exploit to achieve invariance, and the type of image structures they are designed to detect [46, 86, 43, 48]. In this work, we use the difference of Gaussians (DoG) point detector [43]. The DoG point detector is based on the difference of Gaussian filters at various scales s. Its implementation can be summarized in the following four points:

- Convolution of the image with Gaussian filters at different scales (see Figure 3.3 (a) and (b)).
- Construction of the difference of Gaussian images (see Figure 3.3 (c)) from adjacent blurred images.
- Scale-space extrema detection, illustrated in Figure 3.4: each pixel of a DoG image (in black) is compared to its 8 neighboring pixels fomr the same image (in blue) and the 2×9 neighboring pixels in the two adjacent DoG images (in green). This pixel is a valid candidate keypoint if it is a minimum or a maximum.
- Post-processing :
  - keypoints with low contrast are removed,
  - responses along edges are eliminated,
  - the keypoint is assigned an orientation and a scale.

This detector essentially identifies blob-like regions where a maximum or minimum of intensity occurs in the image, and is invariant to translation, scale, rotation and constant illumination variations. We chose this detector since it was shown to perform well in comparisons previously published by other authors [47]. An additional reason to prefer this detector over fully affine-invariant ones [46, 86, 45] is also motivated by the fact that an increase of the degree of invariance may remove information about the local image content that is valuable for classification.

Local descriptor

Local descriptors are computed on the region around each interest point identified by the detector, relatively to the scale at which the point was detected. We use the SIFT (for Scale Invariant Feature

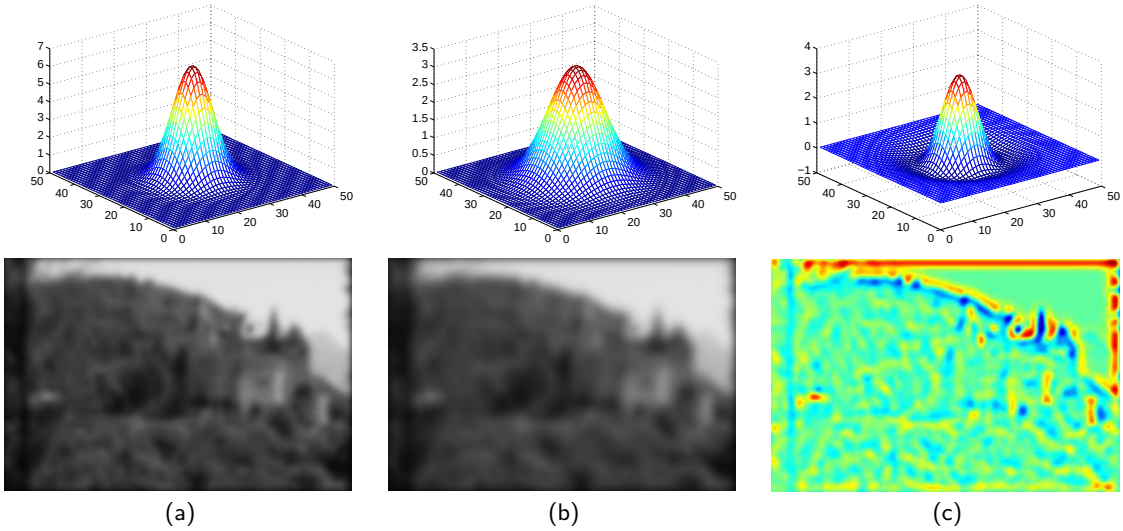


Figure 3.3: The difference of Gaussian (DoG) filtering of an image. A first Gaussian filter with a standard deviation σ_1 is applied to the image, resulting in a first Gaussian blurred version (a). A second Gaussian filter with a standard deviation σ_2 is applied to the image, resulting in a second Gaussian blurred version (b). The difference of the two Gaussian blurred images (a) and (b) results in the DoG image (c), which is equivalent to applying the difference of Gaussian filter.

Transform) feature as local descriptors [43]. This choice is motivated by the findings of several publications [47, 23]. The SIFT descriptor is computed from the grayscale information of images, and was shown to perform best in terms of specificity of region representation and robustness to image transformations [47]. Given the orientation of the image region estimated from the interest point detector, the SIFT features are computed as local histograms of edge directions computed over different parts of the interest region (see Figure 3.5 (right)). This allows to capture the structure of the local image regions, which correspond to specific geometric configurations of edges or to more texture-like content. In [43], it was shown that the use of 8 orientation directions and a grid of 4×4 parts gives a good compromise between descriptor size and accuracy of representation. We use the same settings, that correspond to a descriptor of size 128.

Quantization of local descriptors into visterms

From the two preceding point detection and description steps, we obtain a set of real-valued local descriptors. In order to obtain a simple, fixed size image representation, we quantize each local descriptor s into a discrete set of visterms v according to a nearest neighbor rule:

$$s \longmapsto Q(s) = v_i \Longleftrightarrow \text{dist}(s, v_i) \leq \text{dist}(s, v_j) \qquad \forall j \in \{1, \ldots, N\} \tag{3.1}$$

where N denotes the size of the visterm set. We will call *vocabulary* the set of all visterms.

The construction of the vocabulary is performed through clustering. We apply the K-means algorithm to a set of local descriptors extracted from training images, each cluster corresponding to a visterm. We used the Euclidean distance in the clustering (and in Equation 3.1) and choose the number of clusters depending on the desired vocabulary size. The choice of the Euclidean distance to compare SIFT features is common [43, 46].

Technically, the grouping of similar local descriptors into a specific visterm can be thought of as being similar to the *stemming* preprocessing step of text documents, as already mentioned in Chapter 2. The intuition behind stemming is that the meaning of words is carried by their stem

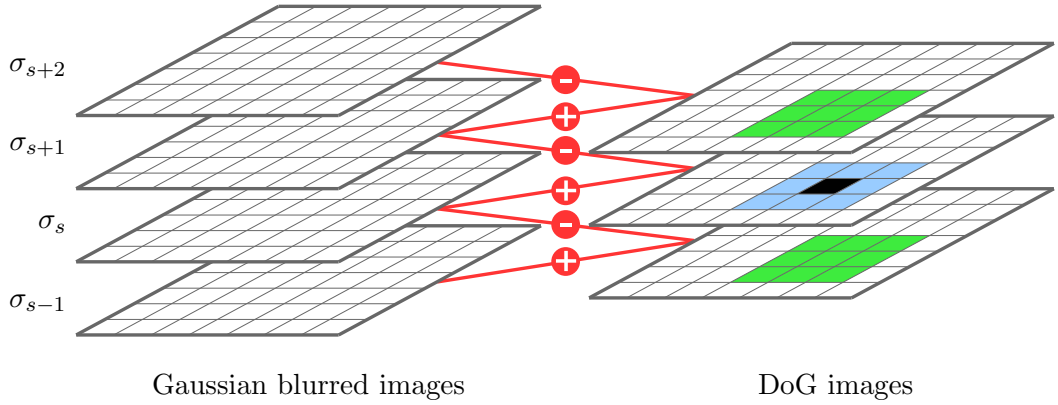


Figure 3.4: Construction of the difference of Gaussian images from Gsuassian blurred images(left), and scale-space extrema detection (right). DoG images are constructed by substracting Gaussian blurred images at different σ_s. Given three adjacent DoG images, each pixel value (in black) is compared to its 8 neighboring pixels from the same image (in blue) and its 2×9 neighboring pixels in the two adjacent DoG images.

rather than by their morphological variations [4]. The terms *test*, *tests* and *testing* are for instance mapped to the same stem *test*, and become equivalent in the bag-of-words representation of a text. The same motivation applies to the quantization of similar descriptors, that are mapped onto a single visterm. Furthermore, local descriptors will be considered as distinct whenever they are mapped to different visterms, regardless of whether they are close or not in the SIFT feature space. This also resembles the text modeling approach which considers that all information is in the stems, and that any distance defined over their representation (e.g. strings in the case of text) carries no semantic meaning.

To illustrate what visterms correspond to in practice, we show image patches sampled from three visterms obtained when building the vocabulary V_{1000} (see Section 3.4.2 for details about the data) in Figure 3.6. The quantization of local descriptors achieves the intended goal: image regions that contain a similar type of visual content, in term of texture and edge directions in this case, are grouped into the same set of image patches. The first visterm in Figure 3.6 (a) corresponds to image patches containing vertical, corner-like sharp structures that can be found on windows. The visterm (b) contains image patches that correspond to high frequency textures, very likely to occur in natural scenes (tree, rocks, ...). The visterm (c) corresponds to heterogeneous image patches, having a centered, darker spot in common. The examples from visterm (d) are composed of both window and eyes patches, sharing a similar look. As a first observation, it seems that the quantized local patches are good candidates to represent object and scenes as sets of parts, as suggested in Section 3.1. A large number of visterms should for instance allow to represent a variety of scene types. Visterms such as the one illustrated on Figure 3.6 (a) represent local structures encountered in cities, while visterms such as the one illustrated on Figure 3.6 (b) are likely to correspond to vegetation regions.

The visterms (c) and (d) in Figure 3.6 illustrates the ambiguity issue that arises from the quantization of descriptors. The patches sampled from visterm (c) contain eyes, window parts, and a variety of other content. This could be described as *polysemy*, as a single visterm represents different visual content. The visterm (d) mostly contain eye regions, captured with a different orientation than visterm (c), which illustrates the potential *synonymy* of visterms: two visterms characterize the same image content. As we have discussed in Chapter 2, these ambiguities can be addressed by a latent aspect model. Latent aspects, defined by multinomial distribution over visterms learned from a relevant dataset, can potentially disambiguate the visterms from their co-occurrence context in the image.

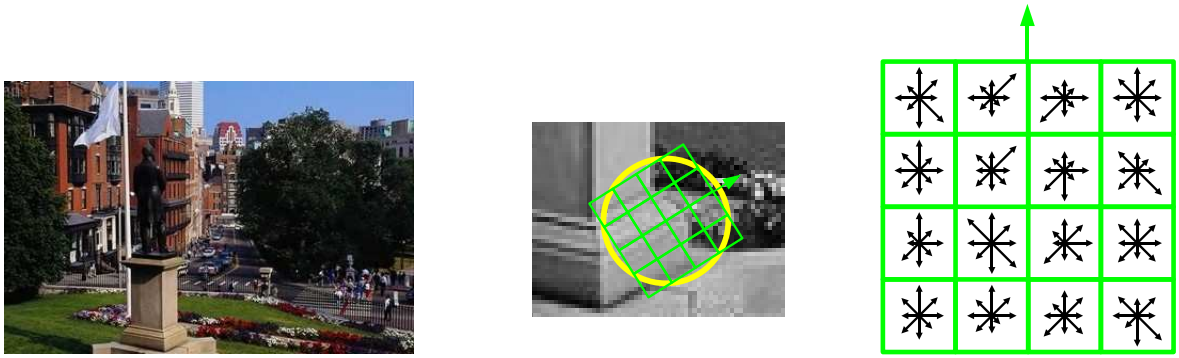


Figure 3.5: The Scale Invariant Feature Transform (SIFT) descriptor. The detected regions are segmented into a 4×4 grid, and each square is represented by an eight-bin histogram of the edge directions in this region, resulting in a description vector of dimension 128.

Bag-of-visterms

The first image representation that we will evaluate for classification is the bag-of-visterms (BOV), constructed from the local descriptors according to:

$$s(d_i) = \{n(d_i, v_1), \ldots, n(d_i, v_j), \ldots, n(d_i, v_N)\}, \tag{3.2}$$

where $\mathrm{n}(d, v_i)$ denotes the number of occurrences of visterm v_j in image d. This vector-space representation of an image contains no information about spatial relationship between visterms. The standard bag-of-words text representation results in a very similar 'simplification' of the data: even though word ordering contains a significant amount of information about the original data, it is completely removed from the final document representation.

3.3.2 Aspect-based image representation

We propose to evaluate an aspect-based image representation, estimated from the BOV representation using a PLSA model. The PLSA model parameters are estimated on a first set of images, and the mixture aspect weights are inferred on the documents to classify using the folding-in method described in Section 2.4.3. As an illustration, the Figure 3.7 shows the distribution over aspects for two images, for an aspect model trained on a collection of 6600 images of landscape and city images for a vocabulary of 1000 visterms. The conditional distributions of visterms given the $K = 60$ aspects are represented on the right column of Figure 3.7, representing an aspect by its specific visterm co-occurrence pattern. We see in Figure 3.7 that the BOV representations of the two images are modeled by two dissimilar distributions over aspects, reflecting their differences in content. The two images are composed of different visterm co-occurrences that exist in the image collection, resulting in different image-dependent contexts.

The aspect mixture parameters $P(z|d_i)$ given an image d_i is proposed as an image representation that for scene and object classification:

$$a(d_i) = \{P(z_1|d_i), \ldots, P(z_k|d_i), \ldots, P(z_L|d_i)\} \tag{3.3}$$

3.4 Experimental setup

In this section, we describe the scene and object classification tasks that we consider, the origin and the composition of the corresponding datasets, and the baseline methods that were implemented for comparison purposes. A description of the specific classifier used in experiments is given here.

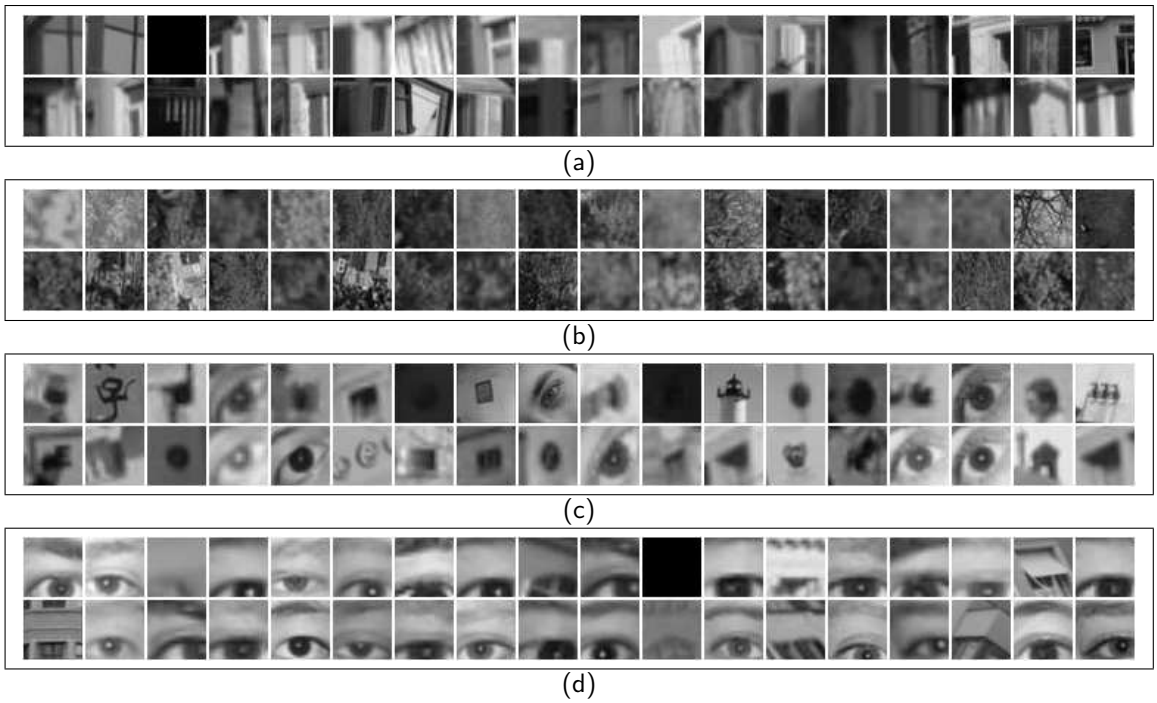


Figure 3.6: Illustration of 4 visterms based on 36 randomly sampled regions attributed to these visterms. We see that similar image patches are mapped onto the same visterm.

3.4.1 SVM classification

To classify an input image d represented either by the BOV vectors s, the aspect parameters a, or any of the feature vector of the baseline approach (see next section), we employed Support Vector Machines (SVMs) [12]. SVMs have proven to be successful in solving machine learning problems in computer vision and text categorization tasks, especially those involving large dimensional input spaces. In the current work, we use Gaussian kernel, whose bandwidth was chosen based on a 5-fold cross-validation procedure.

Standard SVMs are binary classifiers, which learn a decision function $f(x)$ through *margin* optimization [12], such that $f(x)$ is large (and positive) when the input x belongs to the target class, and negative otherwise. For multi-class classification, we adopt a one-against-all approach [93]. Given a n-class problem, we train n SVMs, where each SVM learns to differentiate images of one class from images of all other classes. In the testing phase, each test image is assigned to the class of the SVM that delivers the highest output of its decision function.

3.4.2 Classification tasks

We investigate the relevance of the aspect-based image representation for scene and object classification, comparing its performance with the BOV representation. In the case of scene classification, we measure the effect of changing the vocabulary size, varying the number of aspects. Moreover, results obtained with baseline approaches based on global descriptors are reported for comparison.

Scene classification

Four scene classification tasks, ranging from binary to five-class classification, have been considered to evaluate the performance of the proposed approaches. We first considered two standard, unambiguous

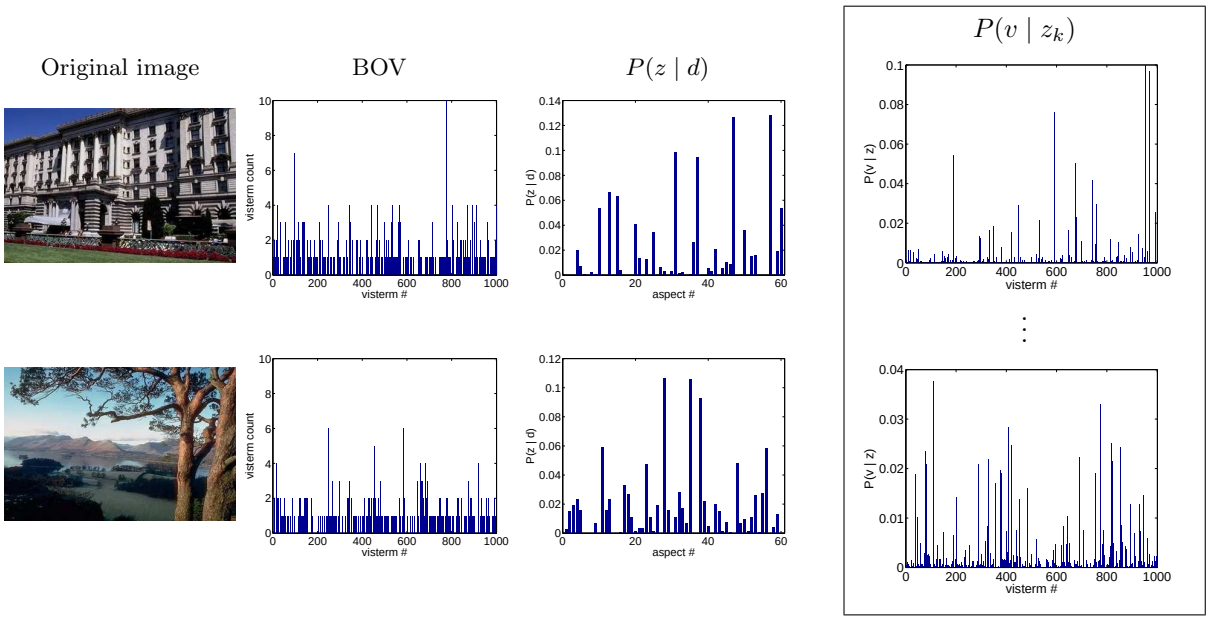


Figure 3.7: Two scene images and their decomposition into a mixture of $K = 60$ aspects, estimated by the PLSA model. The second column is the histogram of 1000 visterms (BOV) corresponding to the image on the same row, the third column shows the estimated distribution over aspects given this BOV representation. The right column represents the K conditional distributions over visterms given the aspects z_k.

binary classification tasks: *indoor* vs. *outdoor*, and *landscape* vs. *city*. These two binary classification tasks allow a first evaluation of the classification performance, and a fair comparison with approaches that have been proposed for the same tasks [87]. For a more detailed analysis of the performance, we then merged the two binary classification tasks to obtain a three-class problem (*indoor* vs. *city* vs. *landscape*). We also subdivided the *landscape* class into *mountain* and *forest*, and the *city* class into *street view* and *panoramic view* to obtain a five-class dataset. As discussed in Section 3.5, the performance can vary depending on the classification tasks that is considered. In total, five datasets were created for our scene classification experiments.

D1: this dataset of 6680 images contains a subset of the Corel database [87], and is composed of 2505 city and 4175 landscape images of 384×256 pixels.

D2: this set is composed of 2777 indoor images retrieved from the Internet. The size of these images is on average 384×256 pixels. Original images with larger dimensions were resized using bilinear interpolation. The image size in the dataset was kept approximately constant to avoid a potential bias in the BOV representation, since it is known that the number of detected interest points is highly dependent on the image resolution.

D3: this dataset is constituted by 3805 images from several sources: 1002 building images (ZuBud) [73], 144 images of people and outdoors [62], 435 indoor human faces [94], 490 indoor images (Corel) [87], 1516 city/landscape overlap images (Corel) [87], and 267 Internet photographic images.

D4: this dataset is composed of all images from the datasets **D1** and **D2**. The total number of images in this dataset is 9457.

D4v: this is a subset of **D4** composed of 3805 randomly chosen images.

D5: this is a five-class dataset. It comprises all images from the dataset **D2**, and images from **D1** whose content corresponds to the selected classes. From the 6680 images of **D1** we kept : 590 mountain images, 492 forest images, 1957 city street images (close-up of buildings), and 548 city panoramic images (middle to far views from buildings). The datasets contains a total of 6364 images.

We use the dataset **D1** for the city vs. landscape scene classification task, and the dataset **D4** for indoor vs. outdoor scene classification. We also use **D4** in the three-class case. Dataset **D5** is employed in the five-class problem. Alternative vocabularies were constructed from either **D3** or **D4v**, allowing us to study the data's influence on the vocabulary model, and its impact on classification performance. With 3805 images, we obtained in both cases approximately one million descriptors to train the vocabulary models.

Object classification

In addition to the scene classification experiments, one object classification task is considered.

D6: The image classes are: faces (792), buildings (150), trees (150), cars (201), phones (216), bikes (125) and books (142), adding up to a total of 1776 images. The size of the images varies considerably: images can have between 10k and 1,2M pixels while most image sizes are around 100-150k pixels. We resize all images to 100k pixels since the local invariant feature extraction process is highly dependent of the image size. This ensures that no class-dependent image size information is included in the representation, but does introduce some resizing artifacts at the same time.

3.4.3 Experimental protocol

The protocol for each of the classification experiments is as follows. The full dataset of a given experiment was divided into 10 parts, defining 10 different splits of the full dataset. One split corresponds to keeping one part of the data for testing, while using the other nine parts for training (hence the amount of training data is 90% of the full dataset). In this way, we obtain 10 different classification results. Reported values for all experiments correspond to the average error over all splits, and standard deviations of the errors are provided in parentheses after the mean value.

Additional experiments were conducted with a decreasing amount of training data for the SVM model, to test the robustness of the image representation. In that case, for each of the splits, images were chosen randomly from the training part of the split to create a reduced training set. Care was taken to keep the same class proportions in the reduced set as in the original set, and to use the same reduced training set in those experiments involving two different representation models. The test data of each split was left unchanged.

3.4.4 Baseline method for scene classification

As a baseline method, we use the image representations proposed by Vailaya et al. [87], combined with the same SVM classification. We selected this approach, as it reports some of the best results from all scene classification approaches for datasets with *landscape*, *city* and *indoor* images and since it has already been proven to work on a significant enough dataset.

Two different representations are used for each binary classification tasks: color features are used to classify images as *indoor* vs. *outdoor*, and edge features are used to classify *outdoor* images as *city* or *landscape*. Color features are based on the LUV first- and second-order moments computed over a 10×10 spatial grid of the image, resulting in a 600-dimensional feature vector. Edge features are based on edge coherence histograms calculated on the whole image. Edge coherence histograms are computed by extracting edges in only those neighborhoods exhibiting some edge direction coherence, eliminating in this way areas where edges are noisy. Directions are then discretized into 72 directions,

and their histogram is computed. An extra non-edge pixels bin is added to the histogram, leading to a feature space of 73 dimensions.

In the three-class problem, this approach applies both methods in a hierarchical way [87]. Images are first classified as *indoor* or *outdoor* given their color representation. All correctly classified *outdoor* images are further classified as either *city* or *landscape*, according to their edge direction histogram representation.

3.5 Classification results

This section reports the classification performance of the BOV and the aspect-based image representations for the different scene and object classification tasks presented above. The influence of different parameters, related to the visterm construction and the aspect model is successively analyzed.

3.5.1 Image classification with bag-of-visterms

Binary scene classification

To analyze the effect of varying the size of the vocabulary employed to construct the BOV representation, we considered four vocabularies of 100, 300, 600, and 1000 visterms, denoted by V_{100}, V_{300}, V_{600}, and V_{1000}, respectively, and constructed from **D3** as described in Section 3.3. Additionally, four vocabularies V'_{100}, V'_{300}, V'_{600}, and V'_{1000} were constructed from **D4v**. Table 3.1 provides the classification error for the two binary classification tasks, *indoor/outdoor* and *city/landscape*, comparing the 8 BOV representations and the baseline. We can observe that the BOV approach consistently outperforms the baseline methods. This is confirmed in all cases with a paired T-test, for $p = 0.05$. Note that, contrarily to the baseline methods, the BOV representation uses the same features for both tasks, and no color information.

	indoor/outdoor	city/landscape
baseline	10.4 (0.8)	8.3 (1.5)
BOV V_{100}	8.5 (1.0)	5.5 (0.8)
BOV V_{300}	7.4 (0.8)	5.2 (1.1)
BOV V_{600}	7.6 (0.9)	5.0 (0.8)
BOV V_{1000}	7.6 (1.0)	5.3 (1.1)
BOV V'_{100}	8.1 (0.5)	5.5 (0.9)
BOV V'_{300}	7.6 (0.9)	5.1 (1.2)
BOV V'_{600}	7.3 (0.8)	5.1 (0.7)
BOV V'_{1000}	7.2 (1.0)	5.4 (0.9)

Table 3.1: Classification error for the baseline model and the BOV representation, for 8 different vocabularies. The size of vocabulary is varied (100, 300, 600 and 1000 visterms) and the K-means model are learned from the dataset **D3** or **D4v** for the V and V' vocabularies, respectively. Means and standard deviations over the ten splits are shown in parentheses.

Regarding the vocabulary size, we can see that for vocabularies of 300 visterms or more the classification errors are equivalent. The comparison of the rows 2-5 and 6-9 in Table 3.1 shows that using a vocabulary constructed from a dataset different than the one used for the classification experiments does not affect the results (error rates differences are within random fluctuation values) for these tasks. This result confirms the observations made in [94], and suggests that it might be feasible to build a generic visterm vocabulary that can be used for different tasks. Based on these results, we use the vocabularies built from **D3** in all the remaining experiments.

Three-class scene classification

Table 3.2 shows the results of the BOV approach for the three-class classification problem. First, we can see that the aspect-based representation significantly outperforms the combination of representations proposed in [87], according to a paired T-test with $p = 0.05$. Secondly, the classification performance does not vary significantly with vocabularies of 300 or more visterms, the vocabulary of 1000 visterms giving slightly better performance. Based on these observations, the vocabulary V_{1000} is chosen for all experiments in the rest of this chapter.

	indoor/city/landscape
baseline	15.9 (1.0)
BOV V_{100}	12.3 (0.9)
BOV V_{300}	11.6 (1.0)
BOV V_{600}	11.5 (0.9)
BOV V_{1000}	11.1 (0.8)

Table 3.2: Three-class classification error for the baseline and BOV representations. The baseline system is hierarchical (cf Section 3.4.4).

We show the confusion matrix of the three-class task in Table 3.3, when the vocabulary V_{1000} is used. *Landscape* images are well classified, but a confusion between the *indoor* and *city* classes exists. This can be explained by the fact that both classes share not only similar local image structures (which will be reflected in the same visterms appearing in both cases), but also similar visterm distributions, due to the resemblance between some more general patterns (e.g. doors or windows). The two images on the left of Figure 3.8 illustrate some typical errors made in this case, when *city* images contain a majority of geometric shapes and little texture. In the third place, the confusion matrix also tells us that *city* images are also misclassified as *landscape*. The main explanation is that *city* images often contain natural elements (vegetation like trees or flowers, or natural textures), and specific structures which produce many visterms. The two images on the right in Figure 3.8 illustrate typical mistakes in this case.

Five-class scene classification

Table 3.4 presents the confusion matrix obtained with the BOV approach in the five-class experiment, along with the baseline total classification error. The latter number was obtained using the edge coherence histogram global feature [87]. The BOV representation performs much better than the global features in this task, and the results show that we can apply the BOV approach to a larger number of scene classes and obtain good results. Note that a random class attribution would lead to an 80% error rate, and a majority class attribution (indoor in this case) to a 56% error rate. From the confusion matrix, we see that mistakes are made between the *forest* and *mountain* classes, reflecting

	classification performance (%)				
	indoor	city	landscape	class error (%)	# of images
indoor	89.7	9.0	1.3	10.3	2777
city	14.5	74.8	10.7	25.2	2505
landscape	1.2	2.0	96.8	3.1	4175

Total classification error: **11.1 (0.8)**, (baseline: 15.9 (1.0))

Table 3.3: Confusion matrix for the three-class classification problem, using vocabulary V_{1000}. Percentage of correctly classified and misclassified images is presented, along with the class dependent error-rates and the number of images per class. The total classification error is given below the table.

Figure 3.8: Typical classification errors of city images in the three-class problem. Left: city images classified as indoor. Right: city images classified as landscape.

	classification performance (%)						
	mountain	forest	indoor	city-panorama	city-street	class error (%)	# of images
mountain	85.8	8.6	2.5	0.5	2.6	14.2	590
forest	8.9	80.3	1.6	2.4	6.7	19.7	492
indoor	0.4	0	91.1	0.4	8.1	8.9	2777
city-panorama	3.5	1.8	8.0	46.9	39.8	53.1	549
city-street	2.0	2.2	20.8	6.0	68.9	31.1	1957

Total classification error: **20.8 (2.1)** (Baseline: 30.1 (1.1))

Table 3.4: Confusion matrix for the five-class scene classification problem, using vocabulary V_{1000}. Percentage of correctly classified and misclassified images in each class is presented, along with the class dependent error-rates and the number of images per class. The total classification error is given below the table.

the fact that they share similar textures, and the presence of *forest* in some *mountain* images. A second observation is that *city-panorama* images are often confused with *city-street images*. This result is not surprising given the ambiguous definition classes (see Figure 3.9) which was already perceived during the human annotation process. The errors can be further explained by the scale-invariant nature of the interest point detector, which makes no distinction between some far-field street views in the *city-panoramic* images, and close-to middle-view similar structures in the *city-street* images. Finally, the main source of confusion lays between the *indoor* images and the *city-street* images, for the same reasons exposed in the three-class case.

Seven-class object classification

The confusion matrix shown on Figure 3.5 shows that the classification performance, obtained with the V_{1000} visterm vocabulary, greatly depends on the object class. For instance, *trees* is a well defined class, dominated by high frequency texture visterms, and therefore does not get confused with other classes. Similarly, most *faces* image examples exhibit an homogeneous background and consistent layout which will not create ambiguities with other classes in the BOV representation. This explains the good performance of these two object classes. The *buildings* class shows the worst classification performance, with a classification error of 33.3% that is largely explained by the confusion with the

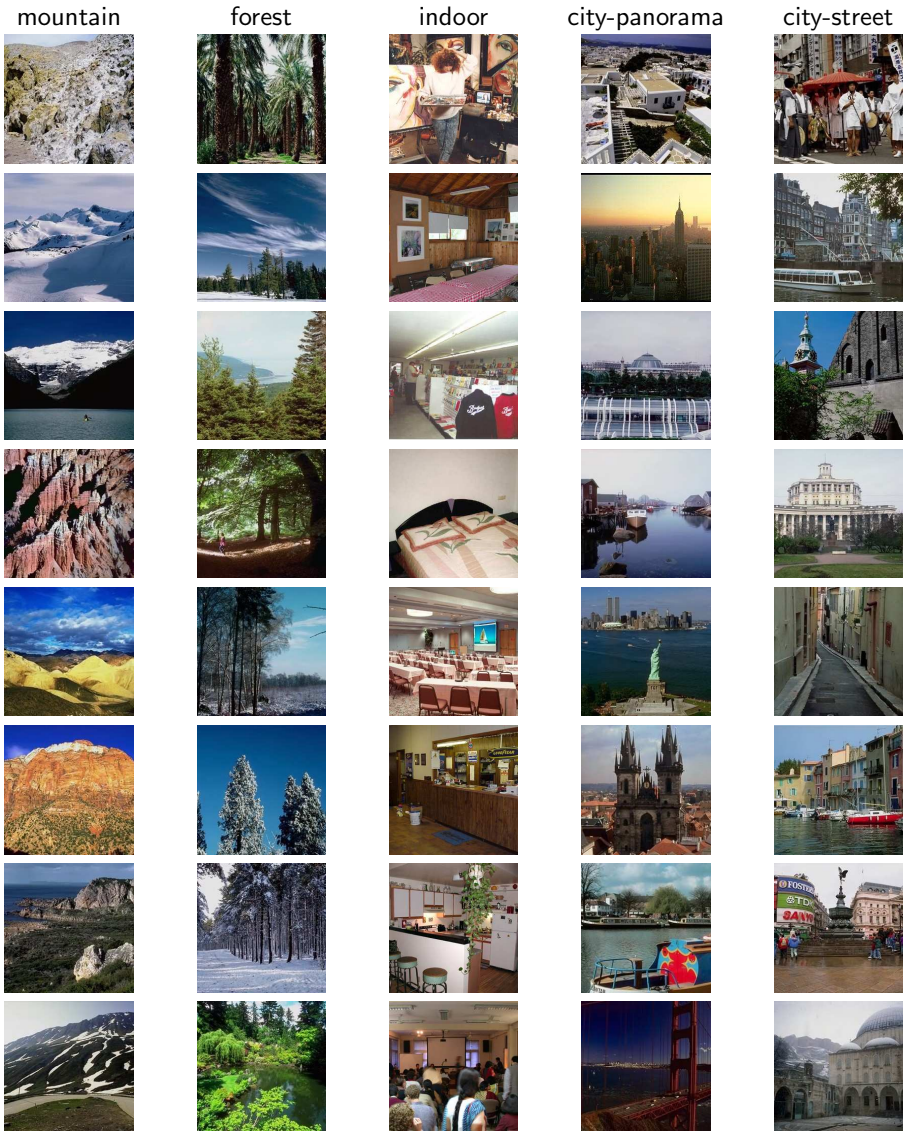


Figure 3.9: Illustration of the five scene classes, with 8 randomly selected examples per class. From left to right: *mountain*, *forest*, *indoor*, *city-panorama*, *city-street*. The definition of the *city-panorama* and *city-street* classes is debatable. All images have been cropped to square size for convenient display.

cars and *books* classes: the similarity between the *buildings* and *cars* backgrounds makes the bag-of-visterms representations of the two classes very similar; the vertical and horizontal structures from the *books* images also explains the confusion with the *buildings* class. Overall, the total classification error of 11.0% proves that the V_{1000} visterm vocabulary captures coherent information from each of the classes.

3.5.2 Image classification with the aspect-based representation

We use the aspect mixture weights $P(z_k|d_i)$ given each document d_i as a L dimensional image representation (Equation 3.3). Given that the PLSA parameters are estimated in an unsupervised way, this

	classification performance (%)								
	faces	buildings	trees	phones	cars	bikes	books	class error (%)	# of images
faces	97.5	0.3	0.9	0.4	0.4	0.3	0.4	2.5 (0.04)	792
buildings	4.0	67.7	4.0	3.3	8.0	3.3	10.6	33.3 (1.7)	150
trees	0.7	2.0	94.0	0.7	2.0	0.7	0	6.0 (0.6)	150
phones	6.5	0	0	85.2	2.8	0.9	3.2	13.4 (1.2)	216
cars	8.9	0.5	1.0	5.8	80.6	1.5	1.5	19.4 (1.5)	201
bikes	0	2.4	2.4	0.8	1.8	92.8	0	7.2 (0.4)	125
books	9.2	5.6	0	6.3	6.3	0.7	71.8	28.2 (1.9)	142

Total classification error: **11.0 (1.1)**

Table 3.5: Confusion matrix for the seven-class object classification problem, using vocabulary V_{1000}. Percentage of correctly classified and misclassified images in each class is presented, along with the class dependent error-rates and the number of images per class. The total classification error is given below the table.

representation can be inferred based on a model learned from unlabeled data, different from the one used to learn the SVM classifier. To test the influence of the PLSA training data on the classification performance, we propose to use two aspect-based image representations for the scene classification problems, which differ in the data used to estimate the initial PLSA model:

PLSA-I: For each dataset split, the labeled training data used to train the SVM classifier is used to learn the PLSA model. The aspect-based representation $a(d_i)$ is inferred for each test image, using the folding-in method described in Chapter 2.

PLSA-O: A single PLSA model is learned from the **D3** dataset, and the aspect-based representation $a(d_i)$ is inferred for each training and test document in each split, using the folding-in method described in Chapter 2.

As the dataset **D3** comprises *city*, *outdoor*, *indoor*, and *city-landscape* overlap images, a PLSA model learned on this set should capture valid latent aspects for all the scene classification tasks simultaneously. Only the PLSA-I representation is evaluated for the 7-class object classification problem, given that a majority of the object classes (eg. *phones*, *books*, *bikes* and *cars*) are not represented in **D3**.

Binary and three-class classification

Table 3.6 shows the classification performance of the aspect-based representation for 20 and 60 aspects for the PLSA-I and PLSA-O representations, using V_{1000}. The corresponding results for BOV with the same vocabulary are re-displayed for comparison purposes. The performance of the PLSA-I and PLSA-O representations is comparable for the *city/landscape* scene classification, while the PLSA-O representation improves over PLSA-I for the *indoor/outdoor* classification (paired T-test, with $p = 0.05$). This suggests that an aspect-based representation learned on the same set used for SVM training causes some over-fitting in the *indoor/outdoor* case. Since using PLSA-O allows to learn one single model for all tasks, we chose this approach for the rest of the scene classification experiments.

Comparing the 60-aspect PLSA-O model with the BOV approach, we remark that their performance is similar, and that PLSA performs better in the *city/landscape* case (although not significantly), while the opposite holds for the three-class task. An aspect-based representation with $L = 60$ corresponds to a dimensionality reduction of a factor of 17, while keeping the discriminant information contained in the original BOV representation. Note that PLSA-O representation with 60 aspects performs better than the BOV representation with the vocabulary V_{100} in all cases (see Tables 3.1 and 3.2).

	indoor/outdoor	city/landscape	indoor/city/landscape
BOV	7.6 (1.0)	5.3 (1.1)	11.1 (0.8)
PLSA-I ($L = 20$)	9.5 (1.0)	5.5 (0.9)	12.6 (0.8)
PLSA-I ($L = 60$)	8.3 (0.8)	4.7 (0.9)	11.2 (1.3)
PLSA-O ($L = 20$)	8.9 (1.4)	5.6 (0.9)	12.3 (1.2)
PLSA-O ($L = 60$)	7.8 (1.2)	4.9 (0.9)	11.9 (1.0)

Table 3.6: Comparison of the BOV, PLSA-I, and PLSA-O representations on the indoor/outdoor, city/landscape and indoor/city/landscape scene classification tasks, using $L = 20$ and $L = 60$ aspects. All experiments were done with vocabulary V_{1000}.

Table 3.7 displays the evolution of the error with the number of aspects for the *city/landscape* classification task. The performance is relatively independent of the number of aspects in the range [40,100], and $L = 60$ aspects will be considered for the rest of the chapter. For comparison purposes, we present in Table 3.8 the confusion matrix in the three-class classification task. The errors are similar to those obtained with the BOV (Table 3.3). The only noticeable difference is that more indoor images were misclassified in the city class.

L	20	40	60	80	100
Classification error	5.6 (0.9)	4.9 (0.8)	4.9 (0.9)	4.8 (1.0)	5.0 (0.9)

Table 3.7: Classification results for the *city/landscape* task, using different number of aspects for PLSA-O.

	classification performance (%)				
	indoor	city	landscape	class error(%)	# of images
indoor	86.6	11.8	1.6	13.4	2777
city	14.8	75.4	9.8	24.5	2505
landscape	1.3	1.9	96.8	3.1	4175
				Total classification error: **11.9 (1.0)**	

Table 3.8: Classification error and confusion matrix for the three-class problem using the PLSA-O representation with $L = 60$ aspects.

Table 3.9 compares classification errors for the BOV and the PLSA representations for the different tasks when the amount of labeled data to train the SVM classifier is decreased. The amount of training data is given both in proportion to the full dataset size, and as the total number of training images. The test sets remain identical in all cases.

For all image representations, a larger training set for the classifier translates in better results, showing the need for building large and representative datasets for training (and evaluation) purposes. Qualitatively, with the PLSA and BOV approaches, performance degrades smoothly initially, and degrades sharply when using 1% of training data. With the baseline approach, on the other hand, performance degrades more steadily. Comparing the different representations, we first see that PLSA with 10% of training data outperforms the baseline approach with full training set (i.e. 90%), this is confirmed in all cases by a paired T-test, with $p = 0.05$. BOV with 10% of training still outperforms the baseline approach with full training set (i.e. 90%) for *indoor/outdoor* (paired T-test with $p = 0.05$). More generally, we observe that both PLSA and BOV perform not worse than the baseline for -almost- all cases of reduced training set. An exception is the city/landscape classification case, where the baseline is better than the BOV when using 2.5% and 1% training data, and better than the PLSA

	Percentage of training data				
	90%	10%	5%	2.5%	1%
Indoor/Outdoor					
# of training images	8511	945	472	236	90
PLSA-O	7.8 (1.2)	9.1 (1.3)	10.0 (1.2)	11.4 (1.1)	13.9 (1.0)
BOV	7.6 (1.0)	9.7 (1.4)	10.4 (0.9)	12.2 (1.0)	14.3 (2.4)
Baseline	10.4 (0.8)	15.9 (0.4)	19.0 (1.4)	23.0 (1.9)	26.0 (1.9)
City/Landscape					
# of training images	6012	668	334	167	67
PLSA-O	4.9 (0.9)	5.8 (0.9)	6.6 (0.8)	8.1 (0.9)	17.1 (1.2)
BOV	5.3 (1.1)	7.4 (0.9)	8.6 (1.0)	12.4 (0.9)	30.8 (1.1)
Baseline	8.3 (1.5)	9.5 (0.8)	10.0 (1.1)	11.5 (0.9)	13.9 (1.3)
Indoor/City/Landscape					
# of training images	8511	945	472	236	90
PLSA-O	11.9 (1.0)	14.6 (1.1)	15.1 (1.4)	16.7 (1.8)	22.5 (4.5)
BOV	11.1 (0.8)	15.4 (1.1)	16.6 (1.3)	20.7 (1.3)	31.7 (3.4)
Baseline	15.9 (1.0)	19.7 (1.4)	24.1 (1.4)	29.0 (1.6)	33.9 (2.1)

Table 3.9: Comparison of classification performance for PLSA-O with 60 aspects, BOV with vocabulary V_{1000}, and baseline approaches, when using a SVM classifier trained with progressively less data. The amount of training data is first given in proportion of the full dataset, and then for each task, as the actual number of training images.

model for 1%. This can be explained by the fact that edge orientation features are particularly well adapted for this task, and that with only 25 city and 42 landscape images for training, global features are competitive. Furthermore, we can notice from Table 3.9 that the performance of the aspect-based representation deteriorates less as the training set is reduced than the BOV representation for all percentages (although not always significantly better). Previous work on probabilistic latent space modeling has reported similar behavior for text data [7]. The aspect-based representation describes an image as a mixture of visterm co-occurrence patterns, which is less affected by the visterm ambiguities discussed in Chapter 2 and Section 3.3.

Five-class scene classification

Table 3.10 reports the overall error rate and the confusion matrix obtained with PLSA-O in the five-class problem, and with the full training set. As can be seen, the aspect-based representation performs slightly worse than BOV, but still improves over the baseline. By comparing the confusion matrix with that of the BOV case (Table 3.4), we see that, while the *forest*, *mountain*, and *indoor* classification performance remains almost unchanged, the classification performance of the two city classes are significantly altered. In particular, the classification performance of the *city-panorama* class drops from 46.9% to 12.6%. As we already mentioned, these two classes contain very similar image examples (see Figure 3.9), and similar visterm co-occurrence patterns are thus identified within the two classes.

Table 3.11 presents the evolution of the classification error when less data is used to learn the SVM classifier. The loss of discriminative power between the *city-panorama* and *city-street* classes affects the PLSA-O representation, and in this particular case, the BOV representation outperforms the PLSA-O representation in this particular case. Both representations, however, perform better than the global image representation baseline.

	classification performance (%)						
	mountain	forest	indoor	city-panorama	city-street	class error (%)	# of images
mountain	85.5	12.2	0.8	0.3	1.2	14.5	590
forest	12.8	78.3	0.8	0.4	7.7	21.7	492
indoor	0.3	0.1	88.9	0.2	10.5	11.1	2777
city-panorama	3.6	4.9	8.8	12.6	70.1	87.4	549
city-street	1.6	1.4	20.4	1.7	74.9	25.1	1957

Total classification error: **23.1 (1.1)** (BOV: 20.8 (2.1), Baseline: 30.1 (1.1))

Table 3.10: Classification error and confusion matrix for the five-class classification problem using PLSA-O with 60 aspects, and using 90% training data to learn the SVM classifier.

Percentage of training data	90%	10%	5%	2.5%	1%
# of training images	5727	636	318	159	64
PLSA-O	23.1(1.2)	27.9(2.2)	29.7(2.0)	33.1(2.5)	38.5(2.6)
BOV	20.8(2.1)	25.5(1.7)	28.3(1.3)	30.8(1.6)	37.2(3.4)
Baseline	30.1(1.1)	36.8 (1.4)	39.3 (1.4)	42.8 (1.6)	49.9 (3)

Table 3.11: Comparison of the classification performance obtained with BOV, PLSA-O, and the baseline method, when using a SVM classifier trained with progressively less data on the 5-class problem.

Seven-class object classification

Table 3.12 shows the confusion matrix for the seven object classification problem and the per class error when the aspect-based image representation is used. As we already mentioned, only the PLSA-I representation is considered in this case, given that the bf D3 dataset is not representative of the different classes. The aspect-based representation for training images is thus learned from the SVM training data themselves, and inferred on the test data with the folding-in method. The total classification error (11.1%) is comparable to the one obtained with the BOV representation (11.0). The benefit of the aspect-based representation is shown in Table 3.13. The classification performance of the PLSA-I and BOV representations is reported for 90% 50%, 10% and 5% of training data. The aspect-based representation is learned from the 90% of training data each time. The total classification errors show that the aspect-based representation outperforms the BOV representation for the same amount of labeled data.

	classification performance (%)								
	faces	buildings	trees	phones	cars	bikes	books	class error (%)	# of images
faces	97.5	0.3	0.6	0.1	1.3	0.1	0.1	2.5 (0.02)	792
buildings	1.3	75.4	2	2	12	3.3	4	24.6 (1.4)	150
trees	2	2	93.3	0	1.3	1.3	0	6.7 (0.4)	150
phones	4.2	2.3	0	76.9	10.6	0.9	5	23.1 (0.6)	216
cars	7.0	2.5	0	1.5	85.6	2	1.5	14.4 (0.7)	201
bikes	0	2.4	3.2	0	3.2	90.4	0.8	9.6 (0.7)	125
books	4.9	9.1	0	4.2	9.9	0	71.8	28.2 (1.5)	142

Total classification error: **11.1 (1.6)** (BOV: 11.0 (2.1))

Table 3.12: Confusion matrix for the 7-class object classification problem using the PLSA-I representation with $L = 60$. The standard deviation over the ten splits is given in parentheses, and the number of image per class is given.

Table 3.9, 3.11 and 3.13 the aspect-based representation allows to take advantage of unlabeled data to improve the classification performance for the binary/three-class scene classification and the seven object classification tasks. This makes the aspect-based representation suitable for scenarios where data labeling is an expensive process, which is usually the case. The performance of a classification system will be less affected by the size of the labeled training set if the aspect-based representation is used.

Percentage of training data	90%	50%	10%	5%
# of training images	1598	888	178	89
PLSA-I ($L = 60$)	11.1(1.6)	12.5(1.5)	18.1(2.7)	21.7(1.7)
BOV	11.1(2.0)	13.5(2.0)	21.8(3.6)	26.7(2.8)

Table 3.13: Comparison between the bag-of-visterms (BOV) and the PLSA-based representation (PLSA) for classification with an SVM classifier trained with progressively less training data on the 7-class problem. The number in brackets is the variance over the different data splits.

3.6 Aspect-based image ranking

The aspect mixture weights can serve as an image representation which allows to take advantage of unlabeled data to improve the classification performance. Without any classification step, an intrinsic dependence between latent aspects and the classes that were considered in the previous section can be shown. We propose to rank the images in a dataset with respect to their probability given a latent aspect z_k, to illustrate what this aspect captures in the dataset. Assuming that $P(d)$ is uniform, $P(d_i|z_k)$ becomes proportional to the corresponding aspect mixture weight:

$$P(d_i|z_k) = \frac{P(z_k|d_i)P(d_i)}{P(z_k)} \propto P(z_k \mid d_i), \tag{3.4}$$

Given each latent aspect z_k, the top-ranked images according to $P(d|z_k)$ illustrate its potential 'semantic meaning'. Figure. 3.10 displays the 10 most probable images from the 668 test images of the first split of the **D1** dataset, for 7 out of 20 aspects learned on the **D3** dataset. The top-ranked images representing aspect 1, 6, 8, and 16 all belong to the *landscape* class. More precisely, aspect 1 is mainly related to horizon/panoramic scenes, aspect 6 and 8 to *forest/vegetation*, and aspect 16 to *rocks*. Conversely, aspect 4 and 12 are related to the *city* class. However, as aspects are identified by analyzing the co-occurrence of visterms that are local texture patterns in our case, they may be consistent from this point of view (e.g. aspect 19 is consistent in terms of texture) without allowing for a direct semantic interpretation.

The same image ranking procedure is used for the seven object dataset, and seven aspects-based ranking are shown on Figure 3.11, for a PLSA model learned on the **D6** dataset. We observe that aspects 3 and 17 are closely related to *face* images. The first ten images ranked with respect to aspect 8 are all *bike* images, while top-ranked images for aspect 10 mostly contain *phones*. *Buildings* are present in aspect 5, and all top-ranked images with respect to aspect 7 are *tree* images. Similarly to the ranking of scene one aspect (aspect #12) does not correspond to any specific object category.

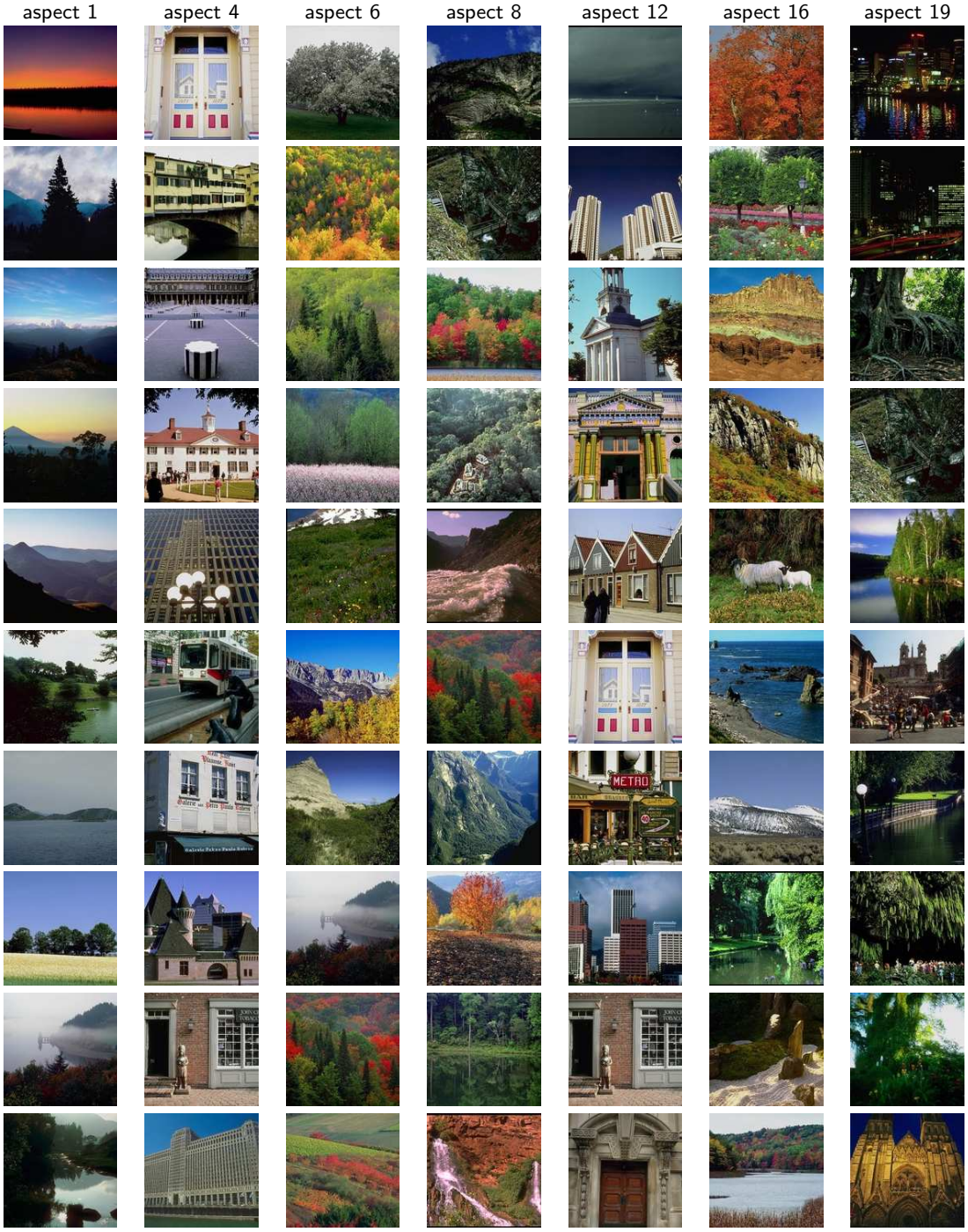


Figure 3.10: The 10 most probable images from the **D1** dataset for six aspects (out of 20) learned on the **D3** dataset. Aspect 1 relates to horizon images, aspects 4 and 12 relate to building structures, aspects 6, 8 and 12 relate to images containing vegetation/landscape, and aspect 19 relate to both man-made and natural structures that contain high frequencies. All images have been cropped to square size for convenient display.

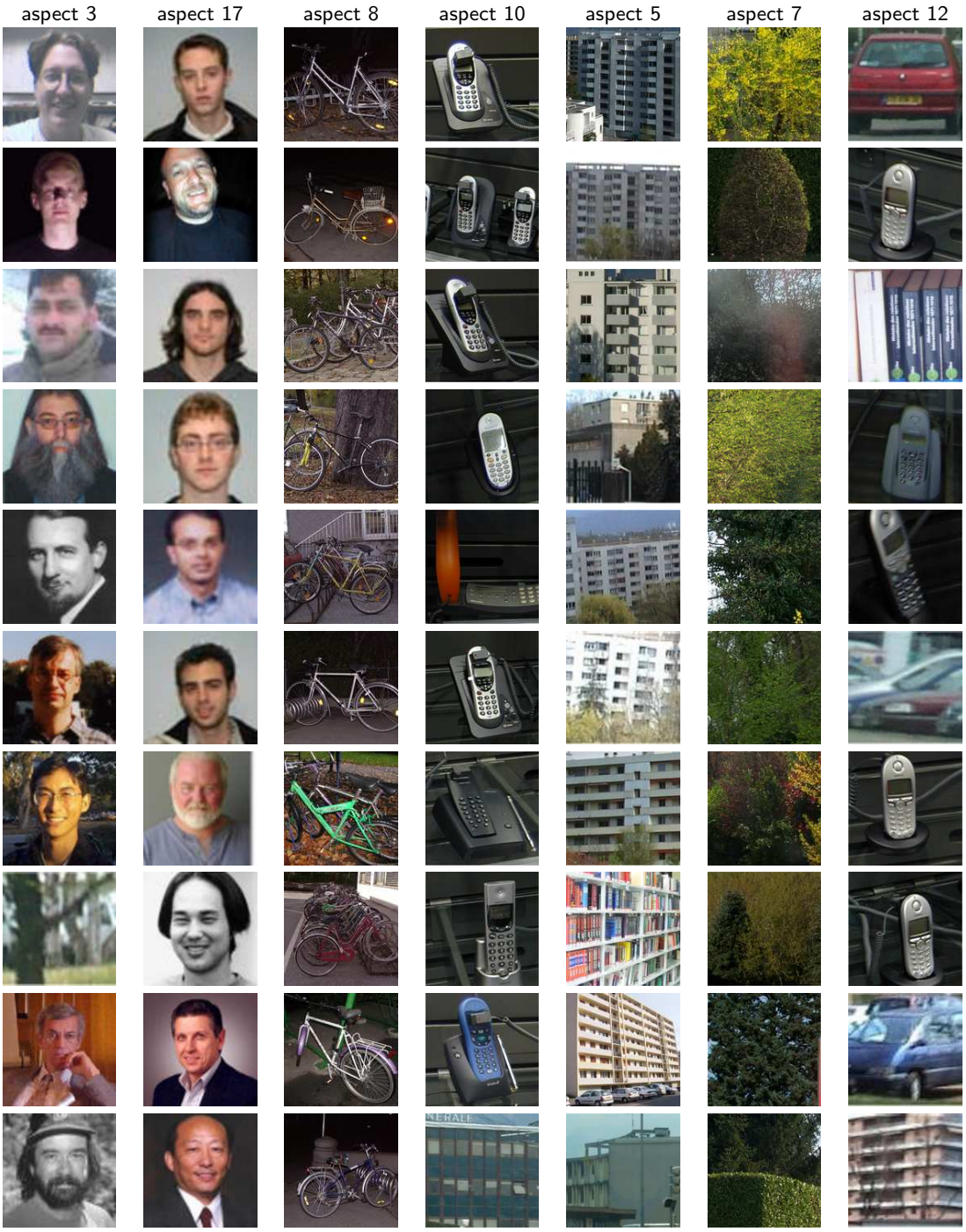


Figure 3.11: The 10 most probable images from the **D6** dataset, for seven aspects (out of 20) learned on the **D6** dataset. In this case, aspect 3 and 17 clearly relate to *faces*, aspect 8 relates to *bike* examples, aspect 10 mostly correspond to *phone* images, aspect 5 relates to *buildings*, aspect 7 corresponds to *trees*, and aspect 12 relates to a variety of visual content. All images have been cropped to square size for convenient display.

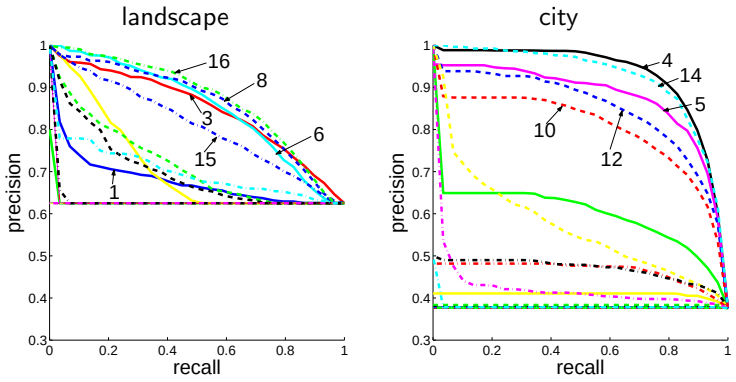


Figure 3.12: Precision/recall curves for the image ranking based on each of the 20 individual aspects, relative to the *landscape* (left) and *city* (right) query. Each curve represents a different aspect. Floor precision values correspond to the proportion of landscape(resp. city) images in the dataset. Note that the correspondence between aspects and visual concepts, observed on Figure 3.10 is confirmed: aspects 6, 8 and 16 are related to the *landscape* class; aspects 4 and 12 are related to the *city* class.

The aspect-based image rankings shown on Figure 3.10 and 3.11 give an indication of how much aspects can be related to a given type of image content. The quality of this ranking can be evaluated as an information retrieval system, measuring the correspondence between aspects and scene/object objectively. Defining the *Precision* and *Recall* paired values by:

$$Precision(r) = \frac{RelRet}{Ret} \quad Recall(r) = \frac{RelRet}{Rel},$$

where *Ret* is the number of retrieved images, *Rel* is the total number of relevant images and *RelRet* is the number of retrieved images that are relevant, we can compute the precision/recall curves associated with each aspect-based image ranking considering either *city* and *landscape* queries, as illustrated in Figure 3.12. Those curves prove that some aspects are clearly related to such concepts, and confirm observations made previously with respect to aspects 4, 6, 8, 12, and 16 on Figure 3.10. As expected, aspect 19 does not appear in either the *city* or *landscape* top precision/recall curves. The *landscape* related ranking from aspect 1 does not hold as clearly for higher recall values, because the co-occurrences of the visterm patterns appearing in horizons that it captures is not exclusive to the *landscape* class.

The same precision and recall curves are shown on Figure 3.13 for four of the seven object classes (eg. *face*, *cars*, *bikes* and *trees*) to measure the ranking obtained from $P(d \mid z_k)$. The top-left graph shows that the homogeneous ranking holds on for more than 10 retrieved images in aspect 3 and 17, confirming the observations made from Figure 3.11. We see that another aspect (13) is closely related to *faces* images. The top-right graph from Figure 3.13 shows that top-ranked images with respect to aspect 7 are mainly *tree* images. The bottom-left graph confirms that aspect 8 is linked to *bike* images, as well as aspect 1 even if less obvious. The bottom-right graph from Figure 3.13 shows that aspect number 12 is related to car images if looking deeper in the ranking, what is not obvious from the observation of Figure 3.11. Note however that the precision/recall values are not as high as for the *faces* case. Overall, these results illustrate that the latent structure identified by PLSA highly correlates with the semantic structure of our data. This makes PLSA potentially a very attractive tool for browsing/annotating unlabeled image collections.

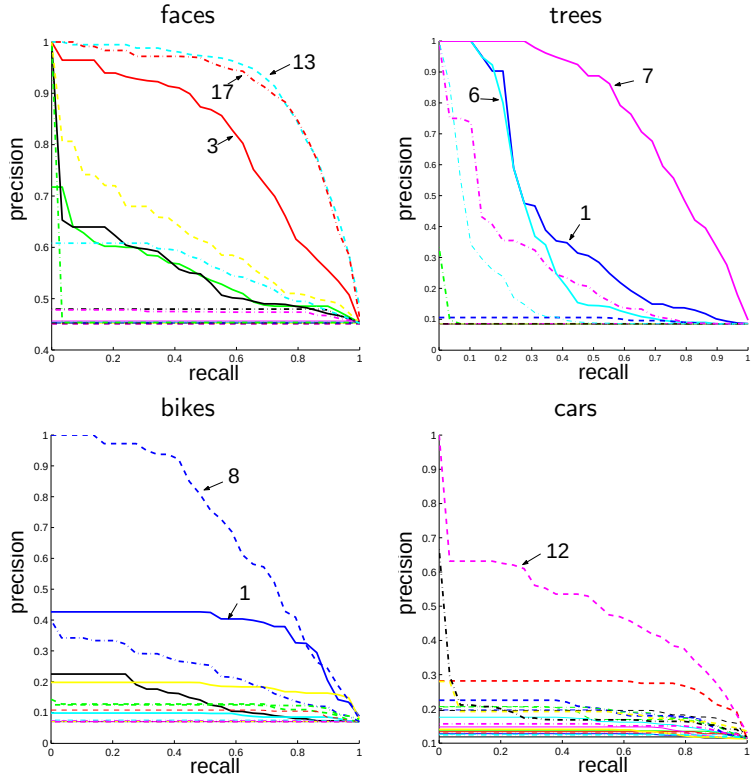


Figure 3.13: Precision and recall curves for the *face*, *car*, *bike* and *tree* categories, according to an aspect-based unsupervised image ranking. The lowest precision values correspond to the proportion of each class in the dataset. Note that the correspondence between aspects and visual concepts, observed on Figure 3.11 is confirmed: aspects 17 and 3 are related to the *faces* class; aspect 7 is related to the *trees* class; aspect 8 is related to the *bike* class. Interestingly, the aspect 12 appears to be related to the *cars* class, although no correspondence was visible on Figure 3.11.

3.7 Conclusion

The bag-of-visterms representation proposed in this chapter, relying on the combination of the DoG point detector and SIFT local descriptor, can capture a large variety of visual content, ranging from different scene types to various object classes. This was shown with a large number of classification experiments, for which the influence of the vocabulary size and the data used to learn this vocabulary has been analyzed in details.

We also showed that an aspect-based image representation learned from the bag-of-visterms representation can further improve the classification performance in cases when the available number of labeled images used to learn the classifier is decreased. The visterm co-occurrence patterns identified by the aspect model from unlabeled data allow to represent an image as a mixture of these visterm co-occurrence patterns that define an aspect. If the aspects are consistent with the object or scene classes, the resulting representation based on aspect mixture weights is more adequate for these reduced labeled data scenarios. We have illustrated what the aspects effectively capture in the dataset by ranking the images in a dataset given an aspect. This confirms that latent aspects can correspond to coherent information in an image collection, which can be seen as an interesting browsing structure for unannotated image collections.

Chapter 4

Contextual scene segmentation with aspect models

The aspect-based classification and the aspect-based image ranking, presented in Chapter 3, provided clear insights on what type of information is captured by latent aspect models in images. In these two tasks, only the aspect mixture weights $P(z \mid d_i)$ were exploited. In this chapter, we propose to take advantage of the aspect visterm distributions $P(v \mid z_k)$ to classify visterms in an image, producing a form of image segmentation. Figure 4.1 illustrates how the density of classified visterms can lead to the segmentation of a scene into a *man-made* and *natural* regions. We propose to include the visterm class information in the aspect model formulation, taking advantage of both the visterm distributions $P(v \mid z_k)$ and the aspect mixture weights $P(z \mid d_i)$ to classify visterms. The intuition is the following: two regions, indistinguishable from each other when analyzed independently, might be classified in the correct class with the help of the *context* captured by the aspect model. This form of context differs from the spatial context traditionally taken into consideration for image segmentation: no information about the neighborhood of each image region to classify is exploited, which is generally modeled by a Markov Random Field (MRF). The co-occurrence of visterms in an image, taken as a set, define the context that drives the classification of the visterm in the image. We consider a *man-made* vs. *natural* region classification task, two concepts that are well captured by the visterm representation, and show that the contextual information learned from the visterm co-occurrence improves the performance compared to a non-contextual approach. This constitutes a new application of the concept of mixture of aspects for images, and simultaneously illustrates the type of information that is captured by the latent aspects from a new perspective.

We discuss the closest related work in Section 4.1, Section 4.2 presents the baseline considered for visterm classification, and introduces the two methods to incorporate visterm class information in the aspect model. The visterm classification performance is evaluated in Section 4.3, with a comparison of the baseline to the two aspect-based visterm classification strategies. The spatial context of visterms in an image is exploited in Section 4.4 with an MRF model, combined with the other approaches. The work presented here was first presented in [55] and [56].

4.1 Related work

Image segmentation is a research field that has been developed for many years, with evolving goals that led to different approaches. Classic image segmentation is defined as a process of partitioning the image into non-intersecting regions, such that each region is homogeneous and the union of no two adjacent regions is homogeneous [65]. The main issue is defining the property for which we are imposing homogeneity. In most cases the properties on which segmentation is based are: grayscale, color, texture, or a combination of those properties. Image segmentation defined this way is performed

Figure 4.1: Scene segmentation by classifying visterms in an image: (a) Image containing *man-made* and *natural* structures; (b) local invariant regions (in yellow) are detected in the image, partially covering both the *man-made* and *natural* image regions; (c) after the quantization of the region descriptors into visterms, each visterm is classified in the *man-made* (in blue), or *natural* class (not shown).

on each image independently.

A review of traditional region-based and boundary-based approaches are given in [65], and it is not the purpose of this section to exhaustively review all the existing literature, but rather discuss closely related ideas. More recent alternatives have been proposed. For instance, Carson et al. [14] present a blob-based segmentation method that models the color, texture and position of all the pixels in a given image with a Gaussian mixture model (GMM), and attribute the label of its most likely GMM component to each pixel. This creates roughly homogeneous image regions called 'blobs', that are used for image retrieval, allowing the user to query the database at the blob level instead of the image level. In [44] a direct global optimization strategy is employed based on normalized cuts. Quantized local descriptors are used to build histogram representations of windowed image regions. The similarity between these regions is then defined based on this histogram representation, and segmentation is conducted for each individual image using a spectral clustering technique [44].

The perspective on image segmentation that we consider in this chapter differs from the above approaches in two main aspects. First, we intend to segment an image based on class labels that are predefined and applicable to the whole database, and not based on an homogeneity criterion of the regions in an image. Second, our segmentation is based on a set of small image regions that do not cover the whole image in general. The region descriptors are classified into categories, and the density of the region class labels gives a sparse segmentation of the image. We present a selection of image segmentation models that are based on class labels in the next paragraphs, with regions that cover the whole image [33, 18, 91, 92], or only a part of it [17, 36, 75].

The work in [18] relies on the Normalized Cuts segmentation algorithm [74] to segment the image into regions that are then quantized. Derived from the machine translation literature, an Expectation-Maximization (EM) estimates the probability distributions linking a set of words and blobs. Once the model parameters are learned, words are attached to each region. This *region naming* process is comparable to image segmentation.

Extending the Markov Random Field (MRF) model, Kumar and Herbert proposed a Discriminative Random Field (DRF) model that includes neighborhood interactions in the class labels, as well as at the observation level. They apply the DRF model to the segmentation of man-made structures in natural scenes [33], with an extraction of images features based on a grid of blocks that fully covers the image. The DRF model is trained on a set of manually segmented images, and then used to infer the segmentation into the two target classes.

Using a similar grid layout, Vogel and Schiele presented a two-stage classification framework to perform scene retrieval [91] and scene classification [92]. This work performs an implicit scene segmentation as an intermediate step, classifying each image block into a set of semantic classes such as

grass, *rocks*, or *foliage*.

In [17], invariant local descriptors are used for an object segmentation task. All region descriptors that belong to the object class in the training set are modeled with a Gaussian Mixture Model (GMM), and a second GMM is trained on non-object regions. In this non-contextual approach, new descriptors are independently classified depending on their relative likelihood with respect to the object and non-object models. A similar approach introducing spatial contextual information through neighborhood statistics of the GMM components collected on training images is proposed in [36], where the learned prior statistics are used for relaxation of the original region classification.

As an extension to local descriptors' representation of images, probabilistic aspect models have been recently proposed to capture descriptors co-occurrence information with the use of a hidden variable (latent aspect). The work in [23] proposed a hierarchical Bayesian model that extended LDA for global categorization of natural scenes. This work showed that important visterms for a class in an image can be found. However, the problem of region classification for scene segmentation was not addressed. The combination of local descriptors and PLSA for image segmentation has been illustrated in [75]. However this work has two limitations. First, visterms were classified into aspects, not classes, unless we assume as in [75] that there is a direct correspondence between aspects and semantic classes. This seems however a quite unrealistic assumption in practice. Secondly, evaluation was limited, e.g. [75] does not conduct any objective evaluation of the segmentation performance.

Unlike these previous approaches, we propose a formal way to integrate the latent aspect modeling in the class information. In addition, we explore the integration of the more traditional spatial MRF model into our system and compare the obtained segmentations.

4.2 Scene segmentation by visterm classification

The visterm construction presented in Chapter 3, based on the combination of the DoG point detector and the SIFT local descriptor, showed a good performance for *city* vs. *landscape* scene classification. Here, we consider the *man-made* vs. *natural* scene segmentation, and we therefore have recourse to the same visterm construction. As shown on Figure 4.1, the classification of regions sampled with an the DoG interest point detector produces a sparse image segmentation. However, this approach can take advantage of the interest point detection step to identify stable regions that should have a better correspondence accross the images than an arbitrary grid segmentation. We therefore use an interest point detector to sample the image regions that needs to be classified. Unlike [17] and [36], we quantize the region descriptors into a fixed number of image patches (visterms), clustering similar local descriptors into the same entity, as already described in Chapter 3.

We rely on likelihood ratio computation to classify each visterm v in a given image d into a class c. The ratio is defined by:

$$LR(v) = \frac{P(v|c = \text{man-made})}{P(v|c = \text{natural})}, \tag{4.1}$$

where the $P(v|c)$ probabilities will be estimated using different strategies. The visterm classification rule is :

$$LR(v) > T \Rightarrow v \in \text{man-made}, \tag{4.2}$$

where T is a threshold value.

4.2.1 Baseline: empirical distribution

Given a set of training data, the $P(v|c)$ probabilities can simply be estimated using the empirical distribution of visterms, as done in [17]. Given a set of *man-made* or *natural* image regions, the $P(v|c)$ probabilities are simply estimated as the number of times the visterm v appears in regions of class c, divided by the total number of occurrences of v. The empirical estimation is simple, but may suffer from several drawbacks. A first one is that a significant amount of labeled training data might be necessary to avoid noisy estimates, especially when using large vocabulary sizes. A second one is

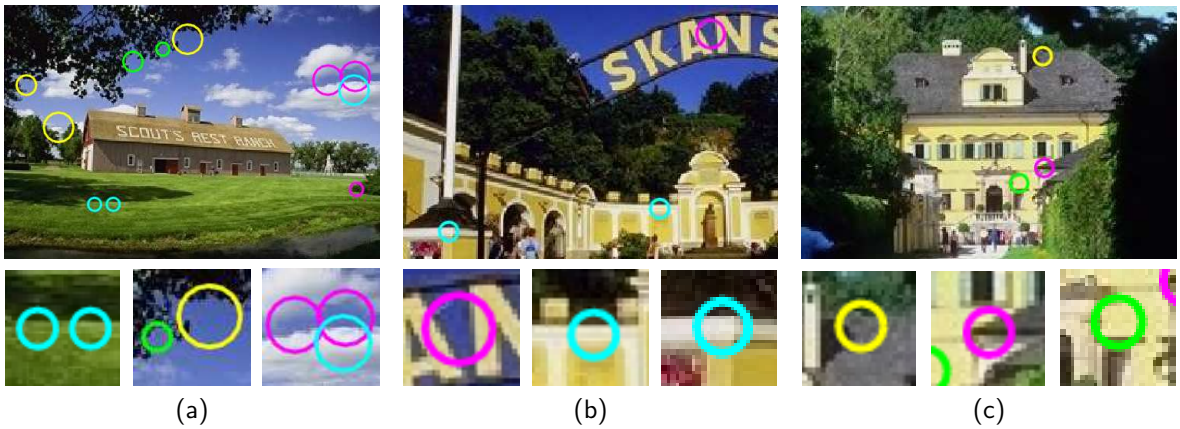


Figure 4.2: Regions (represented by visterms) can have different class labels depending of the images where they are found. (a) : various regions (4 different colors, same color means same visterm) that occur on *natural* parts of an image. (b) and (c) : the same visterms occur in man-made structures. All these regions were correctly classified by our approach, switching the class label for the same visterms depending on the context.

that such an estimation only reflects the individual visterm occurrences, and does not account for any kind of relationship between them. Visterms however describe regions extracted from full images, and should be interpreted in this context.

This visterm ambiguity is illustrated on Figure 4.2, where the same visterms appear both in *natural* (a) and *man-made* image regions (b) and (c), depending on the image. This situation, although expected since the visterm construction does not make use of class label information, constitutes a problematic form of visual polysemy, already mentioned in Chapter 3. Our postulate is that the type of co-occurrence context captured by aspect models could be used to identify the general context of an image, changing the classification of image regions based on this information. This is investigated in the following section.

4.2.2 Aspect and visterm class correspondence

As we have shown in Chapter 3 some aspects learned from *city* and *landscape* images do correlate with the *man-made* or the *natural* classes. The conditional distribution of visterms given an aspect $P(v \mid z)$ can therefore be exploited for the classification of visterms in an image once a class label is attached to the aspects. Based on the learned conditional distributions of visterms given aspects, the most likely aspect is attributed to a given visterm according to:

$$\begin{aligned} z_{v_j} &= \arg\max z(P(z|v_j)) \\ &= \arg\max z(\frac{P(v_j|z)P(z)}{P(v_j)}). \end{aligned} \tag{4.3}$$

In Figure 4.3, we show two examples of image segmentation based on the following procedure: from the image ranking illustrated in Chapter 3, we selected the ten aspects that are the more closely related to the *city* class and the ten aspects that are the more closely related to the *landscape* class. Restricting the aspect attribution to these 20 *man-made* and *natural* aspects, each visterm can be independently classified as a *man-made* or a *natural* descriptor, according to Equation 4.3. These two examples show a reasonable match between the ground-truth segmentation and the density of red and green points. The unsupervised learning based on co-occurrence thus allows to identify *man-made* and *natural* latent aspects in the data, that can be later used to classify visterms into these two classes.

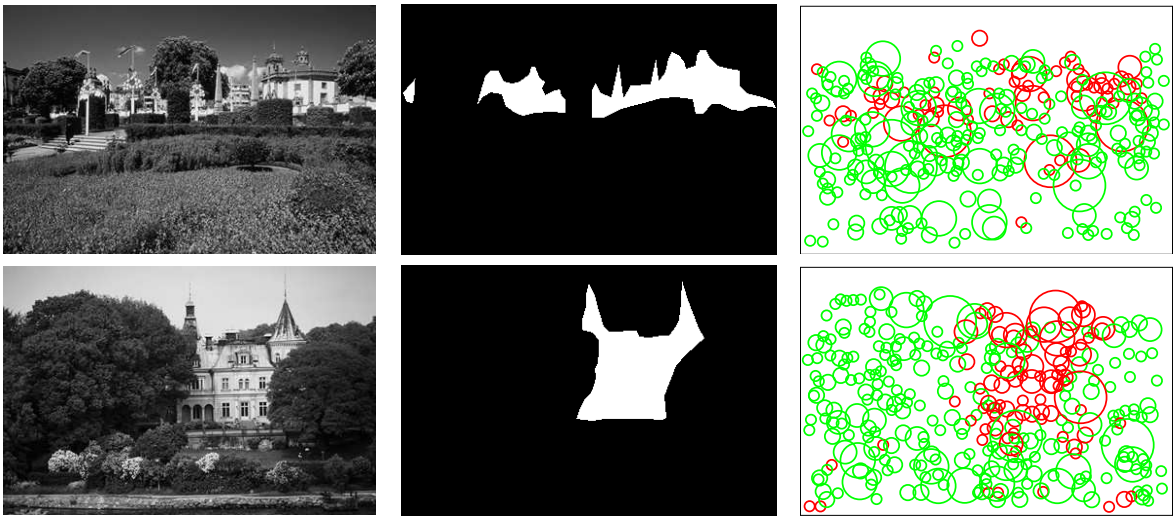


Figure 4.3: Classification of visterms based on the 10 aspects that are the more closely related to the *man-made* class, and the 10 aspects that are the more closely related to the *natural* class. The first column is the original image, the second column is the ground-truth segmentation (white is *man-made*, black is *natural*), and the last column is the segmentation based on classification of visterms. Red circles correspond to visterms classified as *man-made*, green circles correspond to visterms classified as *natural* (see text). The respective densities of red and green points show a good correspondence with the ground-truth segmentation.

Based on this first observation, we introduce two aspect models that estimate visterm class-likelihoods based on the decomposition of scenes in mixtures of aspects.

4.2.3 Aspect model 1

We propose to associate a class label c to each document-visterm observation pair, according to the graphical model shown in Figure 4.4, leading to the joint probability defined by:

$$P(c, d, z, v) = P(c)P(d|c)P(z|d)P(v|z). \tag{4.4}$$

This model introduces several conditional independence assumptions. The first one, traditionally encountered in aspects models, is that the occurrence of a visterm v is independent of the image d it belongs to, given an aspect z. The second assumption is that the occurrence of aspects is independent of the class the document belongs to. The parameters of this model are learned using the maximum likelihood (ML) principle [29]. The optimization is conducted using the Expectation-Maximization (EM) algorithm, allowing us to learn the aspect distributions $P(v|z)$ and the mixture parameters $P(z|d)$.

Given this model, the EM equations do not depend on the class label. Besides, the estimation of the class-conditional probabilities $P(d|c)$ do not require the use of the EM algorithm. We will exploit these points to train the aspect models on a large dataset (denoted $\mathcal{D}$) where only a small part of it has been manually labeled according to the class (we denote this subset by $\mathcal{D}_{lab}$). This allows for the estimation of a precise aspect model, while alleviating the need for tedious manual labeling. Regarding the class-conditional probabilities, as the labeled set will be composed of *man-made*-only or *natural*-only images, we simply estimate them according to:

$$P(d|c) = \begin{cases} 1/N_c & \text{if } d \text{ belongs to class } c \\ 0 & \text{otherwise,} \end{cases} \tag{4.5}$$

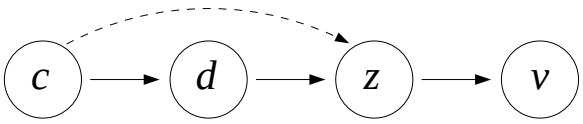


Figure 4.4: Aspect model 1 and aspect model 2 (if including dashed line)

where M_c denotes the number of images belonging to class c in the labeled set $\mathcal{D}_{lab}$. Given this model, the likelihood we are looking for (cf. Equation 4.1) can be expressed as

$$P(v|c) = \sum_{k=1}^{L} P(v, z_k|c) = \sum_{k=1}^{L} P(v|z_k)P(z_k|c), \tag{4.6}$$

where the conditional probabilities $P(z_k|c)$ can in turn be estimated through marginalization over labeled documents,

$$P(z_k|c) = \sum_{d \in \mathcal{D}_{lab}} P(z_k, d|c) = \sum_{d \in \mathcal{D}_{lab}} P(z_k|d)P(d|c). \tag{4.7}$$

These equations allow us to estimate the likelihood ratio defined in Equation 4.1, extending the PLSA model by introducing the class variable.

4.2.4 Aspect model 2

From Equation 4.6, we see that, despite the fact that the above model captures co-occurrence of the visterms in the distributions $P(v|z)$, the context provided by the specific image d has no direct impact on the likelihood. To explicitly introduce this context knowledge, we propose to evaluate the likelihood ratio of visterms conditioned on the observed image d,

$$LR(v, d) = \frac{P(v|d, c = \text{man-made})}{P(v|d, c = \text{natural})}. \tag{4.8}$$

The evaluation of $P(v|d, c)$ can be obtained by marginalizing over the aspects,

$$P(v|d, c) = \sum_{k=1}^{L} P(v, z_k|d, c) = \sum_{k=1}^{L} P(v|z_k)P(z_k|d, c), \tag{4.9}$$

where we have exploited the conditional independence of visterm occurrence given the aspect variable. Under the aspect model 1 assumptions, $P(z_k|d, c)$ reduces to $P(z_k|d)$, which clearly shows the limitation of this model to introduce both context and class information. To overcome this, we assume that the aspects depend on the class label as well (cf dashed link in Figure 4.4). The parameters of this model are the aspect multinomial distributions $P(v|z_k)$ and the mixture multinomial distributions $P(z|d, c)$, which could be estimated from labeled data by EM as before. However, as our model is not fully generative [7], only $P(v|z)$ can be kept fixed, and we would have to estimate $P(z|d_{new}, c)$ for each new image d_{new}. Since the class is obviously unknown for new images, this means that in practice all the dependencies between aspects and labels observed in the training data would be lost. To avoid this, we propose to separate the contributions to the aspect likelihood due to the class-aspect dependencies, from the contributions due to the image-aspect dependencies. Thus, we approximate $P(z_k|d, c)$ as:

$$P(z_k|d, c) \propto P(z_k|d)P(z_k|c), \tag{4.10}$$

where $P(z_k|c)$ is still obtained using Equation 4.7. The complete expression is given by

$$P(v|d,c) \propto \sum_{k=1}^{L} P(v|z_k)P(z_k|c)P(z_k|d). \tag{4.11}$$

The main difference with Equation 4.6 is the introduction of the contextual term $P(z_k|d)$, which means that visterms will not only be classified based on them being associated to class-likely aspects, but also on the aspect distribution of these aspects in the given image.

Inference on new images

With aspect model 1 (and also with empirical distribution, cf. baseline model in Section 4.2.1), visterm classification is done once for all at training time, through the visterm co-occurrence analysis on the training images. Thus, for a new image d_{new}, the extracted visterms are directly assigned to their corresponding most likely label. For aspect model 2, however, the likelihood-ratio $LR(v, d_{new})$ (Equation 4.8) involves the aspect mixture weights $P(z|d_{new})$ (Equation 4.11). Given our approximation (Equation 4.10), these parameters are inferred for each new image, with the folding-in PLSA method presented in Chapter 2: $P(z_k|d_{new})$ is estimated by maximizing the likelihood of the BOV representation of d_{new}, fixing the learned $P(v|z_k)$ parameters in the Maximization step.

4.3 Experiments and discussion

We validate our proposed models on the segmentation of scenes into *natural* vs. *man-made* structures. These two classes were chosen for our investigation, as we have shown that our visterm vocabulary is well suited for classifying *city* vs. *landscape* images and finding *city* and *landscape* aspects. The same visterms should therefore correspond to *natural* and *man-made* regions, and related latent aspects are very likely to exist. In this Section, we first present the baseline segmentation model, then our experimental setup. It is followed by detailed objective performance evaluation illustrated with segmentation results on a few test images.

4.3.1 Experimental setup

Data sets

Three image subsets from the *Corel Stock Photo Library* were used in the experiments:

$\mathcal{D}$: contains 6600 photos depicting mountains, forests, buildings, and cities. This dataset was used to construct the vocabulary, and learn the class-independent aspect model parameters $P(v \mid z)$ and $P(z \mid d)$.

$\mathcal{D}_{lab}$: contains 600 images from $\mathcal{D}$ that have been hand-labeled globally as *man-made* or *natural* images. These images were used to estimate the visterm likelihoods for each class.

$\mathcal{D}_{test}$: containing 485 images of man-made structures in natural landscapes, which were hand-segmented with polygonal shapes (Figure 4.1), was used to test the methods.

The empirical distribution is estimated from the images $\mathcal{D}_{lab}$, while aspect model 1 and aspect model 2 are learned from both $\mathcal{D}$ and $\mathcal{D}_{lab}$, according to the procedures described in Section 4.2.3 and 4.2.4.

Performance evaluation

The global performance of the algorithm was assessed using the True Positive Rate (TPR, number of positive visterms correctly classified over the total number of positive visterms), False Positive Rate (FPR, number of false positives over the total number of negative visterms) and True Negative Rate (TNR=1-FPR), where *man-made* is the positive class. The FPR, TPR and TNR values vary with the threshold T applied for classification (see Equation 4.2).

Parameter settings

From the scene classification experiments presented in Chapter 3, we showed that a vocabulary size of 1000 visterms, and PLSA model with 60 aspects allows to learn an effective image representation for the classification of images in the *city* and *landscape* classes. We use the same hyper-parameters to evaluate the visterm classification performance of the proposed methods.

4.3.2 Results

Figure 4.5 displays the Receiver Operating Curve (ROC, TPR vs. FPR) of the two aspect models and the empirical distribution (baseline). As can be seen, the aspect model 1 performs slightly better than the empirical distribution method, while aspect model 2 significantly outperforms the two other methods.

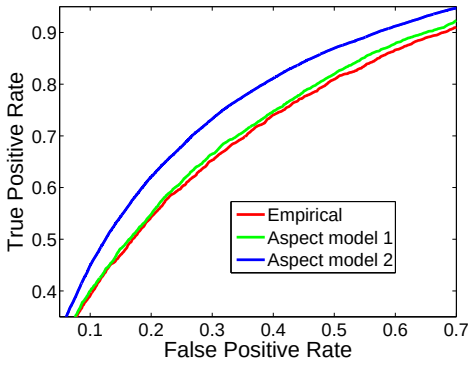


Figure 4.5: True Positive Rate vs. False Positive Rate for the three methods.

To further validate our approach, Table 4.1 reports the Half-Total-Recognition Rate (HTRR) measured by 10-fold cross-validation. For each of the folds, 90% of the test data $\mathcal{D}_{test}$ is used to estimate the likelihood threshold T_{EER} leading to Equal Error Rate (EER, obtained when TPR=TNR) on this data. This threshold is then applied on the remaining 10% (unseen images) of $\mathcal{D}_{test}$, from which the HTRR (HTRR=(TPR+TNR)/2) is computed. This table shows that the ranking observed on the ROC curve is clearly maintained, and that aspect model 2 results in a 7.5% relative increase in performance w.r.t. the baseline approach.

	Empirical distribution	Aspect model 1	Aspect model 2
HTRR	67.5	68.5	72.4

Table 4.1: Half Total Recognition Rate (in percent).

As mentioned in Section 4.2.2, aspect model 1 and the empirical distribution method assign specific visterms to the man-made or natural class independently of the images in which those visterms occur.

This sets a common limit on the maximum performance of both systems, which is referred here as the *ideal case*. This optimal performance is achieved for an optimal estimation of the visterm class likelihood, corresponding to the visterm class likelihood from the segmented test images. In our case, this *ideal case* provides an HTRR of 71.0%, showing that the visterm class attribution from empirical distribution and aspect model 1 is already close to the best achievable performance.

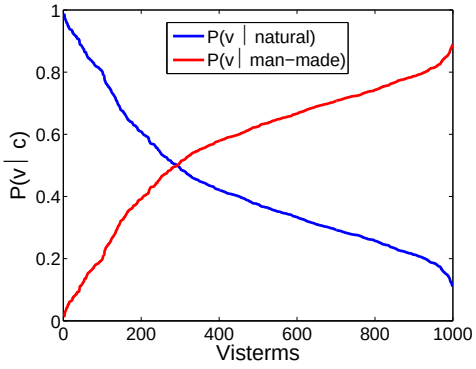


Figure 4.6: $P(v \mid c)$ for *man-made* and *natural* structures, estimated on the segmented test images. The x axis is the visterm indices ordered with decreasing $P(v \mid c = natural)$.

Indeed, the visterm class conditional probabilities shown in Figure 4.6 indicate that only a few visterms are class-specific. The class conditional $P(v \mid c)$ probabilities are obtained by dividing the number of visterm occurrences in one class by the number of that visterm occurrences in both classes. In Figure 4.6, these visterm class conditional probabilities were estimated from the segmented test images. For instance, all visterms appear at least 15% of time in regions labeled as *natural*. To perform better than the *ideal case* described above, visterms must be classified differently depending on the specific image that is being considered. This is the case with aspect model 2, which outperforms the *ideal case* in our experiments. Aspect model 2 switches visterm class labels according to the contextual information identified by the visterm co-occurrences in an image. In our dataset, aspect model 2 successfully switches visterm class labels depending on the image content for 792 out of the 1000 visterms in our vocabulary.

Segmentation examples

The impact of the contextual model can be observed on individual images. Figure 4.7 displays examples of *man-made* structure segmentation, where likelihood thresholds are estimated at EER value. As can be seen, aspect model 2 improves the segmentation with respect to the two other methods in two different ways. On one hand, in the first three examples, aspect model 2 increases the precision of the *man-made segmentation*, producing a slight decrease in the corresponding recall. On the other hand, the fourth example shows aspect model 2 producing a higher recall of *man-made* visterms while maintaining a stable precision. In the fifth example, the occurrence of a strong context causes the whole image to be taken as *natural*a scene, also improving the total visterm classification.

In Figure 4.8, five more examples of segmentation are shown. The first three rows illustrate *natural* image context examples that are correctly estimated by aspect model 2. The fourth row shows a correctly estimated marked *man-made* context that leads to an improved classification of visterms for aspect model 2. In the fifth example, however, the overestimation of the *man-made* related aspects leads to visterms that are dominantly classified as *man-made*. Nevertheless, overall, as indicated in Figure 4.5 and Table 4.1, the introduction of context by co-occurrence is beneficial.

Figure 4.7: Image segmentation examples at T_{EER}. Results provided by: first column, empirical distribution; second column, aspect model 1; third column, aspect model 2. The total number of correctly classified regions (man-made + natural) is given per image. The five rows illustrate cases where aspect model 2 outperforms the other approaches. In the fifth row, an extreme example of a strong natural context that is correctly identified by aspect model 2 leads to the classification of all regions as natural (though some should be labeled as man-made).

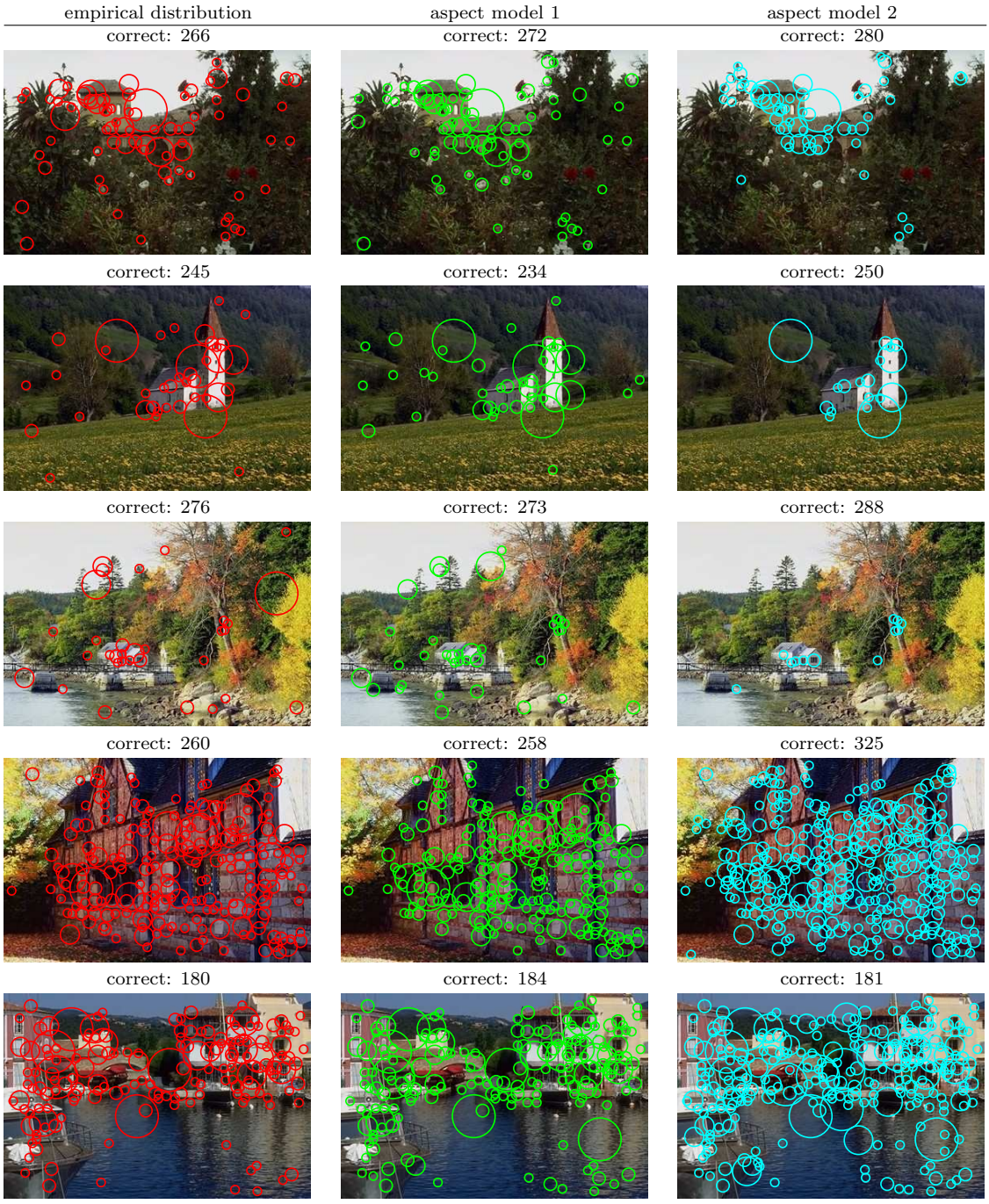


Figure 4.8: Image segmentation examples at T_{EER}. Results provided by: first column, empirical distribution; second column, aspect model 1; third column, aspect model 2. The first three rows illustrate the case of a correctly identified marked *natural* image context by aspect model 2, resulting in a more accurate visterm classification as compared to aspect model 1 and empirical distribution. The fourth row shows a correctly identified marked *man-made* image context by aspect model 2, with an improved number of correctly classified points. The last row shows the confusion of the region classification, when the context is not correctly identified (in this case, overestimated) by aspect model 2.

4.4 Combining co-occurrence and spatial contexts

The contextual modeling with latent aspects can be conveniently integrated with traditional spatial regularization schemes. To investigate this we present the embedding of our contextual model within the MRF framework [26], though other schemes could be similarly employed [34, 36].

4.4.1 Markov Random Field

Let us denote by S the set of sites s, and by $\mathcal{Q}$ the set of cliques of two elements associated with a second-order neighborhood system $\mathcal{G}$ defined over S. The segmentation can be classically formulated using the Maximum A Posteriori (MAP) criterion as the estimation of the label field $C = \{c_s, s \in S\}$ which is most likely to have produced the observation field $V = \{v_s, s \in S\}$. In our case, the set of sites is given by the set of interest points, the observations v_s take their value in the set of visterms $\mathcal{V}$, and the labels c_s belong to the class set $\{man-made, natural\}$. Assuming that the observations are conditionally independent given the label field (i.e. $p(V|C) = \prod_s p(v_s|c_s)$), and that the label field is an MRF over the graph $(S, \mathcal{G})$, then due to the equivalence between MRF and Gibbs distribution ($p(x) = \frac{1}{Z}e^{-U(x)}$), the MAP formulation is equivalent to minimizing an energy function [26]

$$U(C,V) = \underbrace{\sum_{s\in S} V_1(c_s) + \sum_{\{t,r\}\in\mathcal{Q}} V_1'(c_t,c_r)}_{U_1(C)} + \underbrace{\sum_{s\in S} V_2(v_s,c_s)}_{U_2(C,V)}, \tag{4.12}$$

where U_1 is the regularization term which accounts for the prior spatial properties (homogeneity) of the label field, whose local potentials are defined by:

$$\begin{aligned} V_1(\text{man-made}) = \beta_p \ \text{ and } \ V_1(\text{natural}) = 0, \\ V_1'(c_t,c_r) = \beta \ \text{ if } \ c_t \neq c_r, \text{ and } V_1'(c_t,c_r) = 0 \ \text{ otherwise.} \end{aligned} \tag{4.13}$$

β is the cost of having neighbors with different labels, while β_p is a potential that will favor the man-made class label (if $\beta_p < 0$) or the natural one (if $\beta_p > 0$), and U_2 is the data-driven term for which the local potential are defined by:

$$V_2(v_s,c_s) = -\log(p(v_s|c_s)). \tag{4.14}$$

To implement the above regularization scheme, we need to specify a neighborhood system. Several alternatives could be employed, exploiting for instance the scale of the invariant detector (e.g. see [36]). Here we used a simpler scheme: two points t and r are defined to be neighbors if r is one of the $N_{\mathcal{N}}$ nearest neighbors of t, and vice-versa. For this set of experiments we defined the neighborhood to be constituted by the five nearest neighbors. Finally, in the experiments, the minimization of the energy function of Equation 4.12 was conducted using simulated annealing [41].

4.4.2 Results

We investigate the impact of the regularization on the segmentation. The level of regularization is defined by β (a larger value implies a larger effect). The regularization is conducted by starting at the Equal Error Rate point, as defined in the 10-fold cross-validation experiments described in preceding Section. More precisely, for each of the folds, the threshold T_{EER} is used to set the prior on the labels by setting $\beta_p = -\log(T_{EER})$. Thus, in the experiments, when $\beta = 0$ (i.e. no spatial regularization is enforced), we obtain the same results as in Table 4.1. In Figure 4.9 we see that the best segmentation performance corresponds to an HTRR of 73.1% and a β of 0.35 with the empirical modeling, and an HTTR of 76.3% for a β of 0.2 and aspect model 2. This latter value of β is chosen for all the MRF illustrations reported in Figure 4.10 and 4.11.

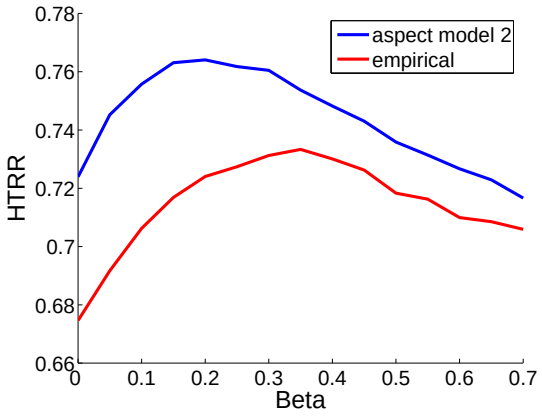


Figure 4.9: Half Total Recognition Rate for different β values.

The inclusion of the MRF relaxation boosted the performance of both aspect model 2 and empirical distribution. However, it is important to point out that aspect model 2 still outperforms the empirical distribution model, though the boosting benefited most to the empirical distribution modeling. This was to be expected, as aspect model 2 was already capturing some of the contextual information that the spatial regularization can provide (notice also that the maximum is achieved for a smaller value of β in aspect model 2).

Besides obtaining an increase of the HTRR value, we see a better spatial coherence of the segmentation, as can be seen in Figure 4.10 and 4.11. The MRF relaxation process reduces the occurrence of isolated points, and tends to increase the density of points within segmented regions. We show on the last row of Figure 4.10 that, as can be expected when using prior modeling, the MRF step can over-regularize the segmentation, causing the attribution of a single class to all the visterms in an image.

Figure 4.10: Effect of the MRF regularization on the man-made structure segmentation. The first three rows illustrate the benefit of the MRF regularization where wrongly classified isolated points are removed. The last row shows the deletion of all man-made classified regions from an image when natural regions dominate the scene.

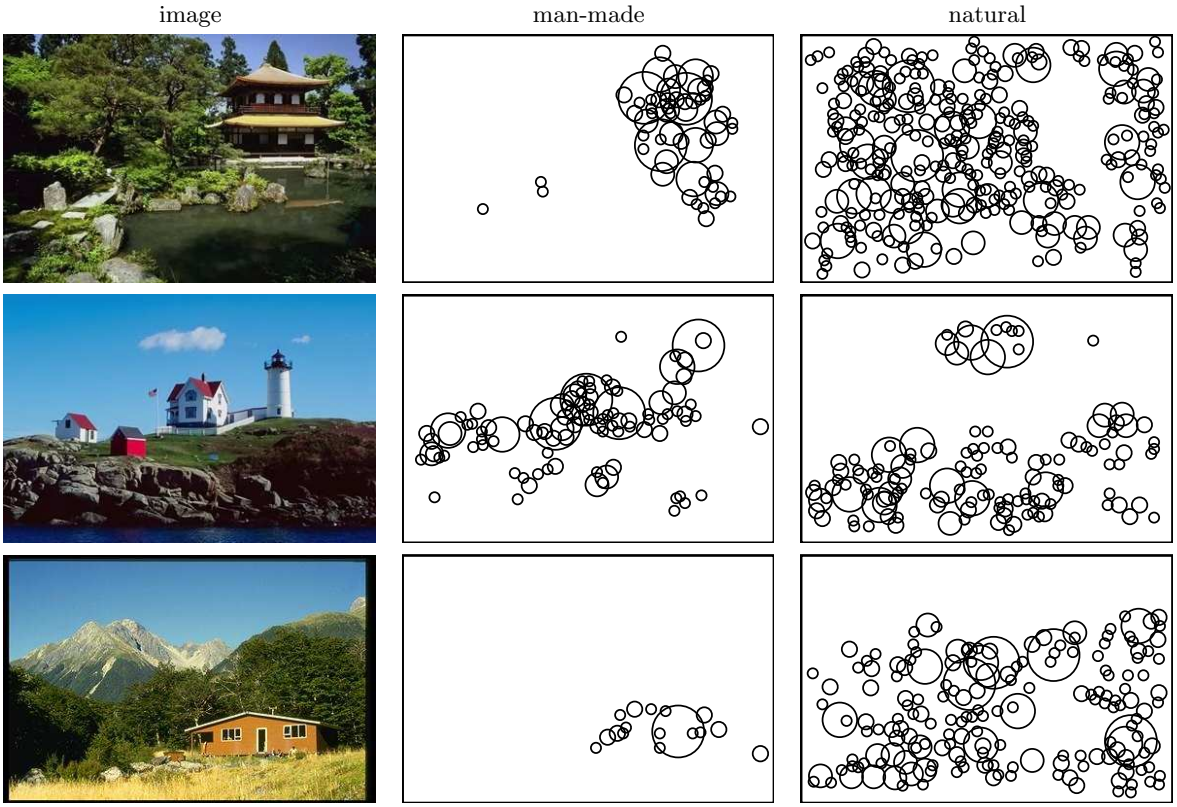


Figure 4.11: Three other examples that illustrate the final segmentation obtained with aspect model 2 and MRF regularization. The display is different than in previous figures to avoid image clutter.

4.5 Conclusion

After investigating the aspect mixture weights learned from a bag-of-visterms representation for image classification, we have proposed to take advantage of the co-occurrence context identified by an aspect model to classify visterms in an image. The interpretation of an image region certainly depends on the image it appears in, and any form of image-related context therefore helps their classification. Considering a *man-made* vs. *natural* visterm classification task, we proposed an aspect-based formulation of the problem by adding a class label to each document-visterm observation pair. Our experiments proved that, if the aspect mixture weights are taken into account, the classification of visterms in an image depends on what the other visterms in this image are. The proposed approach integrates this intuition and outperforms an image-independent visterm classification. This visterm co-occurrence information represents a novel form of context exploited for image segmentation, different from the spatial context traditionally considered. Moreover, this additional co-occurrence information has been successfully combined with the traditional spatial context modeled with a MRF. We can also envision other ways of integrating the two types of context.

Other types of visterms should be considered for different scene segmentation tasks. Visterms built from SIFT descriptors are particularly well suited for the classification of *man-made* vs. *natural* regions, as these are characterized by specific textures. Other classes might require the use of additional - or different - information for building a valid visterm vocabulary for segmentation.

Chapter 5

Aspect models for image annotation

The concept of images as mixtures of latent aspects has been introduced in Chapter 2, in perspective with the concept of mixture of aspects for text documents. Text collections are indeed the standard data on which the aspect models are traditionally applied, and the chapters 3 and 4 investigated and justified their use in the context of image collections. In this chapter, we propose to model the textual and the visual modalities of annotated images with a single aspect model, sharing the same aspect mixture weights $P(z \mid d_i)$ for the two modalities. Three learning procedures are investigated, differing in which modality is used to estimate the aspect mixture weights for each image. Once an aspect model for the two modalities has been learned, a distribution over words given a new, unnannotated image can be inferred. Moreover, two other types of visterms than the one used in the chapters 3 and 4 are successively evaluated and combined with the DoG+SIFT visterms. The work presented here appeared originally in [52] and [53].

5.1 Automatic image annotation

Automatic image annotation systems take advantage of existing annotated image datasets to build a link between the visual and the textual modalities. While this framework seems very close to standard object detection [89, 1], key differences make automatic image annotation a distinct research problem. Although the vocabulary - the set of valid annotation words - might be constrained, captions from image collections can exhibit a large variability in general. Several words can describe one or more regions or even the whole image (see Figure 5.1), which differs from the standard scene and object classification scenarios in Chapter 3. The manual segmentation into positive and negative examples for supervised training is not as straightforward as for the face detection case (see Figure 5.1). Furthermore, the development of class-specific features and classifiers [90] is difficult, as the vocabulary size is usually much larger than the number of classes in standard object detection problems. Automatic image annotation systems therefore tend to rely on generic features, and usually learn one model for the whole vocabulary [5, 6, 35, 24, 32, 57, 18, 31, 66, 51].

Independently of what features are chosen, the question is how to model the relation between captions and visual features to achieve the best textual indexing. A whole range of methods, from a simple empirical distribution estimation to complex generative probabilistic models, have been proposed in the literature, offering a large variety of approaches. However, the difference in the nature of text captions and image features has not yet been fully investigated and exploited. In general, the textual and visual modalities are either considered as equivalent sources of data [57, 18, 51, 35, 66], or the caption words are simply considered as a class label [81, 15, 40] instead of a modality as such. The Corr-LDA (Correspondence Latent Dirichlet Allocation) [6] model is a notable exception, that builds a language-based correspondence between text and images. It first generates a set of hidden variables (latent aspects) that generate the regions of an image, decomposing an image into a mixture

flowers, garden, house, tower | buildings, people, sky, street | clouds, formation, sky, sunset

Figure 5.1: Typical image captioning in the *Corel Stock Photo Library*.

of latent aspects. A subset of these latent aspects is then selected to generate the text caption, what intuitively corresponds to the natural process of image annotation.

The Corr-LDA model acknowledges the complementarity of text and images as sources of information, as well as their difference in carrying semantic content, which needs be taken into account to model the relation between modalities more accurately, with the goal of generating a better textual indexing. This chapter investigates this concept, proposing a new dependence between words and image regions based on latent aspects. The contributions of this chapter are the following. First, we present an alternative image representation to the standard Blob histogram, that combines quantized local color information and quantized local texture descriptors. Quantized versions of invariant local descriptors have been recently proposed as promising representations of objects and scenes [68, 23, 75], and applied to small number of classes. However, to our knowledge, this representation has not been previously used in the context of image annotation, a more challenging problem from the number of concepts that is addressed.

The effect of each type of visual features and their combination is analyzed in details, and we prove their complementarity by demonstrating improvement of the retrieval performance for a majority of word queries for all the models that we consider. Second, we propose a probabilistic framework to analyze the contribution of the textual and the visual modalities separately. We assume that the two modalities share the same conditional probability distribution over a latent aspect variable, that can be estimated from both or one of the two modalities for a given image. In this way, equal importance can be given to the visual and the textual features in defining the latent space, or one of the two modalities can dominate. Based on extensive experiments, this framework allows us to show that the textual modality is more appropriate to learn a semantically meaningful latent space, what directly translates into an improved annotation performance. Finally, a comparison between different recently proposed methods is presented, and a detailed evaluation of the performance shows the validity of our framework.

The chapter is organized as follows. Section 5.2 presents an overview of the research in automatic image annotation and contrasts it with our work. Section 5.6.1 discusses the data and the visual representation considered in this work. Section 5.4 describes our probabilistic framework for image annotation. In Section 5.5 we discuss state-of-the-art models that we implemented for comparison. Results and discussion are presented in Section 5.6.

5.2 Related work

Existing works in automatic image annotation can be differentiated by the way in which they represent images, and by the specific auto-annotation model. These two aspects are used to guide the discussion in the following paragraphs.

A common first step to all automatic image annotation methods is the image segmentation into regions, either using a fixed grid layout or an image segmentation algorithm. Regions have been described by standard set of features including spatial frequencies, color, shape and texture, and

handled as continuous vectors [5, 6, 35, 24, 32, 40], or in quantized form [57, 18, 31, 66, 51]. Different statistical assumptions about these quantized or continuous representations and image captions have led to different models. A representative selection of recent approaches is presented here.

The original approach described in [57] is based on a fixed grid image segmentation and a vector quantization step. The color and texture representations of all training image blocks are quantized into a finite set of visual terms (*visterms*), which transforms an image into a set of visterms. All words attached to an image are attributed to its constituting visterms, and the empirical distribution of each word in the vocabulary given all visterms is computed from the set of training documents. A new image is indexed by first computing its building visterms and then averaging the corresponding posterior distributions over words.

Contrarily to [57], the work in [18] relies on the Normalized Cuts segmentation algorithm [74] to identify arbitrary image regions and build blobs. These blobs coarsely match objects or object parts, which naturally brings up a new assumption: the existence of an implicit one-to-one correspondence between blobs and words in the annotated image. The idea is borrowed from the *machine translation* literature, and considers the word and blob modalities as two different languages. An Expectation-Maximization (EM) procedure to estimate the probability distributions linking words and blobs is proposed. Once the model parameters are learned, words can be attached to a new image region. This *region naming* process is comparable to object recognition, even if regions do not necessarily match objects in the image, due to the obvious limitations of the segmentation algorithm. A new image is annotated by the most probable words given its constituting blobs.

The *cross-media relevance model* described in [31], also relies on a quantized blob image representation. However, unlike [18], it does not assume a one-to-one correspondence between blobs and words in images. Images are considered as sets of words and blobs, which are assumed independent given the image. The conditional probability of a word (resp. blob) given a training image is estimated by the count of this word (resp. blob) in this image smoothed by the average count of this word (resp. blob) in the training set. These posterior distributions allow the estimation of the probability of a potential caption (set of words) and an unseen image (set of blobs) as an expectation over all training images. This annotation system improves the performance w.r.t the machine translation method [18].

Linear algebra-based methods applied on the word-by-document and Blob-by-document matrices are proposed in [66] to estimate the probability of a keyword given a blob. The correlation and the cosine measure between words and blobs are investigated to derive these conditional probabilities. The use of a Singular Value Decomposition (SVD) of the word-by-document and blob-by-document matrices, weighted with the *tf-idf* (term frequency - inverse document frequency) scheme, shows an improvement of the annotation performance over the original data representation. A consistent improvement over the model based on machine translation [18] is shown.

In [35] and [24], the authors of [31] abandon the quantization of image regions. With the same conditional independence assumptions than in their previous model [31], the continuous image region representation, modeled by a Gaussian Mixture Model (GMM), improves the image annotation performance. An additional modification is proposed in [24], where a multiple-Bernoulli distribution for image captions replaces the multinomial distribution.

A statistical model of 600 image categories is proposed in [40]. Categories are labeled with multiple words, and images are manually classified in these categories. A two-dimensional Multi-resolution Hidden Markov Model (2D-MHMM) is learned on a fixed-grid segmentation of all category examples. The likelihood of a new image given each category's 2D-MHMM allows to select caption words for this image. This work is related to the *model vector indexing* approach [81], where one classifier (Support Vector Machine) is trained for each semantic concept (7 concepts), and used for the indexing of a new image. The *Content-based soft annotation* (CBSA) system [15] is also based on binary classifiers (Based Point Machines and SVMs) trained for each word (116 words are considered), and index a new image with the output of each classifier. The drawbacks are the learning of one classifier per word [81, 15], or of one model per set of words [40].

Different models to represent the joint distribution of words and image regions are discussed in [5, 6]. A hidden *aspect* variable is assumed in the data generative process, which links the textual and

visual modalities through conditional relationships. This assumption translates into several variations of Latent Dirichlet Allocation (LDA) based mixture models. Images are represented as a set of continuous region-based image features, and modeled by Gaussian distributions conditioned on the aspects, while caption words are modeled with multinomial distributions. For instance, in the Corr-LDA model [6], words are conditioned on aspects that generated image regions. This additional constraint on word generation improves the overall annotation performance over less constrained LDA-based models.

A whole range of performance measures for automatic image annotation systems has been discussed in the literature. The quality of short image captions ($\leq$5 words), intended to be similar to human annotations, has been evaluated with different measures [15, 18, 66, 31, 5, 51]. Specifically, the retrieval of images based on these short captions is evaluated with the precision and recall values of the retrieved image sets for a number of given queries in [81, 66, 18, 31]. Alternatively, the ratio of the correctly predicted words per image divided by the number of words in the ground truth annotation has also been used for the evaluation of short captions [66, 5]. Proposed by [5], another measure for caption evaluation is the *Normalized Score*, which depends on the number of predicted words, and allows to estimate the optimal number of words to predict [51]. The creation of short, human-like text captions is justified when the task is related to object recognition. A few text labels are attached to the new image, possibly describing the image content accurately. However, the main goal of image annotation is to allow text-based queries for image retrieval, and this does not require the creation of binary text captions. All approaches (binary classification, probabilistic model, linear-algebra-based) actually provide a confidence value for each word, that can be used for ranking all images in a collection. The confidence values for each word enables the construction of an image index, that can be used for text-based image retrieval [81, 15, 31]. The average precision of a query (see Section 5.6.2), summarized by the mean average precision (mAP) for a set of queries, is then the natural metric for the retrieval performance. This way of annotating/indexing images and evaluating retrieval performance has started to become consensual [81, 31], and we therefore use it in this chapter.

As it should be clear from this overview, the existing approaches proposed to learn relationships between visual and textual modalities in annotated images differ in the way images are represented, in the dependence assumptions that are made between image regions and words, and in the way model learning is performed. In this chapter, we propose a probabilistic framework related to [5] and [6] that includes a hidden aspect variable to link the visual and textual modalities. This approach allows to consider regions and words from an image jointly, contrarily to [57], where image regions are considered independently, and to [40] and [81], where categories (or words) are treated independently. Moreover, given that only one model is learned for all the words in the vocabulary, this type of approach might be better suited for large vocabularies than the supervised learning procedures proposed in [40, 81], which need to learn one model for each word. Finally, words and image features are of different nature and carry quite distinct level of semantics, and so we believe that these differences should influence how these two modalities are learned. Words and blobs are assumed equivalent in [18] (translation between two languages), and are treated equivalently in some of the models described in [5] and [6]. Unlike these works, we investigate different possibilities of learning the two modalities jointly while changing their respective influence.

In this sense, the closest work to ours is Corr-LDA, which first samples a latent aspect variable to generate an image region from a conditional Gaussian distribution, and then samples an aspect from the same set of aspects to select a word from a conditional multinomial distribution. In contrast, in our work we do it differently because we use multinomial distributions conditioned on aspects to model the discrete visual features, and therefore we have the possibility to model a similar data generative process as Corr-LDA, or to first generate the words and learn the related aspect distributions that we later link to the visual features. As stated in the introduction, We also propose an enriched image representation that includes quantized local image descriptors that has not been investigated in auto-annotation, but used in very recent work for scene and object classification [23, 75, 68]. We conduct a thorough study comparing various competitive methods with a consistent evaluation procedure, and we show an improvement of performance.

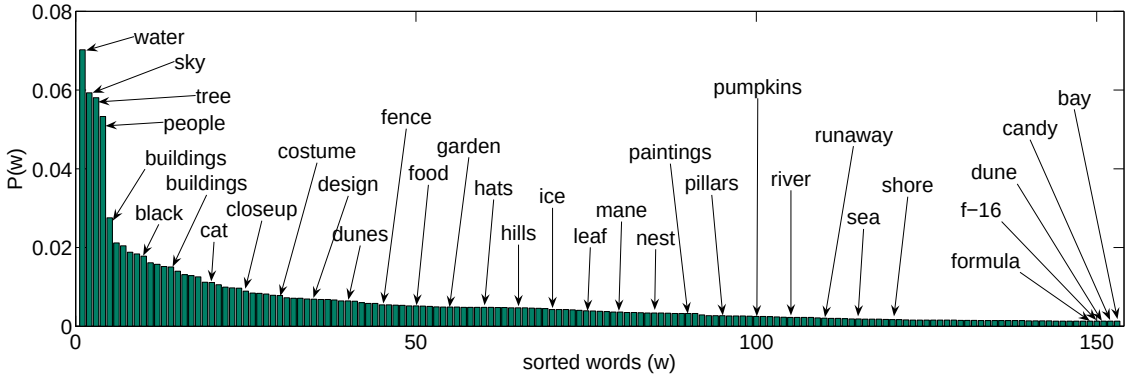


Figure 5.2: Empirical distribution of words in the training images (set #1). The most common words are water (1124), sky (949), tree (929), people (853), and buildings (441). The least common words are formula (21), f-16 (21), dunes (21), candy (21) and bay (21). The numbers in brackets indicate the number of images in which each word occurs. Some other words, whose number of occurrence ranges between these two extremes, are shown to illustrate the nature of the vocabulary.

5.3 Annotated image representation

5.3.1 Text caption representation

Images in our dataset are annotated with a few unordered words selected from a vocabulary of size N_w. The representation of the caption of an image d_i is an histogram $w(d_i)$ of size N_w :

$$w(d_i) = \{n(d_i, w_1), ..., n(d_i, w_j), ..., n(d_i, w_{N_w})\}, \tag{5.1}$$

where $n(d_i, w_j)$ denotes the count of the word w_j in the caption of the image d_i. This is a standard representation for text documents, that could also be used in the case of free-text captions after the word stopping and word stemming preprocessing steps. As shown in Figure 5.2, the distribution of words is highly skewed. As the dataset mainly consists of outdoor images, the words *water*, *sky*, *tree*, *people*, and *buildings* account for a big proportion of the probability mass. The empirical distribution also shows that there are many words represented by only a few training examples that nevertheless will have to be predicted, what advocates for a model that learns the co-occurrence of these infrequent words with more frequent words in order to predict them with higher accuracy. Training a separate model for a specific infrequent word seems difficult, while identifying the words with which this word co-occurs could be, instead, a good strategy.

5.3.2 Image representation

We investigate three types of visterms, as illustrated on Figure 5.3. The first is the DOG+SIFT combination presented in Chapter 3 (see Figure 5.3(a)). The second relies on large-scale image regions, combining both texture and color information (see Figure 5.3(c)). The third image representation is based on a larger number of smaller-scale image regions, uniformly extracted from a fixed grid (see Figure 5.3(d). They capture color or texture information respectively. The three discrete feature types are described in the following.

SIFT

The same visterm type as the one used in chapters 3 and 4 is used. An image d_i is represented by the histogram $s(d_i)$ of its constituting SIFT visterms (see Figure 5.3(b)):

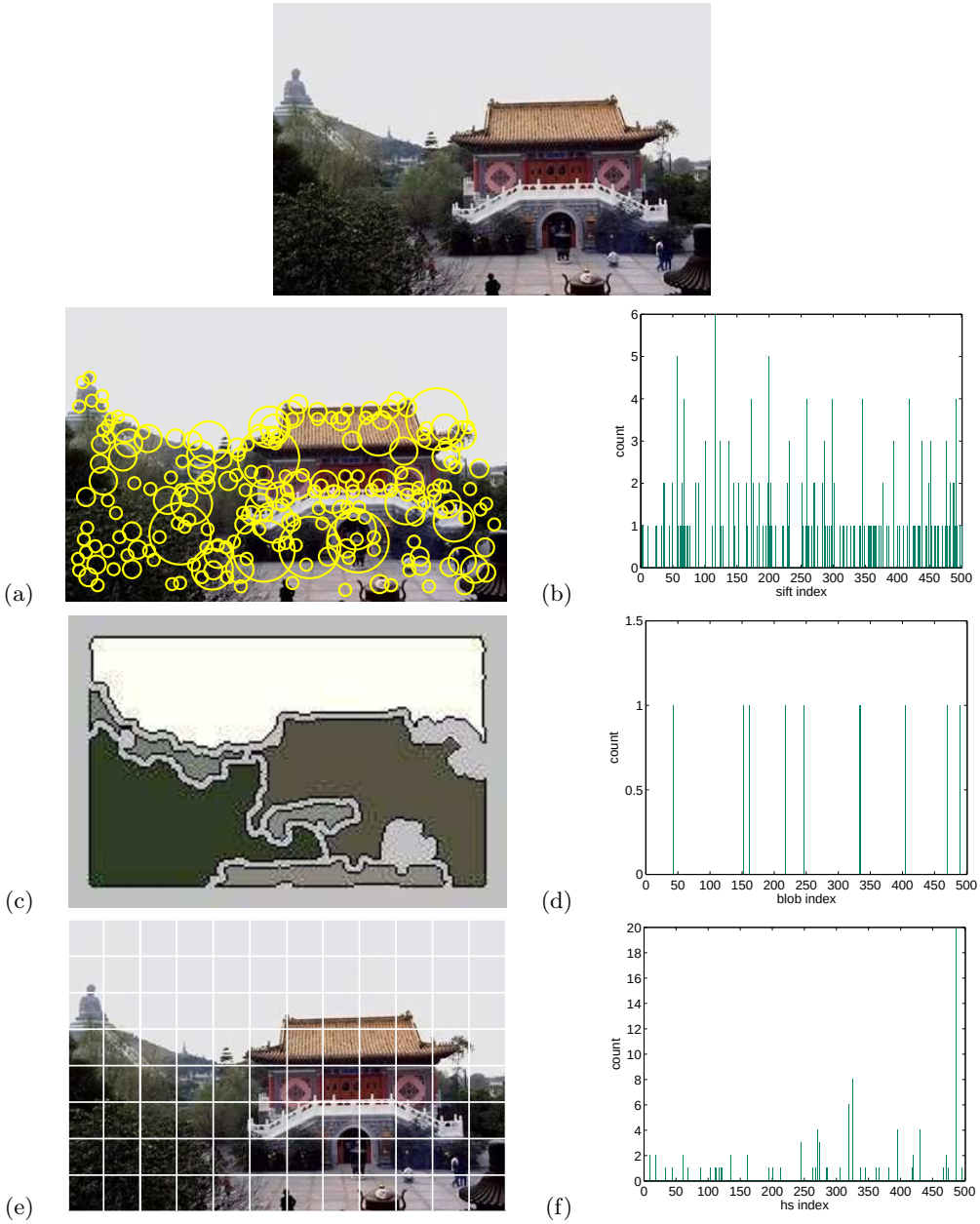


Figure 5.3: SIFT, Blobs, and HS image representations of the same image: (top) original image, (a) regions detected by the Difference-of-Gaussians (DoG) point detector, (b) resulting histogram of the quantized SIFT descriptors, (c) normalized cut image segmentation from which texture, color and shape features are extracted, (d) resulting histogram of the quantized image region features (Blobs), (e) uniform grid segmentation, color features are extracted, (f) resulting histogram of the quantized image region features (HS),

$$s(d_i) = \{n(d_i, s_1), ..., n(d_i, s_j), ..., n(d_i, s_{N_s})\}, \tag{5.2}$$

where $n(d_i, s_j)$ i the number of local descriptors in the image d_i that have been quantized into the visterm s_j. In the rest of the chapter, we refer to this representation as the SIFT representation.

Blobs

We consider an image representation originally proposed for region-based QBE [13], and later used for image annotation [5, 18, 6, 31]. A maximum of 10 regions per image, identified by the normalized cut segmentation algorithm [74], are represented by 36 features including color (18), texture (12), and shape/location (6). The K-means clustering algorithm is then applied to the region descriptors, quantizing them into a N_b-dimensional *Blob* representation. Note that the difference in the number of feature components makes the resulting Blob representation intrinsically biased towards color. An image d_i is segmented into a set of large image regions that are quantized and represented by the corresponding histogram $b(d_i)$ of size N_b (see Figure 5.3(d)):

$$b(d_i) = \{n(d_i, b_1), ..., n(d_i, b_j), ..., n(d_i, b_{N_b})\}, \tag{5.3}$$

where $n(d_i, b_j)$ denotes the number of regions in image d_i that are quantized into the Blob b_j. The motivation behind this representation is a possible match between the automatically segmented image regions and objects in the images. We see for instance on Figure 5.3 (a) that the green region matches trees in the original image, and that sky is covered by exactly one blob. As mentioned in [24], the match between the segmented regions and objects in the image is however relatively poor in general.

HS

No algorithm is currently available to automatically segment an image into meaningful parts. The use of a segmentation algorithm is therefore difficult to justify, and we decided to extract image regions from a uniform grid, as illustrated in Figure 5.3 (e). The pixel color distribution from the resulting regions is represented by a 2D Hue-Saturation histogram, where the color brightness value from the Hue-Saturation-Value (HSV) color-space is discarded for illumination invariance [63]. These HS histograms are then quantized into N_h bins with the K-means clustering algorithm, to obtain the corresponding histogram representation $h(d_i)$ of size N_h for the image d_i (see Figure 5.3(f)):

$$h(d_i) = \{n(d_i, h_1), ..., n(d_i, h_j), ..., n(d_i, h_{N_h})\}, \tag{5.4}$$

where $n(d_i, h_j)$ denotes the number of regions in image d_i that are quantized into the HS bin h_j. Contrarily to a global color histogram, $h(d_i)$ encodes the distribution of color information from local image regions. In the rest of the chapter, we refer to this representation of an image as the HS representation.

The SIFT, Blobs, and HS image representations encode different image properties, and are therefore expected to achieve different performances. The Blob representation is based on the joint quantization of shape, texture and color features, extracted from large image regions. The HS and SIFT representations are respectively based on the quantization of color or texture information, extracted from small-scale image regions. As we show on Figure 5.3, the number of regions that are considered in each case also varies: a maximum of 10 regions per image in the Blob case, 96 32×32 pixels square regions in the HS case, and an average of 240 detected points (depending on the image content) in the SIFT case. This makes the Blob histogram more sparse than the HS and SIFT histograms for an equivalent number of 500 K-means clusters, as shown in Figure 5.3 (b,d, and f). In section 5.6, we investigate the combination of these image representations. Using a direct concatenation of them in a first evaluation, the concatenation of the HS and SIFT features forms the complementary $v(d_i) = \{h(d_i), s(d_i)\}$ histogram of size $N_v = N_h + N_s$ for instance. To take advantage of these complementary source of visual information, the methods have to treat these unbalanced representations efficiently.

5.4 Modeling annotated images with PLSA

We discuss here three alternatives to learn a PLSA model for the co-occurrence of visual and textual features in annotated images. The first is a direct application of PLSA, described in Chapter 2, to the early integration of visual and textual modalities [51]. The two others are based on a variation of the PLSA EM algorithm that constrains the estimation of the conditional distributions of latent aspects given the training documents from one of the two modalities only. This allows to choose between the textual and the visual modality to estimate the mixture of aspects in a given document, what constrains the definition of the latent aspects on one or the other modality. The three learning procedures estimate the probability tables $P(v \mid z)$ and $P(w \mid z)$ from a set of training documents in different ways, allowing the annotation of a new image d_{new}.

5.4.1 PLSA-mixed

The PLSA-MIXED [51] model learns a standard PLSA model from a concatenated representation of the textual and the visual features $x(d) = \{w(d), v(d)\}$, as described in Algorithm 5.1. Using a training set of captioned images, $P(x \mid z)$ is learned for both textual and visual co-occurrences, capturing simultaneous occurrence of visual features and words given an aspect. The distribution over words given an aspect $P(w \mid z)$, and the distribution over visterms given an aspect $P(v \mid z)$ are then extracted from $P(x \mid z)$, and normalized such that $\sum_{j=1}^{N_w} P(w_j \mid z_k) = 1$ and $\sum_{j=1}^{N_v} P(v_j \mid z_k) = 1$.

Algorithm 5.1 Estimation of the $P(v \mid z)$ and $P(w \mid z)$ probability tables with PLSA-MIXED

random initialization of the $P(x \mid z)$, and $P(z \mid d)$ probability tables
while increase in the likelihood of validation data $\mathcal{L}(\mathcal{D}_{valid}) > T$ **do**
{E-step}
for all (d_i, x_j) pairs in training documents, and $k \in \{1, \ldots, L\}$ **do**

$$P(z_k \mid d_i, x_j) = \frac{P(x_j \mid z_k)P(z_k \mid d_i)}{\sum_{l=1}^{L} P(x_j \mid z_l)P(z_l \mid d_i))}$$

end for
{M-step}
for $j \in \{1, \ldots, N\}$ and $k \in \{1, \ldots, L\}$ **do**

$$P(x_j \mid z_k) = \frac{\sum_{i=1}^{M} n(d_i, x_j)P(z_k \mid d_i, x_j)}{\sum_{m=1}^{N}\sum_{i=1}^{M} n(d_i, x_m)P(z_k \mid d_i, x_m)}$$

end for
for $k \in \{1, \ldots, L\}$ and $i \in \{1, \ldots, M\}$ **do**

$$P(z_k \mid d_i) = \frac{\sum_{j=1}^{N} n(d_i, x_j)P(z_k \mid d_i, x_j)}{n(d_i)}$$

end for
estimate $P(z \mid d')$, where $d' \in \mathcal{D}_{valid}$ by folding-in, using $P(x \mid z)$
compute the folding-in likelihood of the validation data $\mathcal{L}(\mathcal{D}_{valid})$
end while
extraction of $P(w \mid z)$ and $P(v \mid z)$ from $P(x \mid z)$, such that $\sum_{j=1}^{N_w} P(w_j \mid z_k) = 1$, and $\sum_{j=1}^{N_v} P(v_j \mid z_k) = 1$

5.4.2 Asymmetric PLSA learning

We propose to model a set of annotated images with a PLSA model for which the conditional distributions over aspects $P(z \mid d_i)$ are estimated from one of the two modality only. Contrarily to PLSA-MIXED, which learns $P(z \mid d_i)$ from both the visual and the textual modalities, this formulation allows to treat each modality differently, giving more importance to the text captions or the image features in the latent space definition. We refer to this alternative learning algorithm as an *asymmetric PLSA learning*. Intrinsically, PLSA-MIXED assumes that the two modalities have an equivalent importance in defining the latent space, given that the latent space is learned from their concatenated representation. The only potential imbalance could result from a marked difference between the number of words and the number of visual features in the images, and these values are not freely controlled in practice.

An asymmetric PLSA learning gives a better control of the respective influence of each modality in the latent space definition. This concretely allows to model an image as a mixture of latent aspects that is either defined by its text captions or by its visual features, resulting in different aspect mixture weights. The aspect distributions $P(z \mid d_i)$ are learned for all training documents from one modality only (visual or textual modality), and are kept fixed for the other modality (textual or visual modality respectively). We refer to PLSA-FEATURES when the aspect distributions $P(z \mid d_i)$ are learned on the visual features, and to PLSA-WORDS when the aspect distributions are learned on the image captions. In the following, we describe how the parameters are learned in the asymmetric learning case.

Learning parameters

The description of the learning process is valid for the PLSA-FEATURES and the PLSA-WORDS approaches, but differs on which modality the multinomial distribution over aspects are learned for the training documents. The first and second modalities are therefore referred to as x^1 and x^2 respectively, and correspond either to the visual or to the textual features in the following. The PLSA-FEATURES and the PLSA-WORDS learning procedures are described in details in Algorithm 5.2 and Algorithm 5.3 respectively. They consist in two steps, one for each modality:

Estimate $P(x^1 \mid z_k)$ and $P(z \mid d_i)$: The first modality is used to estimate the L conditional distributions $P(x^1 \mid z_k)$ and the M conditional distributions $P(z \mid d_i)$, using a standard PLSA learning algorithm.

Estimate $P(x^2 \mid z_k)$: We consider that the aspect mixture weights learned from the first modality are correctly estimated for the training documents. The L conditional probability distributions $P(x^2 \mid z_k)$ for the second modality are therefore estimated by maximizing the likelihood of the training data, defined by the second modality, keeping the $P(z \mid d)$ probability table fixed. Note that this technique is computationally similar to the PLSA folding-in procedure to estimate the aspect mixture weights of an unseen document, introduced in Chapter 2. However, what we are trying to do is, conceptually speaking, very different.

The multinomial distributions $P(x^1 \mid z_k)$ and $P(x^2 \mid z_k)$ are defining the latent aspects z_k based on the visual and textual modalities respectively for PLSA-FEATURES: conversely for PLSA-WORDS. Early stopping is performed for each of the two learning steps described above, using an evaluation set $\mathcal{D}_{valid}$. The first step requires the estimation of the folding-in likelihood, but not the second step. The aspect mixture weights of the validation documents $P(z \mid d')$, estimated from the first step, are not re-estimated by folding-in.

5.4.3 Annotation by inference

Given new visual features $v(d_{new})$ and the previously estimated $P(v \mid z)$ parameters, the conditional probability distribution $P(z \mid d_{new})$ is inferred for a new image d_{new} using the standard folding-in

Algorithm 5.2 Estimation of the $P(v \mid z)$ and $P(w \mid z)$ probability tables with PLSA-FEATURES

random initialization of the $P(v \mid z)$ and $P(z \mid d)$ probability tables

while increase in the likelihood of validation data $\mathcal{L}_{valid} > T$ **do**
{E-step}
for all (d_i, v_j) pairs in training documents, and $k \in \{1, \ldots, L\}$ **do**

$$P(z_k \mid d_i, v_j) = \frac{P(v_j \mid z_k)P(z_k \mid d_i)}{\sum_{l=1}^{L} P(v_j \mid z_l)P(z_l \mid d_i))}$$

end for
{M-step}
for $j \in \{1, \ldots, N_v\}$ and $k \in \{1, \ldots, L\}$ **do**

$$P(v_j \mid z_k) = \frac{\sum_{i=1}^{M} n(d_i, v_j)P(z_k \mid d_i, v_j)}{\sum_{m=1}^{N_v} \sum_{i=1}^{M} n(d_i, v_m)P(z_k \mid d_i, v_m)}$$

end for
for $k \in \{1, \ldots, L\}$ and $i \in \{1, \ldots, M\}$ **do**

$$P(z_k \mid d_i) = \frac{\sum_{j=1}^{N_v} n(d_i, x_j)P(z_k \mid d_i, x_j)}{n(d_i)}$$

end for

estimate $P(z \mid d')$, where $d' \in \mathcal{D}_{valid}$ by folding-in, using $P(v \mid z)$
compute the folding-in likelihood of the validation data $\mathcal{L}(\mathcal{D}_{valid})$

end while

random initialization of the $P(w \mid z)$ probability table
while increase in the likelihood of validation data $\mathcal{L}_{valid} > T$ **do**
{E-step}
for all (d_i, w_j) pairs in training documents, and $k \in \{1, \ldots, L\}$ **do**

$$P(z_k \mid d_i, w_j) = \frac{P(w_j \mid z_k)P(z_k \mid d_i)}{\sum_{l=1}^{L} P(w_j \mid z_l)P(z_l \mid d_i))}$$

end for
{Partial M-step}
for $j \in \{1, \ldots, N_w\}$ and $k \in \{1, \ldots, L\}$ **do**

$$P(w_j \mid z_k) = \frac{\sum_{i=1}^{M} n(d_i, w_j)P(z_k \mid d_i, w_j)}{\sum_{m=1}^{N_w} \sum_{i=1}^{M} n(d_i, w_m)P(z_k \mid d_i, w_m)}$$

end for

compute the likelihood of the validation data $\mathcal{L}(\mathcal{D}_{valid})$ from $P(w \mid z)$ and $P(z \mid d')$ from previous modality

end while

Algorithm 5.3 Estimation of the $P(v \mid z)$ and $P(w \mid z)$ probability tables with PLSA-WORDS

random initialization of the $P(w \mid z)$ and $P(z \mid d)$ probability tables

while increase in the likelihood of validation data $\mathcal{L}_{valid} > T$ **do**
{E-step}
for all (d_i, w_j) pairs in training documents, and $k \in \{1, \ldots, L\}$ **do**

$$P(z_k \mid d_i, w_j) = \frac{P(w_j \mid z_k)P(z_k \mid d_i)}{\sum_{l=1}^{L} P(w_j \mid z_l)P(z_l \mid d_i))}$$

end for
{M-step}
for $j \in \{1, \ldots, N_w\}$ and $k \in \{1, \ldots, L\}$ **do**

$$P(w_j \mid z_k) = \frac{\sum_{i=1}^{M} n(d_i, w_j)P(z_k \mid d_i, w_j)}{\sum_{m=1}^{N_w} \sum_{i=1}^{M} n(d_i, w_m)P(z_k \mid d_i, w_m)}$$

end for
for $k \in \{1, \ldots, L\}$ and $i \in \{1, \ldots, M\}$ **do**

$$P(z_k \mid d_i) = \frac{\sum_{j=1}^{N_w} n(d_i, x_j)P(z_k \mid d_i, x_j)}{n(d_i)}$$

end for

estimate $P(z \mid d')$, where $d' \in \mathcal{D}_{valid}$ by folding-in, using $P(w \mid z)$
compute the folding-in likelihood of the validation data $\mathcal{L}(\mathcal{D}_{valid})$

end while

random initialization of the $P(v \mid z)$ probability table
while increase in the likelihood of validation data $\mathcal{L}_{valid} > T$ **do**
{E-step}
for all (d_i, v_j) pairs in training documents, and $k \in \{1, \ldots, L\}$ **do**

$$P(z_k \mid d_i, v_j) = \frac{P(v_j \mid z_k)P(z_k \mid d_i)}{\sum_{l=1}^{L} P(v_j \mid z_l)P(z_l \mid d_i))}$$

end for
{Partial M-step}
for $j \in \{1, \ldots, N_v\}$ and $k \in \{1, \ldots, L\}$ **do**

$$P(v_j \mid z_k) = \frac{\sum_{i=1}^{M} n(d_i, v_j)P(z_k \mid d_i, v_j)}{\sum_{m=1}^{N} \sum_{i=1}^{M} n(d_i, v_m)P(z_k \mid d_i, v_m)}$$

end for

compute the likelihood of the validation data $\mathcal{L}(\mathcal{D}_{valid})$ from $P(v \mid z)$ and $P(z \mid d')$ from previous modality

end while

procedure for a new document (Chapter 2). Given these estimated mixture weights, the conditional distribution over words given this new image is given by:

$$P(w \mid d_{new}) = \sum_{k}^{L} P(w \mid z_k) P(z_k \mid d_{new}), \tag{5.5}$$

where the probability table $P(w \mid z)$ was estimated from the training data.

5.5 Baseline methods

Three baseline models for image annotation are considered for comparison with our models. The first baseline consists in a visual comparison between the image to annotate and the training images, propagating their annotations based on this similarity. The two other methods correspond to the state-of-the-art performance in image annotation when the discrete, quantized Blob representation $b(d)$ is used [18, 31, 66, 51].

5.5.1 Annotation propagation

Intuitively, training images that are similar to a new image d_{new} should be taken into account to generate its annotation. Our simplest baseline therefore consists in computing the similarity between the image d_{new} and the training images, sequentially attaching their respective annotation to d_{new} based on these similarities. Concretely, we compute the cosine similarity between the image d_{new} and the M training images d_i based on their respective visual representations $v(d_{new})$ and $v(d_i)$:

$$sim_{cos}(v(d_{new}), v(d_i)) = \frac{\sum_{j}^{N_v} n(d_{new}, v_j) n(d_i, v_j)}{\sqrt{\sum_{j}^{N_v} n(d_{new}, v_j)^2} \sqrt{\sum_{j}^{N_v} n(d_i, v_j)^2}} \tag{5.6}$$

The training images are ranked with respect to this similarity measure, and the probability of a word w_i given d_{new} is estimated by the inverse of the best ranked image according to Equation 5.6 that contains the word w_i:

$$P(w_i \mid d_{new}) \propto (rank(d_{best}))^{-1}, \tag{5.7}$$

where d_{best} is the most similar image to d_{new} in the training set that contains the word w_i. and $rank(d_{best})$ is the rank order of this image given d_{new}. The word probabilities are then normalized so that $\sum_{N_w} P(w \mid d_{new}) = 1$.

5.5.2 Cross-media relevance model

In [31], the annotation of an unseen image d_{new} is based on the joint probability of all its m constituting visual elements v_l and the word w_j. This joint probability is estimated by its expectation over the M training images,

$$P(w_j, v_1, ..., v_m) = \sum_{i}^{M} P(d_i) P(w_j, v_1, ..., v_m \mid d_i) \tag{5.8}$$

The visual elements are considered independent given an image d_i, which gives:

$$P(w_j, v_1, ..., v_m) = \sum_{i}^{M} P(d_i) P(w_j \mid d_i) \prod_{l}^{N_v} P(v_l \mid d_i)^{n(d_i, v_l)}, \tag{5.9}$$

where $n(d_i, v_l)$ is the count of the visual element v_l in the image d_i. The probability of a word w in a training image d_i is the likelihood of this word in this image combined with the likelihood of this word in all the training images. A fusion parameter α controls the importance of the image and the training set likelihoods:

$$P(w_j \mid d_i) = (1-\alpha)\frac{n(d_i, w_j)}{\sum_l^{N_v} n(d_i, v_l) + \sum_j^{N_w} n(d_i, w_j)} + \alpha\frac{n(w_j, d)}{M}, \tag{5.10}$$

where $n(d_i, w_j)$ denotes the count of the word w_j in the image d_i, $n(d_i, v_l)$ is the count of the visual element v_l in the image d_i, $n(w_j, d)$ is the number of images in which the word w_j appears, and M is the number of training images. Similarly, the probability of a visual element given an image d_i is estimated by its likelihood in this image smoothed by its likelihood in the training set, controlled by a parameter β:

$$P(v_l \mid d_i) = (1-\beta)\frac{n(d_i, v_l)}{\sum_l^{N_v} n(d_i, v_l) + \sum_j^{N_w} n(d_i, w_j)} + \beta\frac{n(v_l, d)}{M}, \tag{5.11}$$

where $n(d_i, v_l)$ denotes the count of the visual element v_l in the image d_i, $n(d_i, w_j)$ denotes the count of the word w_j in the image d_i, $n(v_l, d)$ is the number of images in which the word v_l appears, and M is the number of training images. The hyper-parameters α and β are estimated on a validation set to optimize the model performance.

5.5.3 Cross-media translation table

In [66], a translation table T_{cos} between words and quantized visual features is proposed. The word-by-image matrix is weighted with the *tf-idf* scheme to obtain the weighted matrix D_w:

$$D_w = (n(d_i, w_j) * log(\frac{M}{n(w_j, d)}))_{M \times N_w}, \tag{5.12}$$

where $n(d_i, w_j)$ is the count of the word w_j in the image d_i, $n(w_j, d)$ is the number of documents the word w_j appears in, M is the number of training images, and N_w is the size of the vocabulary. Similarly, the feature-by-image matrix is weighted with the *tf-idf* scheme to obtain the weighted matrix D_v:

$$D_v = (n(d_i, v_l) * log(\frac{M}{n(v_l, d)}))_{M \times N_v}, \tag{5.13}$$

where $n(d_i, v_l)$ is the count of the word v_l in the image d_i, $n(v_l, d)$ is the number of documents the visual element v_j appears in, M is the number of training images, and N_v is the size of the visual feature space. A Singular Value Decomposition (SVD) is applied on the D_w and D_v matrices, keeping the first r eigenvalues which preserve 90% of the variance to suppress the noise in the data. Let the j-th column of the matrix D_w be d_{wj}, and the l-th column of the matrix D_v be d_{vl}. The cross-media translation table T is defined by:

$$T_{cos} = (sim_{cos}(d_{wj}, d_{vl}))_{N_w \times N_v}, \tag{5.14}$$

where the cosine similarity function $sim_{cos}()$ is defined in Equation 5.6. Normalizing T_{cos} by column, the annotation of a new image d_{new} represented by its histogram $v(d_{new})$ is given by:

$$P(w \mid d_{new}) = T_{cos} \times v(d_{new}). \tag{5.15}$$

5.6 Results

5.6.1 Data

As shown in [58] for the case of Query by example (QBE), contradictory rankings can be obtained if the performance evaluation is conducted on different data subsets, even if these subsets are created from the same original image collection. To prevent this possible inaccuracy, it is crucial to compare different systems on identical data, with clearly defined training and testing sets. We conduct our experiments on an annotated image dataset that was originally used in [5], and consists in ten samples of roughly 16000 annotated images. Each sample is split into a training and a testing set, with an average number of 5240 training and 1750 testing images. The average vocabulary size is 161. The Blob representation, as well as the description of the different samples were downloaded from `http://kobus.ca/research/data/jmlr_2003/`.

5.6.2 Mean average precision measure

A number of papers [18, 66, 24, 32, 31] measure the ability of the system to produce a human-like annotation, selecting a small number of words from the vocabulary. A fixed threshold or a fixed number of words has to be decided to extract short captions that can be used for image retrieval. With this, for a given query word, the number of correctly retrieved images divided by the number of retrieved images is the *word precision*, and the number of correctly retrieved images divided by the total number of correct images is the *word recall*. The average word precision and word recall values summarize the system performance.

One drawback of creating a human-like annotation is that only a fraction of words from the vocabulary are eventually predicted for the test images, because uncommon words tend not to be predicted due to a very low conditional probability. The word precision and recall values have thus to be presented together with the number of predictable words, as done in [18, 24, 32, 31], which makes the comparison between models unclear. Is it better for a system to predict only a few words with a high accuracy, or is a system more efficient if it can predict more words?

However, given that the goal is to index images for image retrieval, there is no need to produce such short, human-like annotation. The conditional probability distribution $P(w \mid d_{new})$ can be used to rank the images for all possible queries. Even if the conditional probabilities of a word are low for the images to rank, the comparison of the relative values allows to rank the image collection for each word query. To illustrate this, the truncated word distribution inferred on two images using the PLSA-WORDS model are shown in Figure 5.4. The word *flowers* is in the top 20 words for both images, but the probability of the word *flowers* given the top image is higher than given the bottom image. This information would be discarded if the model is used to predict a fixed length annotation, although it can be exploited for image ranking. The distribution over word in Figure 5.4 also shows how much more probable the word *ocean* is given the bottom image than given the top-image. This, again, would not be possible if we were only relying on a five-word annotation.

The performance measure used in this work is mean average precision (mAP). This is a standard measure for the retrieval of text documents for years, that has also been used by TRECVID to evaluate the semantic concept video retrieval task for several years (details can be found at http://www-nlpir.nist.gov/projects/trecvid/). mAP has the ability to summarize the performance in a meaningful way. To compute it, the average precision (AP) of a query q is first defined as the sum of the precisions of the correctly retrieved words at rank i, divided by the total number of relevant images $rel(q)$ for this query:

$$AP(q) = \frac{\sum_{i \in relevant} precision(i)}{rel(q)}. \tag{5.16}$$

The average precision measure of a query is thus sensitive to the entire ranking of documents. The mean of the average precision of M_q queries q summarizes the performance of a retrieval system in

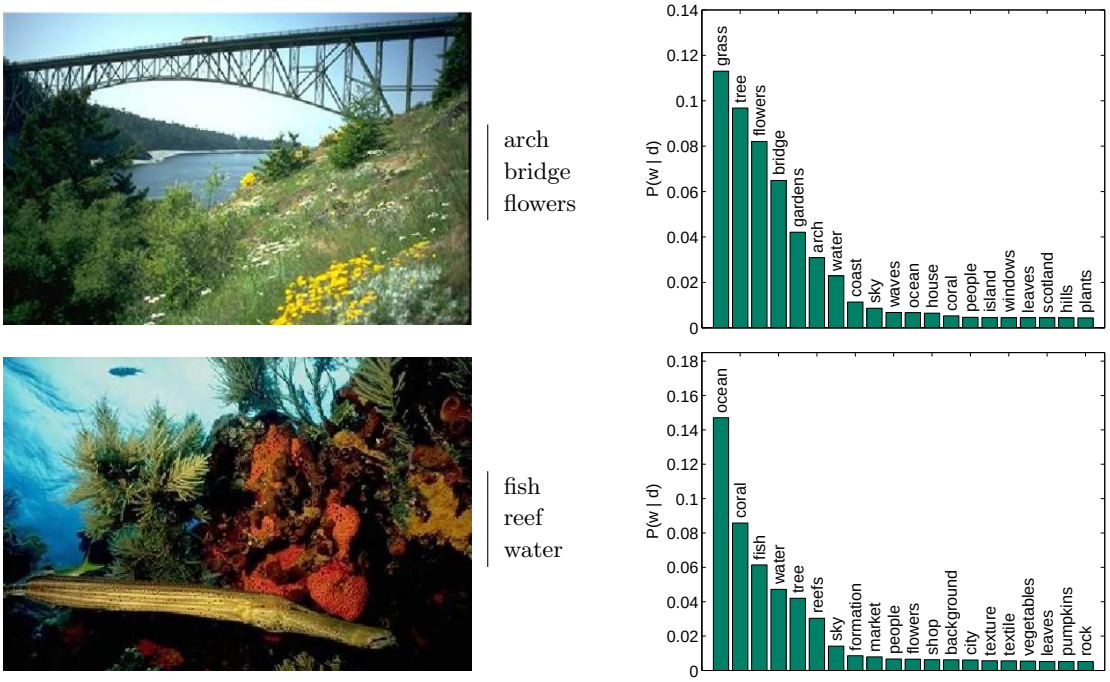


Figure 5.4: The conditional probability distribution $P(w \mid d)$ inferred on two test images from their HS+SIFT representation with the *PLSA-words* approach. The image and the ground truth annotation are shown on the left column, and the top twenty words and their conditional probability are shown on the right column.

one mean Average Precision (mAP) value:

$$mAP = \frac{\sum_{M_q} AP(q)}{M_q} \tag{5.17}$$

5.6.3 Hyper-parameters and cross-validation

We need to estimate two types of hyper-parameters by cross-validation. The first is the number of K-means clusters that defines the quantization of the visual features into visterms, the second is the number of latent aspects for the approaches based on a PLSA model. The number of K-means clusters is cross-validated for the HS, SIFT and HS+SIFT representations, for 100, 200, 500, and 1000 clusters. The value of $N_b = 500$ clusters for the Blob representation is kept fixed, as this representation

	Blobs	HS				SIFT			
	500	100	200	500	1000	100	200	500	1000
propagation	10.2 (0.6)	10.7 (0.7)	10.8 (0.6)	11.7 (0.7)	12.4 (1.0)	10.7 (0.9)	11.4 (0.6)	12.2 (0.8)	13.0 (0.8)
CMRM [31]	12.1 (0.8)	13.4 (1.0)	14.2 (0.9)	14.4 (1.1)	14.5 (1.2)	11.6 (0.9)	12.7 (1.0)	12.3 (1.6)	10.0 (2.0)
SVD-cos [66]	15.6 (0.7)	14.1 (1.0)	15.4 (1.0)	16.4 (1.1)	17.1 (1.1)	10.0 (0.8)	11.6 (0.8)	12.8 (0.9)	14.3 (0.9)

Table 5.1: Average mAP values (%) over 10 cross-validation runs for different quantization of the HS and the SIFT image representations, for the three baseline methods. The standard deviation is given in parentheses.

is provided as is by the authors of [5]. The mAP performance of the Blob representation is given for comparison. The K-means models are learned on the training images of each sample set. On Table 5.1, we show the mAP values obtained with the three baseline methods, averaged over ten cross-validation runs for one sample set. The hyper-parameter values estimated by cross-validation from this sample set will be used for the remaining 9, as one set is assumed to be representative of the entire set. For the three baseline methods, the best number of K-means clusters for both HS and SIFT representations is 1000, except for the SIFT representation in the CMRM case, for which 200 clusters corresponds to the best retrieval performance. We also observe that the HS representation consistently achieves higher performance than the Blob representation for the same number of clusters. We use these estimated number of clusters for the remaining experiments.

	HS+SIFT			
	500-500	500-1000	1000-500	1000-1000
propagation	16.0 (1.3)	15.5 (1.4)	16.8 (1.2)	16.4 (1.3)
CMRM [31]	17.6 (0.8)	17.4 (0.8)	6.2 (0.8)	4.8 (0.8)
SVD-cos [66]	19.9 (1.5)	20.9 (1.7)	20.2 (1.7)	21.2 (1.7)

Table 5.2: Average mAP values (%) over 10 cross-validation runs for representations based on the concatenation of different quantization of the HS and SIFT features, for the three baseline methods. The standard deviation is given in parentheses.

We also estimated the number of clusters by cross-validation for the HS+SIFT concatenation, as reported on Table 5.2. We restricted our analysis to the combination of HS and SIFT features for two reasons. First, as Table 5.1 suggests, the HS representation outperforms the Blob representation. Second, HS and SIFT features result from the quantization of local color-only and local texture-only information, respectively, while the Blob representation corresponds to the joint quantization of color, texture and shape. Analyzing the effect of the combination of separately extracted color-only and texture-only information seems more intuitive than analyzing the combination of a texture-based representation with a joint color-texture-shape representation. The values from Table 5.2 show that the optimal combination for the propagation method is $N_h = 1000$ HS and $N_s = 500$ SIFT clusters, 500 HS and SIFT clusters for the CMRM case, and 1000 HS and SIFT clusters for the SVD-COS case. The results from the CMRM method drop significantly for the $(1000, 500)$ and $(1000, 1000)$ combination, although we carefully selected the α and β parameters.

As we have mentioned, PLSA-based approaches require the number of latent aspects L to be estimated, as this hyper-parameter defines the capacity of the model: the number of parameters $P = (M(L-1)) + (L(N-1))) \sim L(M+N)$ linearly depends on L. The best value for the number of clusters therefore needs to be jointly estimated with the number of latent aspects for the three PLSA-based approaches, what is reported on Figure 5.5. The average of the mAP values computed for 10 cross-validation runs are reported on Figure 5.5, where the number of latent aspects is varied between 10 to 250, for the three PLSA-based approaches. The number of K-means clusters for quantizing the visual features is also varied, and reported as a different line on each plot. The standard deviation over 10 cross-validation runs is shown with error bars.

The plots on Figure 5.5 allow to decide the number of aspects and the number of clusters given each PLSA learning methods and each image representation. The maximum number of K-means clusters seems to be a reasonable choice for the HS, SIFT and HS+SIFT representations. The results in the following sections are therefore computed with $N_h = 1000$ and $N_s = 1000$ clusters for the quantization of the HS and SIFT representations. Regarding the number of aspects L, the following values are chosen:

- PLSA-MIXED : $L = 140$ for HS, $L = 110$ for SIFT, and $L = 170$ for HS+SIFT,
- PLSA-FEATURES: $L = 170$ for HS, $L = 150$ for SIFT, and $L = 180$ for HS+SIFT,
- PLSA-WORDS : $L = 120$ for HS, $L = 110$ for SIFT, and $L = 120$ for HS+SIFT.

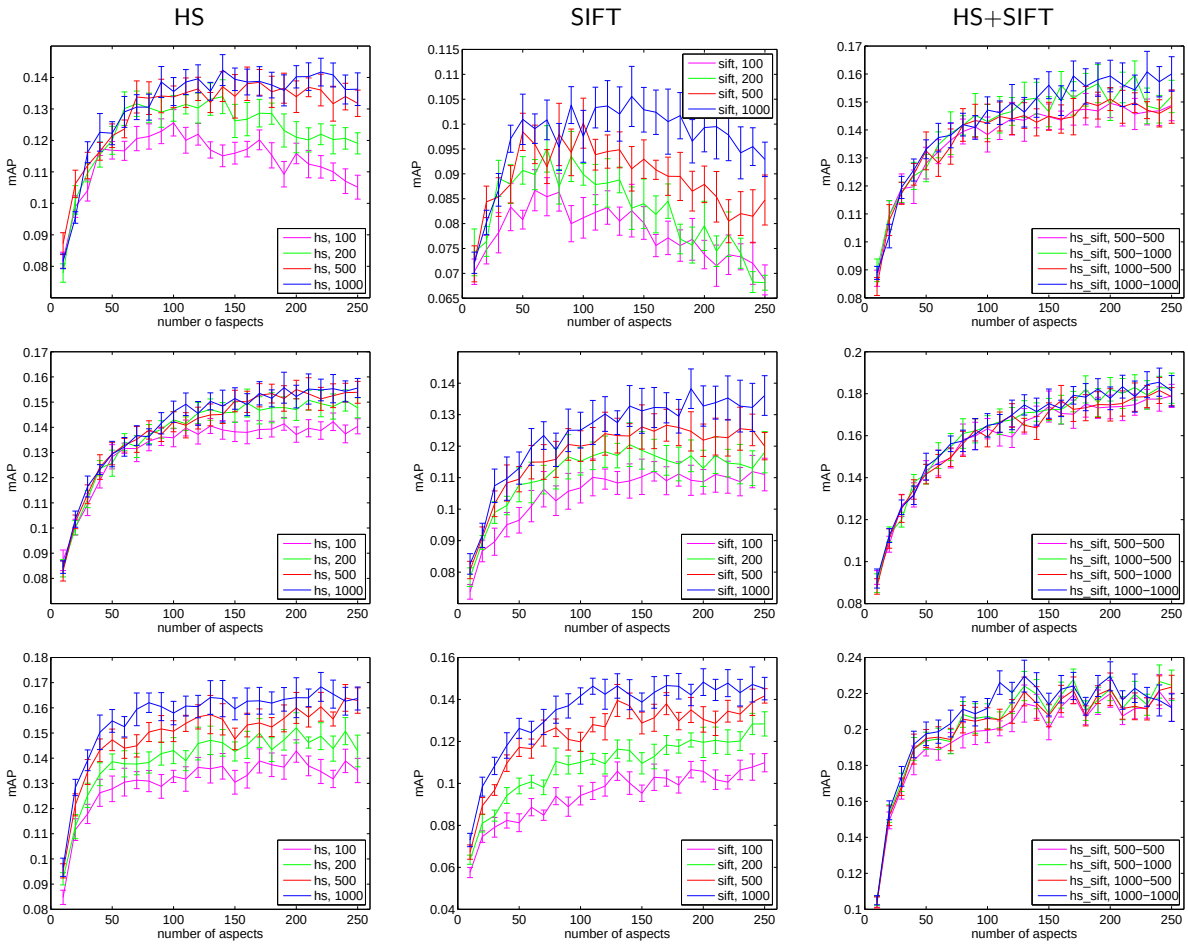


Figure 5.5: Joint cross-validation of the number of aspects and the number of K-means clusters for the the HS (left column), SIFT (middle column), and HS+SIFT (right column) representations, and for the three PLSA learning methods. The mAP value obtained for PLSA-MIXED (top row), PLSA-FEATURES (middle row), and PLSA-WORDS (bottom row) are given. The error bars show the standard deviation of the mAP values for ten runs.

5.6.4 Overall performance

The average of the mAP obtained on the 10 test sets with the hyper-parameters estimated in Section 5.6.3 are shown in Table 5.3, where the performance of the Blob, HS, SIFT and HS+SIFT representations for the six auto-annotation methods presented in Section 5.4 and 5.5 are reported. The standard deviation of the mAP over the ten test sets is shown in parentheses. Note that the mAP values in Table 5.3 are consistently lower than the cross-validation values, because the retrieval tasks on which the mAP are computed is more challenging: an average of 1750 images for test vs. an average of 520 images for cross-validation are ranked.

	Blobs	HS	SIFT	HS+SIFT
propagation	7.8 (0.7)	9.0 (0.2)	9.4 (1.0)	13.1 (0.5)
CMRM [31]	11.5 (1.1)	10.7 (1.1)	7.9 (0.5)	13.4 (1.0)
SVD-COS [66]	12.9 (1.1)	12.9 (0.8)	10.7 (0.7)	16.6 (1.1)
PLSA-MIXED	5.8 (0.8)	10.2 (0.8)	7.5 (0.6)	11.9 (1.3)
PLSA-FEATURES	8.2 (0.7)	11.2 (1.0)	10.1 (0.8)	14.0 (1.3)
PLSA-WORDS	11.0 (0.9)	13.3 (1.0)	11.8 (1.1)	19.1 (1.2)

Table 5.3: Average mAP values (%) over the 10 test sets, for the six methods when combinations of HS and SIFT features are used.

We see that the the PLSA-MIXED approach particularly fails to produce an efficient probabilistic indexing of the test images for all the image representations. In particular, its performance is lower than the simple propagation baseline that relies on a direct image similarity computation. Using a concatenated representation of words and visual features, PLSA-MIXED attempts to simultaneously model the visual and textual modalities. As we already mentioned, this means that intrinsically, PLSA-MIXED assumes that the two modalities have an equivalent importance in defining the latent space, which as the results suggest, is not the most accurate assumption.

Except for the PLSA-MIXED case and the CMRM method when the SIFT representation is used, all methods achieve a higher performance than the propagation baseline. This shows that computing image similarity, although simple and intuitive, can only be considered as a low quality baseline for image annotation. It is however rather competitive with the CMRM and PLSA-FEATURES methods, in particular for the HS and HS+SIFT image representations.

All methods take advantage of the HS+SIFT combination: the performance of a single feature type is always lower than their combination, which confirms that HS and SIFT features encode complementary information. It is interesting to notice that the CMRM and SVD-COS methods achieve the best performance for the Blob representation, which is the representation they were originally evaluated on [66, 31]. These methods however do not produce the best performance, especially when compared to the PLSA-WORDS method. Furthermore, when the conditional probability distributions of the aspects given the training documents d_i $P(z \mid d_i)$ are learned from the visual features with PLSA-FEATURES, the estimation of the conditional distribution over words gives better results than PLSA-MIXED, but also lower mean average precision values than the baseline methods.

Regarding PLSA-WORDS, our method achieves a similar mAP performance than the SVD-COS method for the Blob representation, but it exploits the HS, SIFT, and the HS+SIFT representations more efficiently than both the CMRM and the SVD-COS approaches. Furthermore, it consistently performs better than CMRM. The PLSA-WORDS model achieves the best mAP score overall when the concatenated SIFT and HS+SIFT representations are used. In the HS+SIFT case, the PLSA-WORDS improves over the SVD-COS method by 15% (relative improvement). This improvement is significant according to a paired samples T-test with a p-value of 0.05, showing that the estimation of the aspect distribution based on the textual modality improves over the linear algebra-based SVD-COS method and over the method that does not use aspect variables.

In the following two sections, we analyze the performance of PLSA-WORDS, the best-performing model, in more details.

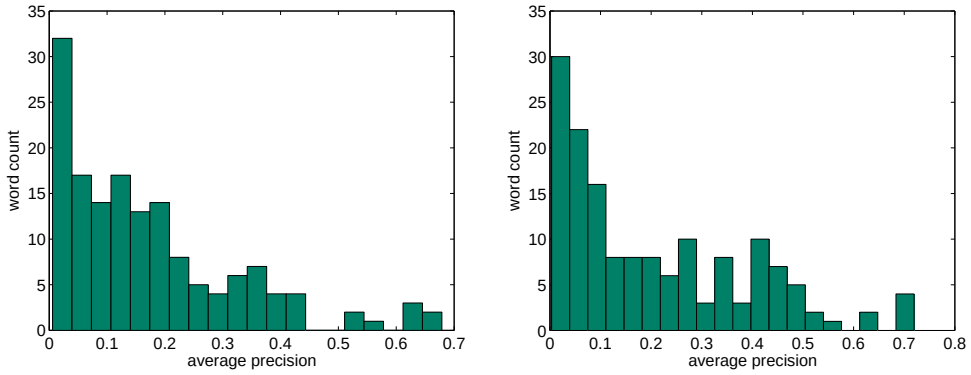


Figure 5.6: Histogram of the 153 average precision values for SVD-COS (left) and PLSA-WORDS (right) methods.

5.6.5 Per-word performance

The histogram of the average precision values obtained with PLSA-WORDS in Figure 5.6 (right) shows a marked difference in performance for different words: half of the words have an average precision value higher than 0.14, 65 words have an average precision value below 0.1 and 10 words have an average precision value above 0.5. A similar trend can be observed with the baseline methods, as shown for the SVD-COS method on Figure 5.6 (left). This important variation goes unnoticed if only the mean average precision is reported, as done in part of the existing literature [35, 24].

The combined effect of three factors could explain why the system does not rank images satisfactorily for some words while achieving a good performance for others. First, the number of training images per word ranges significantly in the dataset, from 21 (for *bay*, *candy*, *formula*, ...) to 1124 (*water*), and obviously the quality of a statistical model depends on the nature and the number of training examples. Second, all words have to be learned from the same set of visual features, which can be better suited for some concepts than for others. Third, the co-occurrence in text captions can have a combined influence with the two previous points; if a given word is correctly learned by the model because it is well represented by the visual features and has a sufficient number of training examples, other words that consistently co-occur with it could have a relatively high performance despite a low number of training examples. We investigate these three factors by analyzing individual word performance together with basic statistics computed on the training set.

The number of training images and the average precision for the 20 words with the best and the worst performance with the PLSA-WORDS model are shown in Figure 5.7, which shows that there is a difference in the average number of examples for the 20 best performing words compared to the 20 worst performing words. The former have 106 training examples on average, while the latter have an average of 29 examples. This fact suggests that the number of examples does indeed influence the performance of a word in general, because a low number of examples often does not allow to capture the statistical variations of a word appearance.

However, we also see in Figure 5.7 that, even though words have a comparable number of training examples, their respective performance is completely different. The words *polar*, *formula*, and *black* (Figure 5.7a) have a high average precision value (~ 0.5), while the words *river*, *woods*, and *road* (Figure 5.7b) are part of the 20 words with the lowest average precision (~ 0.015). The performance of a given word thus not only depends on the number of training examples, but also on the two other factors mentioned above.

In cases when the images a word is attached to depict consistent visual content that is well represented by the feature set, the model can learn the representation from little training data. For instance, images that are annotated with the word *formula* (see Figure 5.8 left) contain distinctive

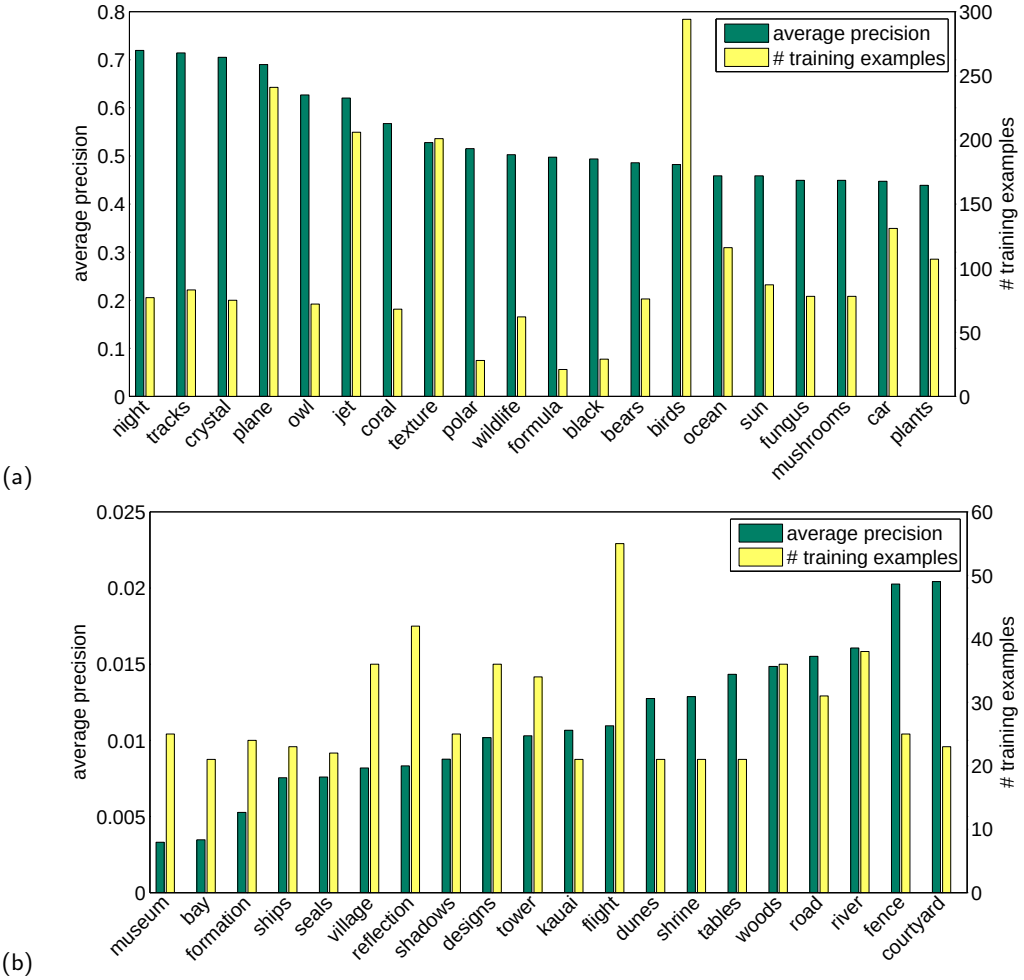


Figure 5.7: Average precision and number of training examples for (a) the twenty best, and (b) the twenty worst average precision values (ranked in decreasing order in (a), and in increasing order in (b)) obtained with the PLSA-WORDS annotation.

visual features that can be captured from a relatively small number of examples (21 in the dataset), while providing an high average precision value of 0.5 for this word. Similarly, the word *polar* is mainly attached to winter images (see Figure 5.8 right) which have a very distinctive white aspect, and is therefore well predicted (average precision of 0.51) despite very few training examples (only 28). On the contrary, the words *reflection* and *museum* for instance are not correctly modeled because the corresponding image content can not be learned properly from 25 and 42 examples, respectively.

For models such as PLSA-WORDS that learn co-occurrences in image captions, there is a possibility to improve the prediction of infrequent words from their co-occurring words. We show three examples of this effect on Figure 5.9, for the words *skis*, *bridge* and *leaves*. For these three words, the four words that co-occur the most with each of them are reported, as well as different statistics, including the number of times they co-occur with the word considered (top row), the number of times they appear in the training set (middle row), and their respective average precision (bottom row. Regarding the first example, although the word *skis* is only represented by 63 examples, the fact that it co-occurs quite often with more frequent words like *people* (which appears in 853 examples), *snow* (252 examples), and *mountain* (82 examples), allows PLSA-WORDS to predict *skis* with a high average precision (0.3).

Figure 5.8: Two examples of images annotated with the word *formula* (left), and two images annotated with the word *polar* (right).

The method SVD-cos also takes advantage of this co-occurrence, but predicts the word *skis* with a lower average precision (0.24). For the second example, the word *bridge* only has 93 examples, but is well predicted by the PLSA-WORDS model, because it co-occurs with words that have more examples in the training set, like *water* (which occurs in 1124 examples), *sky* (949 examples), and *stone* (258 examples). For the last example, the word *leaves* is predicted with an average precision of 0.43 by PLSA-WORDS, although there are only 134 *leaves* image examples. The fact that the word *leaves* co-occurs quite frequently with the words *flowers* (appearing in 224 examples), or *tree* (929 examples) also illustrates why a model that captures co-occurrence information at the caption level performs better than a model that does not model this information explicitly. In the two last examples, *SVD-cos* fails to take advantage of the co-occurrence with more frequent words, as *PLSA-words* does.

5.6.6 Combination of features

To observe in more detail the benefit of combining HS and SIFT features for PLSA-WORDS, their individual and combined effects on the average precision of 10 representative words is shown in Figure 5.10. These 10 words are selected to illustrate different interesting behaviors that are observed when SIFT (dark green), HS (green) or both (yellow) are used.

As a general trend, we see that words that are rather well defined by color regions have higher average precision values when the HS representation is used, compared to SIFT visterms. In Figure 5.10, images annotated with words such as *sun*, *crystal*, *plane*, and *night*, depict colored regions, and are therefore well represented by the HS features. As shown in the Figure 5.10, the average precision of these words for a retrieval system based on the HS representation outperforms the same system based on the SIFT representation. This is a somewhat expected result. For instance, images annotated by the word *sun* present rather non-distinctive image structures, but contain very specific colors. Similarly, *crystal* images have a large variety of textures but present very distinctive colors. The average precision of this word is therefore higher when HS features are used. The word *plane*

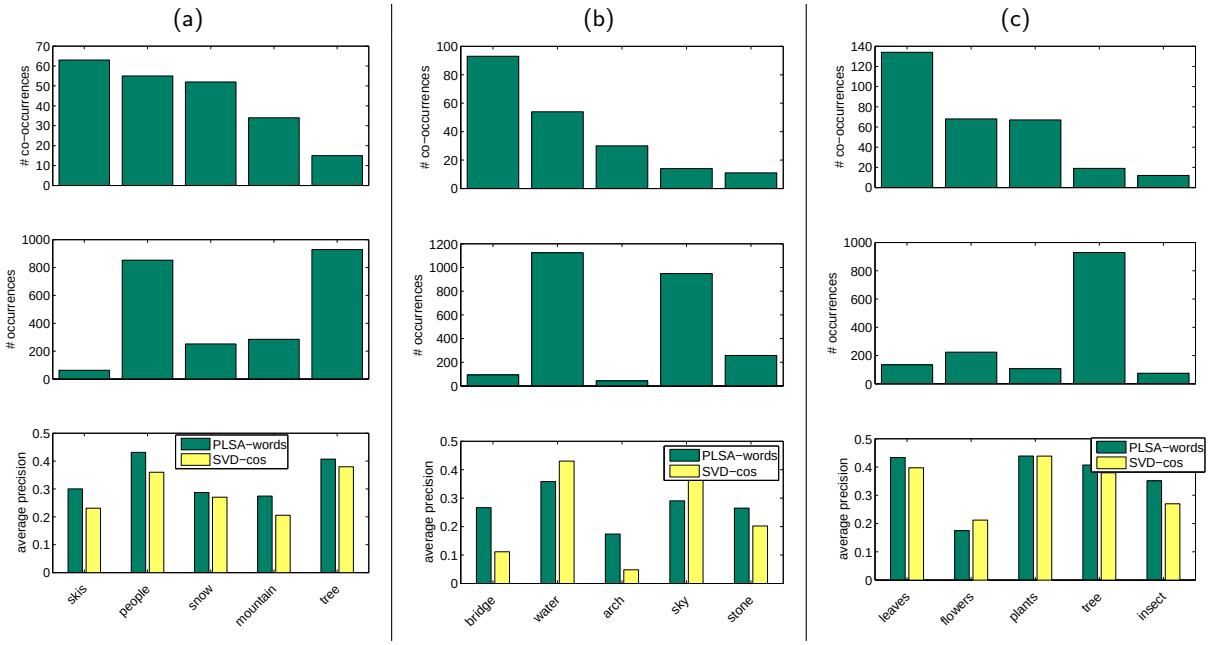


Figure 5.9: Effect of word co-occurrences in captions for the words (a) *skis*, (b) *bridge*, and (c) *leaves*. The first row shows the number of times the four most frequently co-occurring words appear in the same caption as the word considered, the second row shows the total number of times each word appears in the training set, and the third row shows the average precision of these words for the PLSA-WORDS (green) and the SVD-cos (yellow) models.

also happens to be better represented by HS features as shown in Figure 5.10, which could be at first glance counter-intuitive. However, the word *plane* consistently appears in the context of blue sky, which are well identified by the HS representation.

On the contrary, if a word corresponds to images that contain specific textures, the SIFT representation becomes more informative and results in better image ranking. This can be observed in Figure 5.10, where the average precision values for the words *buildings*, *clouds*, and *house*, are higher when the SIFT (instead of the HS) representation is used. All these images contain structures that are poorly represented by HS elements, which encode color information. Based on local gray-scale edge directions, the SIFT visterms can efficiently depict parts of these structures, and allow to discriminate between e.g. white house and a polar bear that would be represented by a similar HS histogram. In Figure 5.10, we see that the *house* average precision values are more than two times bigger for the SIFT representation than for the Blob representation.

As already shown in Table 5.3, the concatenation of the HS and SIFT representations provides the best ranking performance of the system. More precisely, it improves the average precision of 121 words compared to the SIFT-only representation, and 121 words compared to the HS-only representation. This complementarity can be analyzed in more details on the 10 words considered in Figure 5.10. The concatenation of HS and SIFT features improves the average precision of 9 of the 10 words in Figure 5.10 on all of them on average, as shown in Table 5.3.

Regarding limitations of the HS+SIFT combination, note that for some words, like *house* in Figure 5.10, combining the SIFT and the HS representations actually produces a worse image ranking than the SIFT-only case. This indicates that some ambiguity is introduced by the HS features in the related images, making them more similar to other images that are annotated with different words. Better mechanisms for data fusion could thus potentially improve the system performance, because a few words are better represented by one of the two feature types than by their simple concatenation.

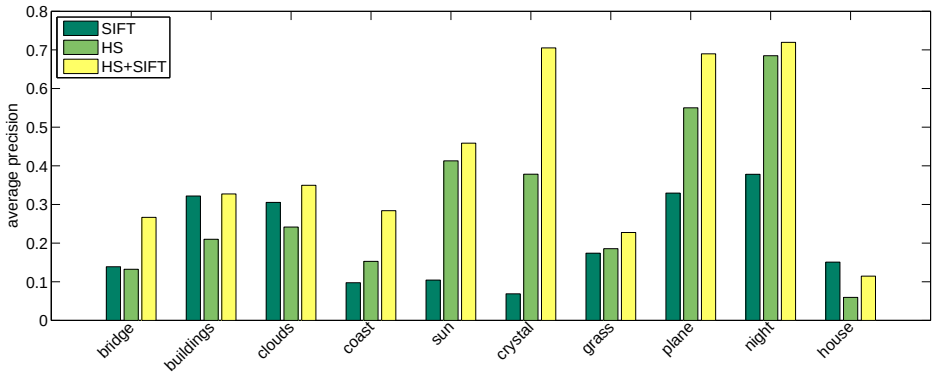


Figure 5.10: Average precision of 10 selected words when SIFT (*dark green*), HS (*green*) and HS+SIFT (*yellow*) features represent the image. Depending on the word, the average precision values are higher for one of the two representations, and the combination of both improves in general.

The fact that one model is learned for all the words does not allow a basic word-dependent weighting of the features, and more elaborate schemes have to be explored in the future.

5.6.7 Ranking examples

The AP measure of a specific query gives a good indication of the system performance, allowing the comparison of different annotation strategies or different feature combinations, as we have done in the previous sections. Here, we illustrate the retrieval performance of five queries, showing the 48 top-ranked images given the $P(w \mid d)$ estimated with the PLSA-WORDS method, using the HS+SIFT feature combination. From a total of 1783 ranked images, these top-ranked images correspond to what could be displayed on a web browser window, what corresponds to an online retrieval scenario. We have chosen five queries, with AP values ranging between 28% and 1.2%, that are representative of the variation in performance observed in our dataset. In each figure, the images that are actually annotated with the query word according to the ground-truth are shown within a green box. Images are ranked as a left-right, top-down sequence.

The first retrieval result, shown on Figure 5.11, corresponds to the query *street*. In this example, our proposed PLSA-WORDS learning procedure allowed to retrieve 16 images out of 64 that are annotated with the word *street* in the ground-truth. The other top-ranked images show what type of visual information is linked with the word *street* by the PLSA-WORDS model. A majority of images contain a building or a boat, corresponding to similar color distributions and textures. For a number of images that contain a building structure, the word *street* could actually be a valid annotation. The same observation can be made from Figure 5.12, that illustrates the retrieval of the query *valley*. If only 9 images from the 48 top-ranked images are actually annotated with the word *valley* in the ground-truth, other images could be considered as correctly representing the concept *valley*. This indicates how the performance evaluation can be penalized by the non-exhaustive ground-truth annotation: although some retrieved images objectively relate to the query, they are not considered as correct according to the ground-truth annotation.

Figure 5.13 shows the top-ranked images for the query *eagle*, where 9 out of 21 *eagle* images are correctly retrieved. Some of the top-ranked images, related to planes, are interestingly similar to a flying eagle example. The blue sky context in images is obviously taken into account in priority, what explains the confusion with *planes* and other images containing *sky*. On Figure 5.14, we show the ranking obtained for the query *grapes*. This word corresponds to a very specific visual representation, that apparently fails to be accurately captured by our image representation. Only one image out of 7 is correctly retrieved in that case, while other retrieved images relate to *vegetation* in general. In that

sense, the model implicitly linked the grapes training examples with images containing *vegetation*, without discriminating *grapes* from *tree* images for instance. Figure 5.15 shows the retrieval obtained for the *lynx* query. While a majority of top-ranked images contains an animal in various environments, none of them actually contains a *lynx*. The type of retrieved images however indicates that the model learned a correspondence between the *lynx* image examples from the training set and the concept of *animal* in the data.

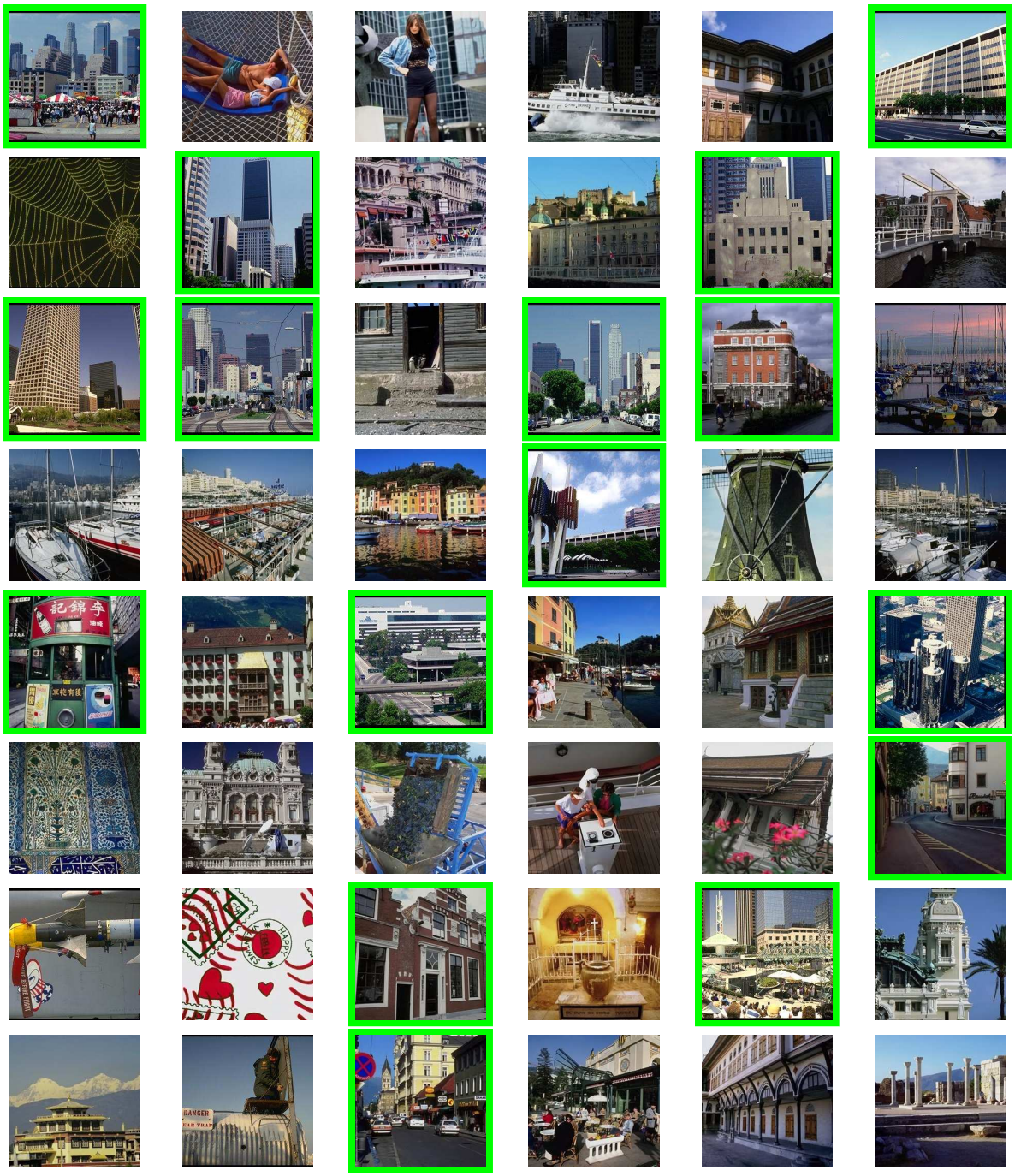


Figure 5.11: 48 top-ranked images from 1783 test images given the query *street*, for $P(w \mid d)$ estimated by PLSA-WORDS, using the HS+SIFT image representation. Images are ranked as a left-right, top-down sequence, and images annotated with the query word in the ground-truth are displayed in a box. 16 *street* images, out of 64, are correctly retrieved in the 48 top-ranked images. The full ranking corresponds to an AP of 28%.

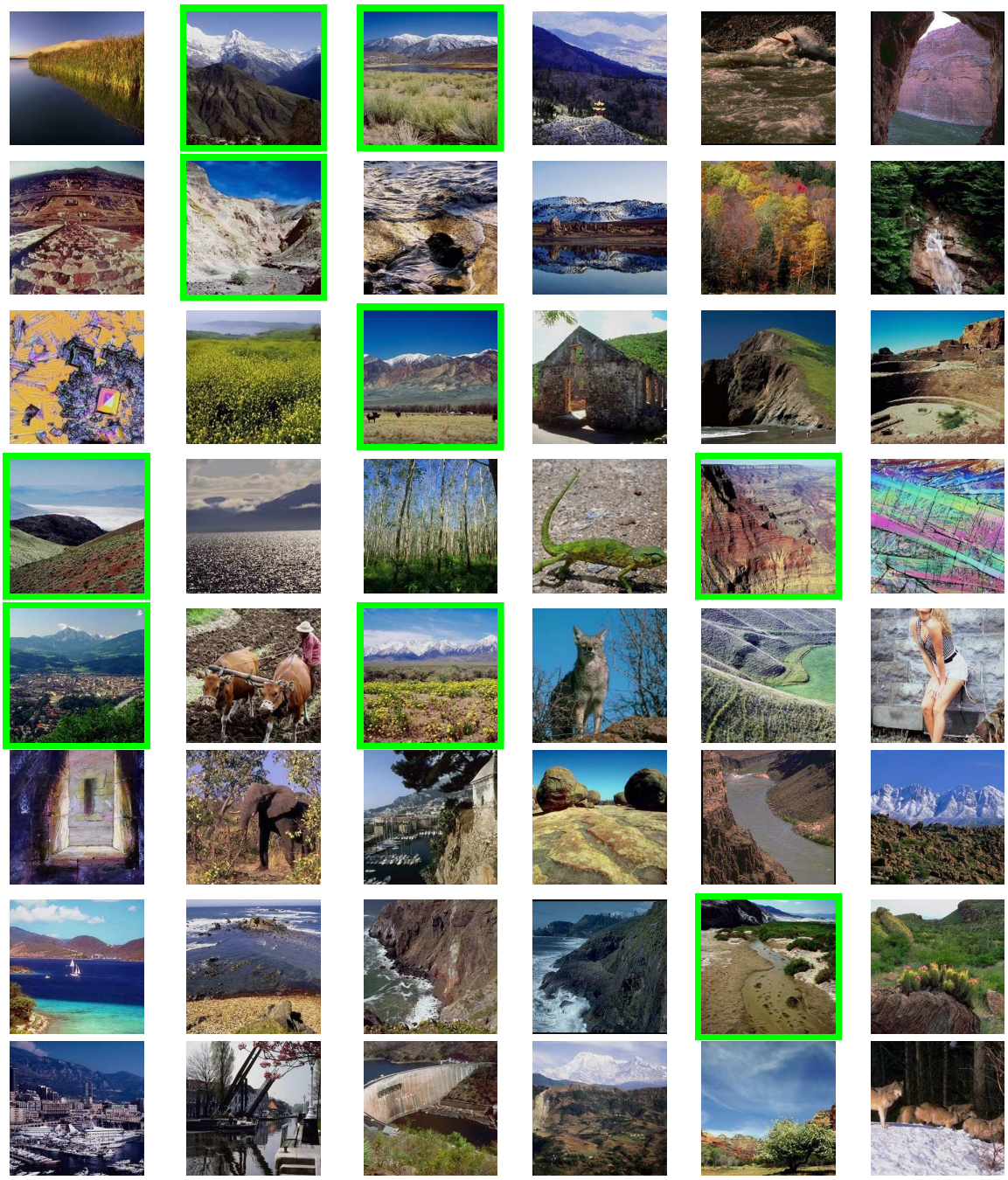


Figure 5.12: 48 top-ranked images from 1783 test images given the query *valley*, for $P(w \mid d)$ estimated by PLSA-WORDS, using the HS+SIFT image representation. Images are ranked as a left-right, top-down sequence, and images annotated with the query word in the ground-truth are displayed in a box. 9 *valley* images, out of 23, are correctly retrieved in the 48 top-ranked images. The full ranking corresponds to an AP of 18.3%.

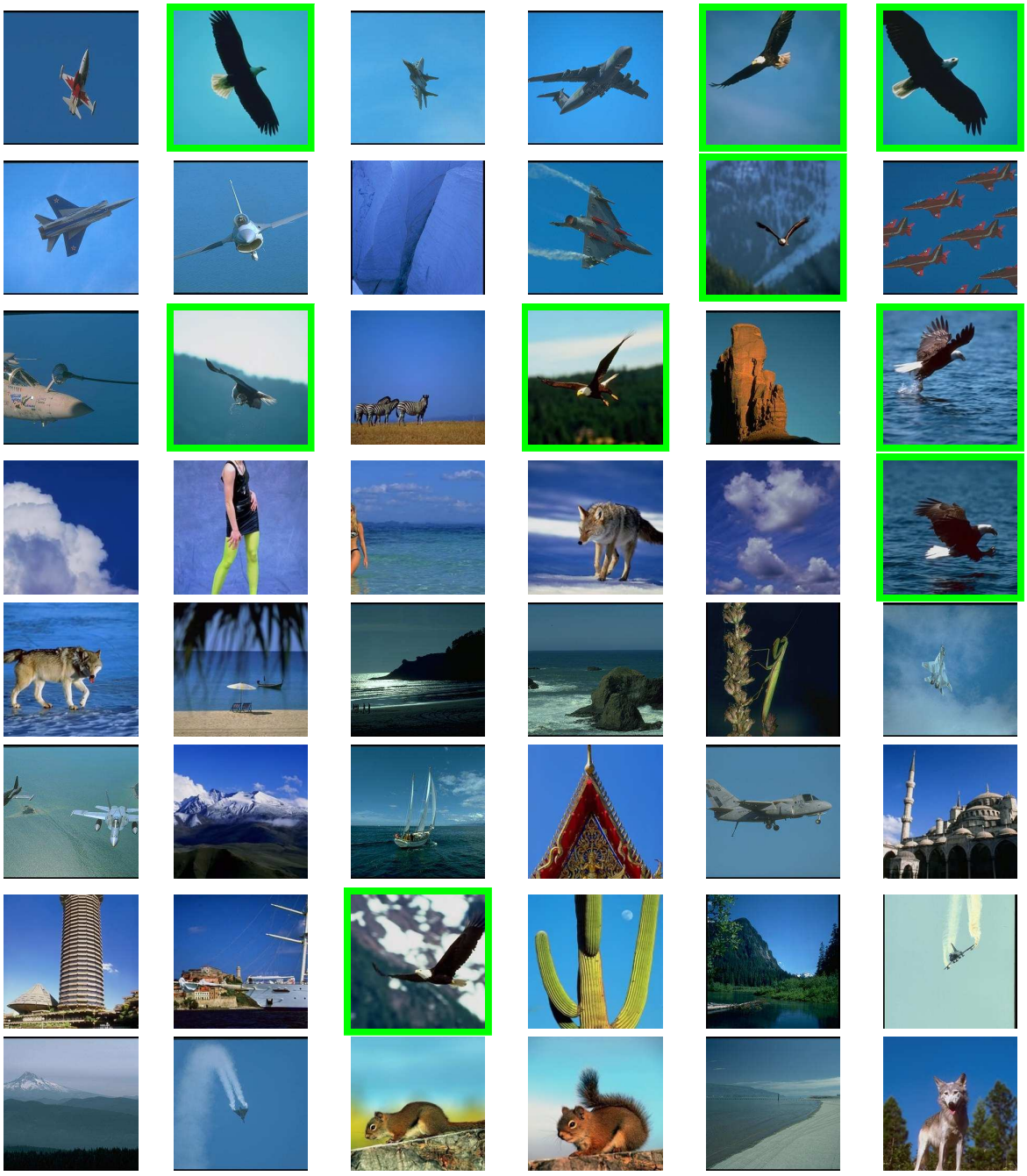


Figure 5.13: 48 top-ranked images from 1783 test images given the query *eagle*, for $P(w \mid d)$ estimated by PLSA-WORDS, using the HS+SIFT image representation. Images are ranked as a left-right, top-down sequence, and images annotated with the query word in the ground-truth are displayed in a box. 9 *eagle* images, out of 21, are correctly retrieved in the 48 top-ranked images. The full ranking corresponds to an AP of 20.5%.

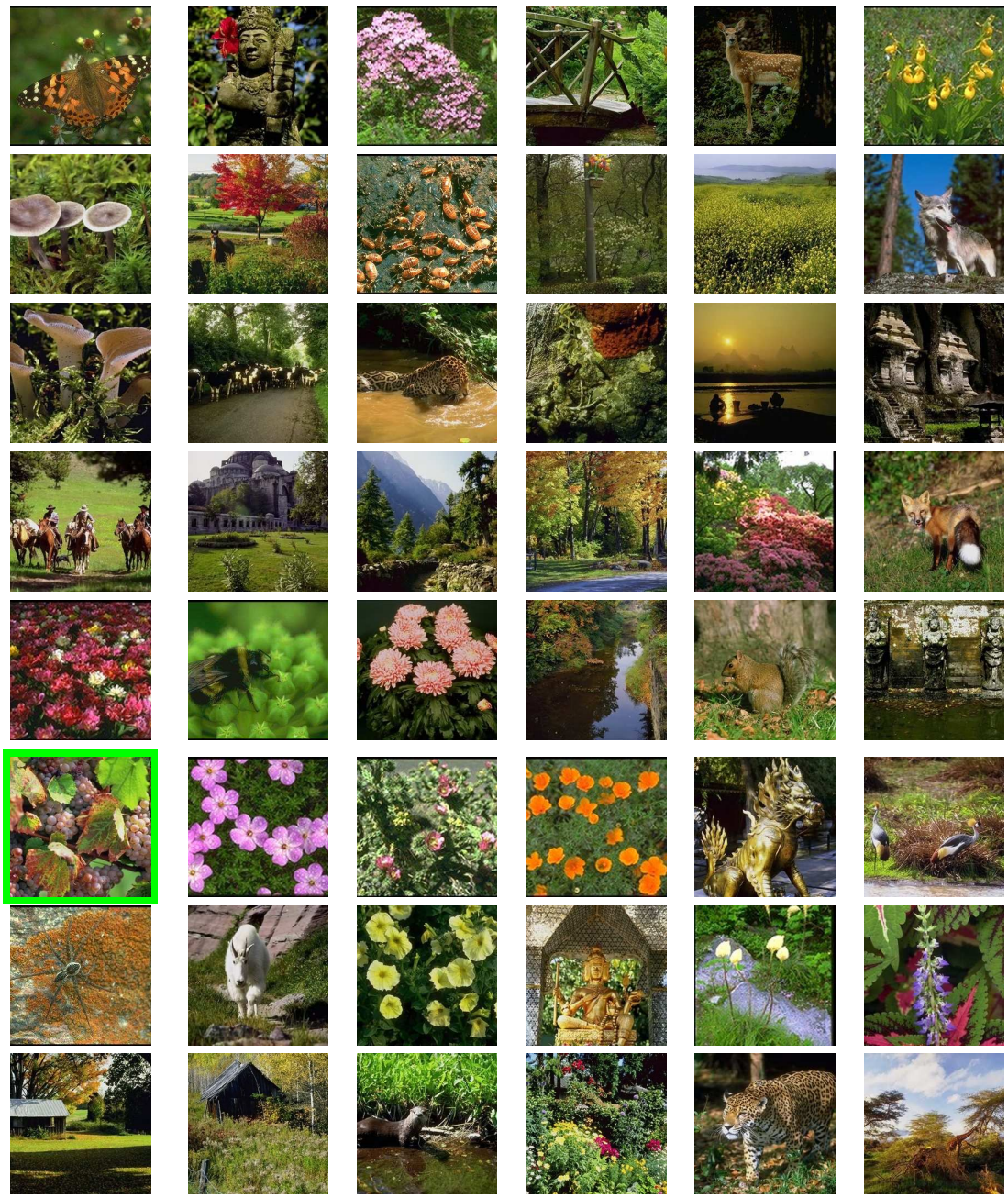


Figure 5.14: 48 top-ranked images from 1783 test images given the query *grapes*, for $P(w \mid d)$ estimated by PLSA-WORDS, using the HS+SIFT image representation. Images are ranked as a left-right, top-down sequence, and images annotated with the query word in the ground-truth are displayed in a box. Only 1 *grapes* image, out of 7, is correctly retrieved in the 48 top-ranked images. The full ranking corresponds to an AP of 2.5%.

Figure 5.15: 48 top-ranked images from 1783 test images given the query *lynx*, for $P(w \mid d)$ estimated by PLSA-WORDS, using the HS+SIFT image representation. Images are ranked as a left-right, top-down sequence, and images annotated with the query word in the ground-truth are displayed in a box. Out of 13 *lynx* images, none is correctly retrieved in the 48 top-ranked images. The full ranking corresponds to an AP of 1.2%.

5.7 Conclusion

In this chapter, we presented three alternative algorithms to learn a PLSA model for annotated images, and evaluated their ability for cross-media image indexing. The learning methods differ in which of the textual or the visual modality is dominant to learn the mixture of aspects for an image and its text caption, and these differences influence the accuracy of the inferred semantic indices. The best retrieval performance is achieved when the mixture of latent aspects is learned from the text captions (our PLSA-WORDS model), creating semantically meaningful aspects. Combining quantized local color information with quantized local image descriptors appeared to be as successful strategy. We have demonstrated their complementarity and their improved performance when compared to the standard Blob representation. The performance of all the models was improved by the use of this combined image representation, that depicts an image as a set of local color-based regions and local texture-based regions. In particular, the PLSA-WORDS model achieved the best performance with respect to recent methods.

The quality of the image ranking greatly varies depending on the query, and we analyzed the possible factors in the case of the PLSA-WORDS model. Besides the difference in the number of training examples or the suitability of the visual features to represent a given concept, we have shown strong indications that PLSA-WORDS can take advantage of the co-occurrence of words in the text captions of the training images.

Chapter 6

Conclusions and future directions

This chapter summarizes the contributions of our work presented in this dissertation. We also point out potential research directions that we either partially addressed in the different chapters, or that were suggested by our investigations.

6.1 Summary and contributions

The central theme of this dissertation was to model images as mixtures of latent aspects, where aspects are defined by multinomial distributions over quantized local patches. Aspect-based image models represent a novel way of processing visual information, capturing patch co-occurrence patterns in an image collection without modeling any spatial relationship between them. In the preceding chapters, we have investigated different implications of this family of image models, opening new perspectives for various computer vision and image retrieval tasks. So far, this idea had been addressed by (for the most part) isolated works that only considered a particular application of the concept. We investigated essential implications of the aspect-based image modeling approach, proposing an in-depth, unifying view of the problem. Several contributions have resulted from these investigations:

- An aspect-mixture image representation can be estimated from the bag-of-patches representation of an image given a model learned on unlabeled data. This allows to take advantage of unlabeled data, a situation increasingly common with the ubiquitous availability of digital cameras, to improve the classification of images depicting scenes or containing objects.
- Aspects, although obtained by unsupervised learning, allow for an interesting, visually-consistent soft-clustering of image collections, which is suitable for the visualization and browsing of unorganized image collections.
- The classification of the quantized image patches, when the co-occurrence context captured by the aspects is taken into account, produces an interesting form of image segmentation.
- The visual and the textual modalities of an annotated image can be linked through a common aspect decomposition given this image. This joint textual and visual modeling allows to automatically annotate a new image.

The evaluation of the aspect-based image representation for classification was conducted in Chapter 3, in which we considered different scene and object classification tasks. Relying on various image representations based on the combination of recent point detectors and local descriptors, we showed the benefit of the unsupervised learning of patch co-occurrence. For the same amount of training data to learn an SVM classifier, an aspect model can take advantage of unlabeled data to derive a new representation from the bag-of-patches that achieves higher classification performance. The mixture

of aspects inferred for a new image incorporates the co-occurrence information learned from the unlabeled data, and this additional information helps the classification process when only a small number of labeled examples is available.

In Chapter 4, the patch co-occurrence context identified by the aspect model was exploited to classify image regions into classes. This was suggested by our previous investigations: decomposing an image into a mixture of latent aspects, defined by a distribution over quantized regions, can actually be interpreted as a basic form of image segmentation. We therefore derived two algorithms to estimate the class conditional probability of a given quantized patch. The first is independent of the image that is considered, the second takes into account the aspect mixture weights of the image from which the region was extracted. We showed that the second option, i.e. making the classification of image regions dependent on the image context, improves the region classification performance. We also showed that this co-occurrence context can be successfully combined with the spatial context modeled by a traditional MRF.

The visual aspects are obtained by unsupervised learning, and there is a priori no reason for an aspect to relate to a particular visual concept. Similarly to what was shown in the text case [7, 29, 10] - where aspects are illustrated by their most probable words - we proposed to visualize aspects from their most representative training images. We showed that aspects can correspond to specific types of image content, defined by a specific pattern of patch co-occurrence.

Different ways to learn an aspect decomposition from the textual and the visual modalities of an annotated image collection were investigated in Chapter 5. The two modalities jointly define a document, and we link them by assuming a common aspect decomposition for the textual and visual modalities of a given image. We proposed three algorithms to learn the aspect distributions and the aspect mixture weights from the two modalities, differing in which modality is used to estimate the aspect mixture weights for training documents. Given a model learned on a set of annotated images, a word distribution can be inferred for any new image, and this word distribution can serve as an index for a keyword-based image retrieval task. We showed that learning the aspect mixture weights of training documents from the textual modality allows to infer the most efficient word indexing, corresponding to the best retrieval performance. Moreover, this aspect-based image annotation outperforms recent image-annotation methods when a representation based on the concatenation of quantized color and texture descriptors, that we proposed, is used.

6.2 Future research directions

We have explored the concept of mixture of aspects for images in detail, showing the possible implications of the approach for various tasks. More investigation, however, could be conducted along the same line, and potential research directions are mentioned in the following two sections.

6.2.1 Integration of spatial information

The bag-of-patches representation was chosen for its simplicity. A histogram of quantized image regions does not contain any information about the spatial relationships between regions, although the relative position of patches is a valid information for its interpretation. In our work, we only considered this information in Chapter 4 to smooth the classification of regions based on the co-occurrence context. A formulation that would jointly model the co-occurrence and the spatial context could be a better alternative.

Absolute or relative region location can be considered, however the best strategy to incorporate spatial information is not obvious. The absolute region location within the image, quantized based on a fixed grid for instance, is a first possibility. In that case, the spatial relationship between image regions is however only modeled implicitly, as resulting from their respective positions. On the other hand, relative position information could be directly built in the same model by reconsidering the observations. Instead of isolated quantized patches, sets of neighboring quantized patches can be

considered as the observation. This directly incorporates the desired information, but leaves several open issues, i.e. for deciding on the size of the neighborhood, or weighting the importance of neighbors based on their respective distance. This idea is mentioned in [75], with a short discussion about the advantage of a vocabulary constructed from two juxtaposed regions for image segmentation.

Spatial information, in the form of absolute or relative positioning, is however not guaranteed to improve the performance on subsequent tasks. Learning the spatial layout of patches in a scene in addition to their co-occurrence certainly adds complexity to the model, but will not necessarily help the classification of scenes in different classes for instance. All these issues should be investigated in details.

6.2.2 Filling incomplete image annotations

In Chapter 5, we presented three models for annotated images and evaluated their performance in the context of image annotation. Instead of predicting an annotation when no word exists, the same models could be used to infer new words given an image with an incomplete annotation. Image annotations are indeed generally incomplete, as the choice of words entirely depends on the image interpretation: two persons are likely to interpret the same image differently, thus attaching different words to it. A more exhaustive annotation would be obtained by guessing the missing words based on both the image content and the current, incomplete annotation. The aspect-based models for annotated images proposed in Chapter 5 are potential candidates for this task. The inference of the aspect mixture weights for a test image should be modified to take the textual and the visual modalities into account, given that the current annotation inference algorithm assumes that only the visual modality is available. The introduction of a textual modality for unseen documents should be investigated, as several strategies are conceivable.

A tool based on such a model, able to re-estimate the word distribution given an image when a new word is selected, would be interesting to support the annotation process. A possible scenario would involve the following steps: (i) a list of words is displayed, ranked according to the word probability estimated from the visual image representation; (ii) the person who annotates selects a correct word from the list; (iii) the word distribution is re-estimated based on this new textual observation. If the re-estimated word distribution is more accurate than the previous one, then iterating between the steps (ii) and (iii) would make the human annotation converge to an exhaustive result faster. Collaborative annotation, which is the current trend to index online image collections, could benefit from this help.

Bibliography

[1] S. Agarwal, A. Awan, and D. Roth. Learning to detect objects in images via a sparse, part-based representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26:1475–1490, 2004.

[2] S. Agarwal and D. Roth. Learning a sparse representation for object detection. In *Proc. of the IEEE European Conference on Computer Vision*, Copenhagen, May 2002.

[3] P. Ahrendt, J. Larsen, and C. Goutte. Co-occurrence models in music genre classification. In *Proc. of the IEEE Workshop on Machine Learning for Signal Processing*, Mystic, Sep. 2005.

[4] R. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval*. ACM Press, 1999.

[5] K. Barnard, P. Duygulu, N. Freitas, D. Forsyth, D. Blei, and M.I. Jordan. Matching words and pictures. *Journal of Machine Learning Research*, 3:1107–1135, 2003.

[6] D. Blei and M. Jordan. Modeling annotated data. In *Proc. of the International Conference on Research and Development in Information Retrieval (SIGIR)*, Toronto, Aug. 2003.

[7] D. Blei, A. Ng, and M. Jordan. Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[8] M.R. Boutell, J. Luo, X. Shen, and C.M.Brown. Learning multi-label scene classification. *Pattern Recognition*, 37(9):1757–1771, 2004.

[9] W. Buntine. Operations for learning with graphical models. *Journal of Artificial Intelligence Research*, 2:159–225, 1994.

[10] W. Buntine. Variational extensions to EM and multinomial PCA. In *Proc. of the European Conference on Machine Learning*, London, Aug. 2002.

[11] W. Buntine and A. Jakulin. Discrete component analysis. In *Proc. of the PASCAL Workshop on Subspace, Latent Structure and Feature Selection techniques: Statistical and Optimisation perspectives*, Bohinj, Feb. 2005.

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

[13] C. Carson, S. Belongie, H. Greenspan, and J. Malik. Blobworld: Image segmentation using expectation-maximization and its application to image querying. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24:1026–1038, 2002.

[14] C. Carson, M. Thomas, S. Belongie, J. M. Hellerstein, and J. Malik. Blobworld: A system for region-based image indexing and retrieval. In *Proc. of the International Conference on Visual Information and Information Systems*, Amsterdam, Jun. 1999.

[15] E. Y. Chang, K. Goh, G. Sychay, and G. Wu. CBSA: Content-based soft annotation for multimodal image retrieval using bayes point machines. *IEEE Transactions on Circuits and Systems for Video Technology*, 13:26–38, 2003.

[16] S. Deerwester, S.T. Dumais, T.K. Landauer, G.W. Furnas, and Harshman. Indexing by latent semantic analysis. *Journal of the Society for Information Science*, 41(6):391–407, 1990.

[17] G. Dorko and C. Schmid. Selection of scale invariant parts for object class recognition. In *Proc. of the IEEE International Conference on Computer Vision*, Nice, Oct. 2003.

[18] P. Duygulu, K. Barnard, J.F.G de Freitas, and D.A. Forsyth. Object recognition as machine translation: learning a lexicon for a fixed image vocabulary. In *Proc. of the European Conference on Computer Vision*, Copenhagen, May 2002.

[19] J. Dy, C. Brodley, A. C. Kak, L. Broderick, and A. Aisen. Unsupervised feature selection applied to content-based retrieval of lung images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25:373–378, 2003.

[20] J. Fauqueur and N. Boujemaa. New image retrieval paradigm: logical composition of region categories. In *Proc. of the International Conference on Image Processing*, Barcelona, October 2003.

[21] L. Fei-Fei, R. Fergus, and P. Perona. A bayesian approach to unsupervised one-shot learning of object categories. In *Proc. of the IEEE International Conference on Computer Vision*, Nice, Oct. 2003.

[22] L. Fei-Fei, R. Fergus, and P. Perona. Learning generative visual models from few training examples: an incremental Bayesian approach tested on 101 object categories. In *Proc. of the IEEE International Conference on Computer Vision, Workshop on Generative-Model Based Vision*, Washington DC, Jun. 2004.

[23] L. Fei-Fei and P. Perona. A Bayesian hierarchical model for learning natural scene categories. In *Proc. of IEEE International Conference on Computer Vision And Pattern Recognition*, San Diego, Jun. 2005.

[24] S. L. Feng, R. Manmatha, and V. Lavrenko. Multiple Bernoulli relevance models for image and video annotation. In *Proc. of the IEEE International Conference on Computer Vision and Pattern Recognition*, Washington, Jun. 2004.

[25] R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. In *Proc. of the IEEE International Conference on Computer Vision and Pattern Recognition*, Toronto, Jun. 2003.

[26] S. Geman and D. Geman. Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6:721–741, 1984.

[27] A. Girgensohn, J. Adcock, and L. Wilcox. Leveraging face recognition technology to find and organize photos. In *Proc. of the ACM SIGMM international workshop on Multimedia information retrieval*, New York, 2004.

[28] M. Gorkani and R. Picard. Texture orientation for sorting photos at glance. In *Proc. of the International Conference on Pattern Recognition*, Jerusalem, Sep. 1994.

[29] T. Hofmann. Unsupervised learning by probabilistic latent semantic analysis. *Machine Learning*, 42:177–196, 2001.

[30] J., S. Belongie, J. Shi, and T. Leung. Textons, contours and regions: Cue integration in image segmentation. In *Proc. of the IEEE International Conference on Computer Vision*, Kerkyra, Sep. 1999.

[31] J. Jeon, V. Lavrenko, and R. Manmatha. Automatic image annotation and retrieval using cross-media relevance models. In *Proc. of the International Conference on Research and Development in Information Retrieval (SIGIR)*, Toronto, Aug. 2003.

[32] J. Jeon and R. Manmatha. Using maximum entropy for automatic image annotation. In *Proc. of the IEEE International Conference on Image and Video Retrieval*, Dublin, Jul. 2004.

[33] S. Kumar and M. Herbert. Discriminative Random Fields: a discriminative framework for contextual interaction in classification. In *Proc. of the IEEE International Conference on Computer Vision*, Nice, Oct. 2003.

[34] S. Kumar and M. Herbert. Man-made structure detection in natural images using a causal multiscale random field. In *Proc. of the IEEE International Conference on Computer Vision and Pattern Recognition*, Toronto, Jun. 2003.

[35] V. Lavrenko, R. Manmatha, and J. Jeon. A model for learning the semantics of pictures. In *Proc. of Advances in Neural Information Processing Systems*, Vancouver and Whistler, Dec. 2003.

[36] S. Lazebnik, C. Schmid, and J. Ponce. Affine-invariant local descriptors and neighborhood statistics for texture recognition. In *Proc. of the International Conference on Computer Vision*, Nice, Oct. 2003.

[37] B. Leibe and B. Schiele. Interleaved object categorization and segmentation. In *Proc. of the British Machine Vision Conference*, Norwich, Sep. 2003.

[38] S. Levy and B. Stone. The new wisdom of the web. *Newsweek*, Apr. 2006.

[39] C.-S. Li and V. Castelli. Deriving texture feature set for content-based retrieval of satellite image database. In *Proc. of the IEEE International Conference on Image Processing*, Washington, Oct. 1997.

[40] J. Li and J. Z. Wang. Automatic linguistic indexing of pictures by a statistical modeling approach. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 25:1075–1088, 2003.

[41] S. Z. Li. *Markov Random Field Modeling in Computer Vision*. Springer, 1995.

[42] J.-H. Lim and J.S. Jin. Semantics discovery for image indexing. In *Proc. of the European Conference on Computer Vision*, Prague, May 2004.

[43] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60:91–110, 2004.

[44] J. Malik, S. Belongie, T. Leung, and J. Shi. Contour and texture analysis for image segmentation. *International Journal of Computer Vision*, 43:7–27, 2001.

[45] J. Matas, O. Chum, U. Martin, and T. Pajdla. Robust wide baseline stereo from maximally stable extremal regions. In *Proc. of the British Machine Vision Conference*, Cardiff, Sep. 2002.

[46] K. Mikolajczyk and C. Schmid. Scale and affine interest point detectors. *International Journal of Computer Vision*, 60(1):63–86, 2004.

[47] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(10):1615–1630, 2005.

[48] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky T. Kadir, and L. Van Gool. A comparison of affine region detectors. *International Journal of Computer Vision*, 65:43–72, 2005.

[49] T. Minka and J. Lafferty. Expectation-propagation for the generative aspect model. In *Proc. of the Conference on Uncertainty in Artificial Intelligence*, Edmonton, Aug. 2002.

[50] B. Moghaddam, Q. Tian, N. Lesh, C. Shen, and T. S. Huang. Visualization and user-modeling for browsing personal photo libraries. *International Journal of Computer Vision*, 56:109–130, 2004.

[51] F. Monay and D. Gatica-Perez. On image auto-annotation with latent space models. In *Proc. of the ACM International Conference on Multimedia*, Berkeley, Nov. 2003.

[52] F. Monay and D. Gatica-Perez. PLSA-based image auto-annotation: Constraining the latent space. In *Proc. of the ACM International Conference on Multimedia*, New York, Oct. 2004.

[53] F. Monay and D. Gatica-Perez. Modeling semantic aspects for cross-media image indexing. IDIAP-RR 56, IDIAP, Martigny, Switzerland, 2005. Accepted for publication in IEEE Transactions on Pattern Analysis and Machine Intelligence.

[54] F. Monay, P. Quelhas, D. Gatica-Perez, and J.-M. Odobez. Constructing visual models with a latent space approach. In *Proc. of the PASCAL Workshop on Subspace, Latent Structure and Feature Selection techniques: Statistical and Optimisation perspectives*, Bohinj, Feb. 2005.

[55] F. Monay, P. Quelhas, D. Gatica-Perez, and J.-M. Odobez. Integrating co-occurrence and spatial contexts on patch-based scene segmentation. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, Beyond Patches Workshop*, New York, Jun. 2006.

[56] F. Monay, P. Quelhas, J.-M. Odobez, and D. Gatica-Perez. Contextual scene segmentation with aspect models. IDIAP-RR 30, IDIAP, Martigny, Switzerland, 2005. Submitted for journal publication.

[57] Y. Mori, H. Takahashi, and R. Oka. Image-to-word transformation based on dividing and vector quantizing images with words. In *Proc. of the International Workshop on Multimedia Intelligent Storage and Retrieval Management*, Orlando, Oct. 1999.

[58] H. Mueller, S. Marchand-Maillet, and T. Pun. The truth about Corel: Evaluation in image retrieval. In *Proc. of the International Conference on Image and Video Retrieval*, London, Jul. 2002.

[59] M. Naphade and T. Huang. A probabilistic framework for semantic video indexing, filtering and retrieval. *IEEE Transactions on Multimedia*, 3(1):141–151, Mar. 2001.

[60] W. Niblack, R. Barber, W. Equitz, M. Flicker, E. Glasman, D. Petkovic, P. Yanker, and C. Faloutsos. The QBIC project: query images by content using color, texture and shape. In *Proc. of the SPIE Conference on Storage and Retrieval for Image and Video Databases*, San Jose, Feb. 1993.

[61] A. Oliva and A. Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International Journal of Computer Vision*, 42:145–175, 2001.

[62] A. Opelt, M. Fussenegger, A. Pinz, and P. Auer. Weak hypotheses and boosting for generic object detection and recognition. In *Proc. of the IEEE European Conference on Computer Vision*, Prague, May 2004.

[63] M. Ortega, Y. Rui, K. Chakrabarti, S. Mehrotra, and T. S. Huang. Supporting similarity queries in MARS. In *Proc. of the ACM International Conference on Multimedia*, Seattle, Nov. 1997.

[64] S. Paek and S.-F. Chang. A knowledge engineering approach for image classification based on probabilistic reasoning systems. In *Proc. of the IEEE International Conference on Multimedia and Expo*, New York, Aug. 2000.

[65] N. R. Pal and S. K. Pal. A review on image segmentation techniques. *Pattern recognition*, 26:1277–1294, 1993.

[66] J.-Y. Pan, H.-J. Yang, P. Duygulu, and C. Faloutsos. Automatic image captioning. In *Proc. of the IEEE International Conference on Multimedia and Expo*, Taiwan, Jun. 2004.

[67] P. Quelhas, F. Monay, J.-M. Odobez, D. Gatica-Perez, and T. Tuytelaars. A thousand words in a scene. IDIAP-RR 40, IDIAP, Martigny, Switzerland, 2005. Accepted for publication in IEEE Transactions on Pattern Analysis and Machine Intelligence.

[68] P. Quelhas, F. Monay, J.-M. Odobez, D. Gatica-Perez, T. Tuytelaars, and L. V. Gool. Modeling scenes with local descriptors and latent aspects. In *Proc. of the IEEE International Conference on Computer Vision*, Beijing, Oct. 2005.

[69] Y. Rui, T. Huang, and S. Chang. Image retrieval: current techniques, promising directions and open issues. *Journal of Visual Communication and Image Representation*, 10:39–62, 1999.

[70] G. Salton. *Automatic Information Organization and Retrieval*. McGraw Hill Text, 1968.

[71] G. Salton, A. Wong, and C. S. Yang. A vector space model for automatic indexing. *Communications of the ACM*, 18:613–620, 1975.

[72] N. Serrano, A. Savakis, and J. Luo. A computationally efficent approach to indoor/outdoor scene classification. In *Proc. of the International Conference on Pattern Recognition*, Quebec, Aug. 2002.

[73] H. Shao, T. Svoboda, V. Ferrari, T. Tuytelaars, and L. Van Gool. Fast indexing for image retrieval based on local appearance with re-ranking. In *Proc. of the IEEE International Conference on Image Processing*, Barcelona, Sep. 2003.

[74] J. Shi and J. Malik. Normalized cuts and image segmentation. In *Proc. of the IEEE International Conference on Computer Vision and Pattern Recognition*, San Juan, Jun. 1997.

[75] J. Sivic, B. C. Russell, A. A. Efros, A. Zisserman, and W. T. Freeman. Discovering objects and their location in images. In *Proc. of the IEEE International Conference on Computer Vision*, Beijing, Oct. 2005.

[76] J. Sivic and A. Zisserman. Video Google: A text retrieval approach to object matching in videos. In *Proc. of the IEEE International Conference on Computer Vision*, Nice, Oct. 2003.

[77] J. Sivic and A. Zisserman. Video data mining using configurations of viewpoint invariant regions. In *Proc. of the IEEE International Conference on Computer Vision and Pattern Recognition*, Washington DC, Jun. 2004.

[78] A. Smeaton and P. Over. The TREC-2002 video track report. In *Text REtrieval Conference*, Gaithersburg, Nov. 2002.

[79] A. W. M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain. Content-based image retrieval: the end of the early years. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22:1349–1380, 2000.

[80] J. R. Smith and S.-F. Chang. Visualseek: a fully automated content-based image query system. In *Proc. of the ACM International Conference on Multimedia*, Boston, Nov. 1996.

[81] J. R. Smith, C. Lin, M. Naphade, A. Natsev, and B. Tseng. Multimedia semantic indexing using model vectors. In *Proc. of the IEEE International Conference on Multimedia and Expo*, Baltimore, Jul. 2003.

[82] M. Steyvers and T. Griffiths. *Latent Semantic Analysis: A road to meaning*, chapter Probabilistic topic models. Erlbaum, 2006.

[83] M. Szummer and R.W. Picard. Indoor-outdoor image classification. In *Proc. of the IEEE International Workshop CAIVD, in ICCV'98*, Bombay, Jan. 1998.

[84] K. Thieu and P. Viola. Boosting image retrieval. *International Journal of Computer Vision*, 56:17–36, 2004.

[85] B. Thorsten. Test data likelihood for PLSA models. *Information Retrieval*, 8:181–196, 2005.

[86] T. Tuytelaars and L. Van Gool. Content-based image retrieval based on local affinely invariant regions. In *Proc. of the International Conference on Visual Information and Information Systems*, Amsterdam, Jun. 1999.

[87] A. Vailaya, M. Figueiredo, A. Jain, and H.J. Zhang. Image classification for content-based indexing. *IEEE Transactions on Image Processing*, 10(1):117–130, 2001.

[88] A. Vailaya, A. Jain, and H.J. Zhang. On image classification: City images vs. landscapes. *Pattern Recognition*, 31:1921–1935, 1998.

[89] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Proc. of the IEEE International Conference on Computer Vision and Pattern Recognition*, Hawaii, Dec. 2001.

[90] P. Viola and M. J. Jones. Robust real-time face detection. *International Journal of Computer Vision*, 57:137–154, 2004.

[91] J. Vogel and B. Schiele. Natural scene retrieval based on a semantic modeling step. In *Proc. of the International Conference on Image and Video Retrieval*, Dublin, Jul. 2004.

[92] J. Vogel and B. Schiele. A semantic typicality measure for natural scene categorization. In *Proc. of the Pattern Recognition Symposium DAGM'04*, Tübingen, Germany, September 2004.

[93] J. Weston and C. Watkins. Multi-class support vector machines. Technical Report CSD-TR-98-04, Department of Computer Science, Royal Holloway, University of London, May 1998.

[94] J. Willamowski, D. Arregui, G. Csurka, C.R. Dance, and L. Fan. Categorizing nine visual classes using local appearance descriptors. In *Proc. of the LAVS Workshop, in ICPR'04*, Cambridge, Aug. 2004.

[95] R. Zhang and Z. Zhang. Hidden semantic concept discovery in region based image retrieval. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, Washington, D.C., Jun. 2004.