NORTHWESTERN UNIVERSITY

Semantic Labeling for Image Classification

A DISSERTATION

SUBMITTED TO THE GRADUATE SCHOOL

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

for the degree

DOCTOR OF PHILOSOPHY

Field of Electrical Engineering and Computer Science

By

Depalov Dejan

EVANSTON, ILLINOIS

June 2007

ABSTRACT

Semantic Labeling for Image Classification

Depalov Dejan

The rapid growth of digital imaging technology and the accumulation of large collections of digital images has created the need for efficient and intelligent schemes for content-based image retrieval. Our goal is to organize the contents semantically, according to meaningful categories. We present a new approach for semantic classification that utilizes a recently proposed color-texture segmentation algorithm (by Chen *et al.*), which combines knowledge of human perception and signal characteristics to segment natural scenes into perceptually uniform regions. The features of these regions are then used as medium level descriptors that can effectively bridge the "semantic gap" between low level primitives and high level semantics. The goal is to extract semantic labels, first at the segment and then at the scene level. The focus of this thesis is on region classification. We develop segment features that consist of spatial texture orientation information and color composition in terms of a limited number of locally adapted dominant colors. We also consider segment size and position. We use a hierarchical vocabulary of segment labels that is consistent with subjective experiments and the labels used in the NIST TRECVID 2003 development set. We have gathered a database of 13000 automatically segmented and manually labeled segments obtained from 3300 photographs of natural scenes. This database is used for training and testing. For training and classification we use the Linear Discriminant Analysis (LDA) technique. We examine the performance of the algorithm (precision and recall rates) when different sets of features (*e.g.,* one or two most dominant colors versus four quantized dominant colors) are used. We also

consider the performance of other techniques such as Gaussian Mixture Models (GMMs) and Support Vector Machines (SVMs). Our results indicate that the proposed approach offers significant performance improvements over existing approaches. We also compare with human performance. For this, we use human segmentations to do the feature extraction and segment classification. We show that both the segment statistics and algorithm performance remain approximately the same when the automatic segmentations are replaced with human segmentations.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Motivation

1.1.1 The Growth of Digital Imaging

People have used images for communication throughout history. Even our ancestors painted pictures on the walls of their caves. However, it was the twentieth century that has witnessed tremendous growth in the number, availability and importance of images. Images play a crucial role in fields as diverse as engineering, medicine, journalism, advertising, design, education and entertainment. Technology, in the form of inventions such as photography and television, has played a major role in facilitating the capture and communication of pictorial data. Notably, the real engine of the imaging revolution has been the computer, introducing a range of techniques for the digital image capture, processing, storage, and transmission. The use of computers in imaging can be dated back to the mid-1960s, although it was rather limited until the mid-1980s when the personal computer was introduced. Once computerized imaging became affordable, it quickly penetrated into almost every area of human activity. The creation of the World-Wide Web in the early 1990s, enabled users to access data in a variety of media from anywhere on the planet, and has provided

further stimulus to the exploitation of digital images.

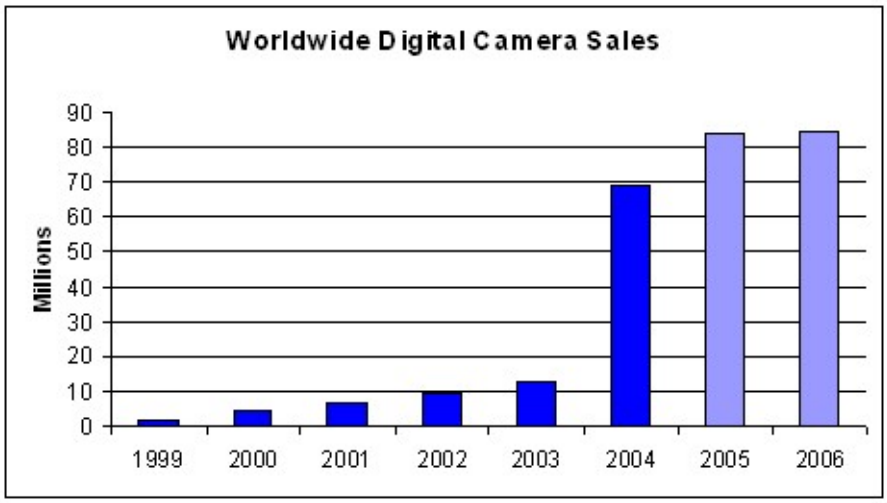


Figure 1.1: Digital Camera Sales

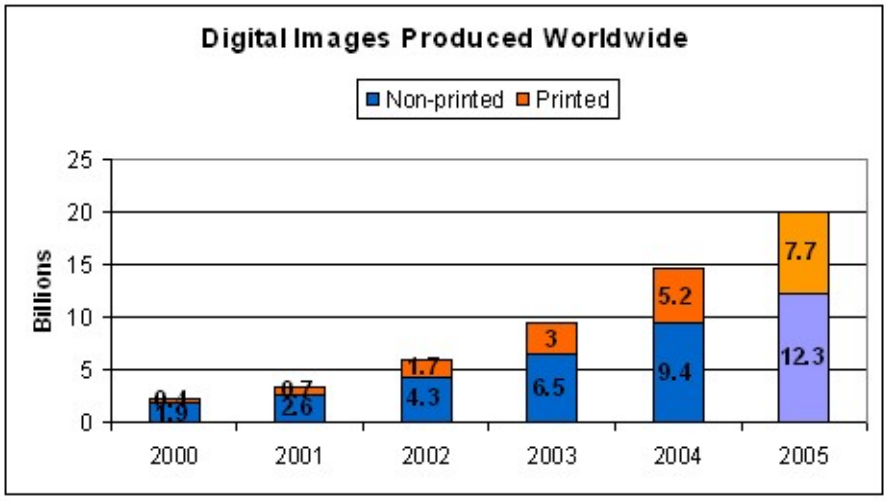


Figure 1.2: Digital Images

The early years of the 21st century have seen an enormous advancement and growth in digital imaging. As digital imagers approach traditional film-based cameras in terms of resolution and price, combined with the additional sets of features they provide, consumers and businesses have embraced this new technology with previously unseen vigor. This has resulted in a tenfold

increase in the amount of digital images created, from 2 billion in year 2000 to estimated 20 billion it 2005 (Source: PMA Marketing Research).

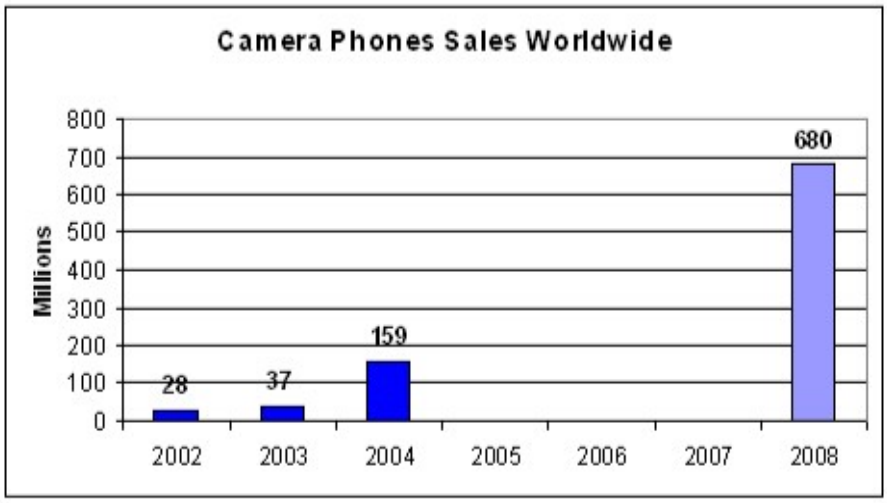


Figure 1.3: Camera Phones Sales

Another important milestone was the introduction of the camera phone. In fact, camera-phones, with 159 million units sold in year 2004 alone (Source: Consumer Phone Report), became the best and fastest selling consumer product in the history of mankind. A recent Zalos Group survey estimates that the total picture messaging market in the United States will approach 440 million dollars by 2008.

1.1.2 The Need for Image Data Management

In addition to consumers, corporate and business users have an increasing need for managing digital image databases. The most important users that might benefit from the Content Based Image Retrieval (CBIR) technology are in the area of publishing and advertising, television and movie studios, fashion and graphic design, museums and galleries, medicine, military and law enforcement, education, geographical and remote sensing systems, and Web search engines. For example: Publishing and advertising companies maintain their own databases of digital images that

might contain millions of images. These industries rely heavily on images to complement a story, illustrate books or articles or promote products and services. In the field of fashion and graphic design, visualization is a part of creative process, and the ability to find a particular combination of color and texture is crucial in the design process [2]. Automatic identification of regions within satellite images by shape, color or texture is another important application that has received lot of attention [3–6]. Currently several Web search engines offer an option of image search. However, they are of limited use as they only search the keywords in the image title or the included metadata.

1.1.3 What is Content Based Image Retrieval?

The term Content Based Image Retrieval was introduced by Kato [7], to describe his experiments into automatic retrieval of images from a database by color and shape features. The term has since been widely used to describe the process of retrieving images from a large collection on the basis of features (such as color, texture, and shape) that can be automatically extracted from the images themselves. The features used for retrieval can be either primitive or semantic, but the extraction process must be predominantly automatic. CBIR differs from classical information retrieval in fact that image databases are essentially unstructured, since digitized images consist purely of arrays of pixel intensities, with no inherent meaning. One of the key issues with any kind of image processing is the need to extract useful information from the raw data (such as recognizing the presence of particular shapes or textures) before any kind of reasoning about the image contents is possible. The image databases thus differ fundamentally from text databases, where the raw material (words stored as ASCII character strings), has already been logically structured by the author [8]. CBIR draws many of its methods from the fields of image processing and computer vision, and is regarded by some as a subset of these fields. Its emphasis is on the retrieval of images with desired characteristics from a large collection. Image processing covers a much wider field, including image enhancement, compression, transmission, and interpretation. Research and devel-

opment issues in CBIR cover a range of topics, many shared with mainstream image processing and information retrieval. Some of the most important are:

- identification of suitable ways of describing image content
- extracting such features from raw images
- matching query and stored images in a way that reflects user needs or human similarity judgments
- efficiently accessing stored images by content
- providing usable human interfaces to CBIR systems

1.2 Proposed Approach - From Segments to Semantics

The rapid accumulation of large collections of digital images has created the need for efficient and intelligent schemes for image retrieval. Since humans are the ultimate users of most retrieval systems, it is important to organize the contents semantically, according to meaningful categories. This requires an understanding of the important semantic categories that humans use for image classification [1, 9, 10], and the extraction of meaningful image features that can be used to distinguish between these categories.

Current algorithms for low-level feature extraction, such as color, texture, and shape, are quite sophisticated and have had a considerable success. [11, 12]. However, the extraction of low-level image features that can be correlated with high-level image semantics remains a challenging task. The focus of this thesis research is on a new methodology for image segmentation, semantic classification, and retrieval, that is based on perceptual models and principles about the processing of texture and color information.

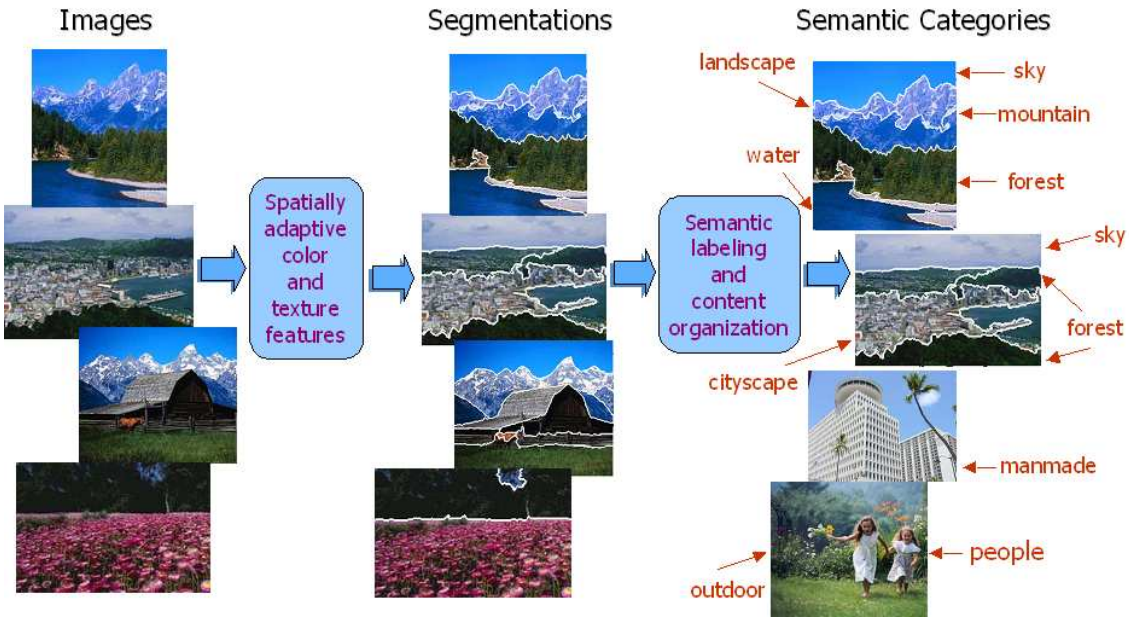
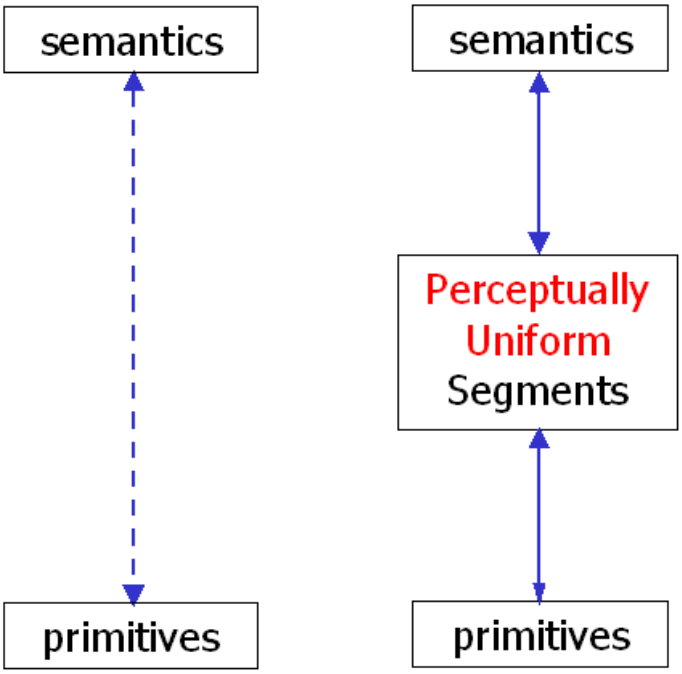


Figure 1.4: Overview of the Proposed Method

This thesis proposes a novel framework for semantic image analysis and retrieval that attempts to bridge the semantic gap between low level primitives and high level semantics by utilizing perceptually uniform segments as medium level descriptors to effectively bridge the semantic gap.

The foundation for this thesis research was set by a recently proposed approach for image segmentation that is based on spatially adaptive color and spatial texture features [13–18]. It is aimed at segmentation of natural scenes, in which color and texture do not typically exhibit uniform statistical characteristics. The new approach combines knowledge of human perception with an understanding of signal characteristics in order to segment natural scenes into perceptually/semantically uniform regions. This new segmentation methodology can be used to extract semantic information from digital images. In particular, it can be used to derive region-wide color and texture features that, together with the segment location, boundary shape, and region size, can be used to extract semantic information.



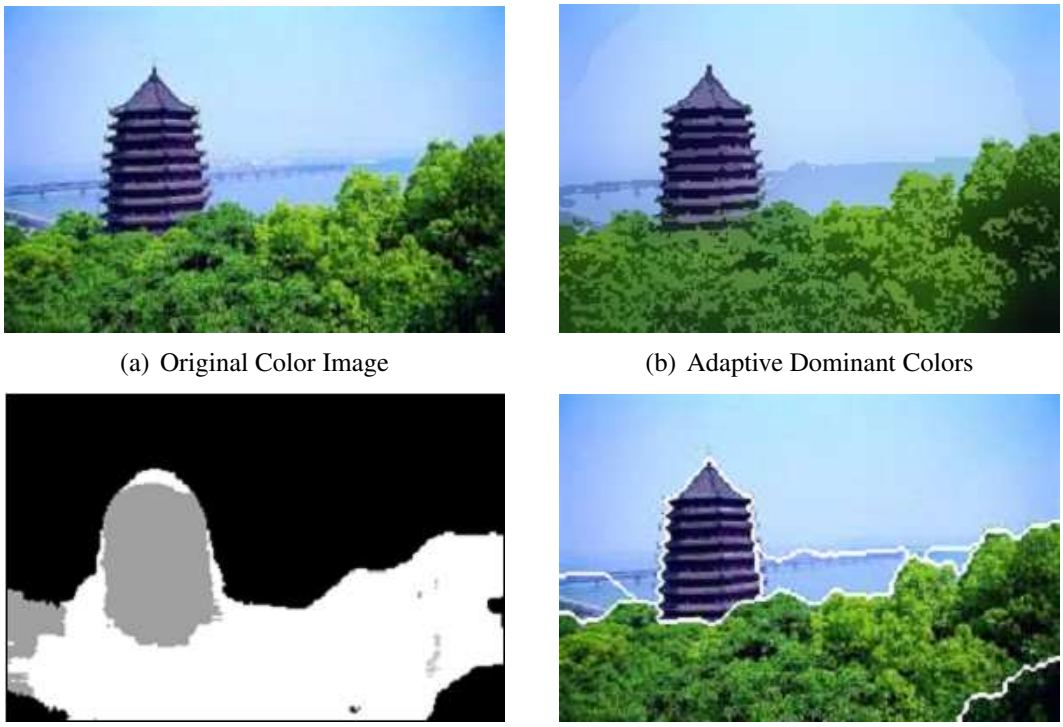
(a) Semantic Gap between low level primitives and image semantics

(b) Proposed method to bridge the Semantic Gap

Figure 1.5: Bridging the Semantic Gap

Once the image has been segmented into regions, the goal is to extract segment features that can be related to semantic concepts. The features that we use to obtain the image segmentation are not necessarily the same as those that are most suitable for assigning a semantic meaning to a segment. Image segmentation requires a combination of local and global features, while region interpretation requires region-wide features. Overall, in this thesis we develop a novel systematic approach for segment labeling that is based on human perception that combines the segment characteristics (color and texture composition, size, shape, relative location). The resulting labels can then be combined to provide an overall scene interpretation.

The focus of this thesis is on still images. The techniques we discuss, however, can also form the basis for content-based analysis of video sequences. We consider the domain of photographic images with an essentially unlimited range of content (landscapes, cityscapes, buildings,



(a) Original Color Image (b) Adaptive Dominant Colors

(c) Texture Classes (d) Final Segmentation

Figure 1.6: Segmentation Overview

indoor scenes, people, plants, animals, objects, etc.).

1.3 Contributions

This dissertation focuses on techniques for semantic labeling on segments for image analysis and classification. The following contributions have been made:

- Semantic classification of image segments.
- Incorporating the knowledge of human perception and signal characteristics into segment feature extraction, representation and classification.
- Evaluation of classification techniques that that are best suited for the selected features.

- Low level color and texture based retrieval.
- Comparison with human performance.
  - human vs. automatic segmentation statistics.
  - human vs. automatic segmentations classification performance.

1.4 Organization

Chapter 2 presents the background material and a high level overview of Content Based Retrieval problem. Chapter 3 presents an overview of perceptually tuned adaptive color-texture segmentation algorithm recently proposed by Chen and Pappas. Chapter 4 introduces the low level features used in retrieval and their representations in greater detail. Chapter 5, describes how proposed features were extracted from segments and their representation. Segment statistics of dominant colors is also presented in this chapter. In Chapter 6 we present semantic label selection, verification database and ground truth generation were. Chapter 7 presents clustering, classification and pattern recognition algorithms used in classification, their tradeoffs and numerical implementations. Chapter 8 presents experimental results and detailed analysis demonstrating benefits of this new approach. Feature Evaluation in Terms of Human Segmentations and statistics of human segmentations are presented in Chapter 9. Summary, conclusions and future research directions are proposed in Chapter 10.

Chapter 2

Background

2.1 User Needs for Image Data Indexing

CBIR research reports, which have focused on specific collections, user types or populations [19–26], reveal that there is a significant difference between features sought by users in different disciplines. For example, journalists prefer to undertake their own searches for photographs based on such attributes as symbolic value, atmosphere and feelings, or the context of a particular illustration task [20], while health professionals ask for images in a manner more in keeping with the Library's orientation (e.g., do you have pictures of cholera?). Such information leads to the conclusion that CBIR systems have to be designed with particular applications in mind. In addition, users' expressed needs are likely to be heavily biased by their expectations of the kinds of query the system can actually handle [19]. To resolve this problem, van der Starre [27] advocates that indexers should "stick to 'plain and simple' indexing, using index terms accepted by the users, and using preferably a thesaurus with many lead-ins," thus placing the burden of further selection on the user. Shatford Layne [28] suggests that, when indexing images, it may be necessary to determine which attributes provide useful groupings of images; which attributes provide information that is useful once the images are found; and which attributes may, or even should, be left

to the searcher or researcher to identify. In addition, she advocates further research into the ways images are sought and the reasons that they are useful in order to improve the indexing process. Constantopulos and Doerr [29] also support a user centered approach to the designing of effective image retrieval systems. They urge that the attention needs to be paid to the intentions and goals of the users, since this will help define the desirable descriptive structures and retrieval mechanisms as well as understanding what is 'out of the scope' of an indexing system.

The last decade has seen the appearance of a number of commercial databases and image data management systems such as: Corbis, Getty, iBase, Index+, Digital Catalogue, Fastfoto, FotoWare, Signpost, Cumulus. These systems store representations of pictorial documents (such as photographs, prints, paintings, drawings, illustrations, slides, video clips, and so on) in static archival databases, and incorporate multimedia database management systems in the storage of, and provision of wider access to, these repositories. It should be noted, however, that none of these systems provide CBIR facilities - all rely on text keywords which have to be entered by human indexers to provide retrieval of stored images.

2.1.1 Research Into Indexing Effectiveness

There is a wide range of available text retrieval software to automate the actual process of searching. However, the process of manual indexing, whether by keywords or classification codes, suffers from two significant drawbacks. First, it is inherently very labor-intensive. Indexing times quoted in the literature range from about 7 minutes per image for stock photographs at Getty Images, using their in-house system, to more than 40 minutes per image for a slide collection at Rensselaer Polytechnic. Manual indexing times for video are likely to be even longer. Both newspapers and stock shot agencies maintain archives of still photographs to illustrate articles or advertising copies. These archives can often be extremely large (running into millions of images) and dauntingly expensive to maintain if detailed keyword indexing is needed. The broadcasting corporations are

faced with an even bigger problem, having to deal with millions of hours of archived video footage, which is almost impossible to annotate without some degree of automatic assistance. Secondly, manual indexing does not appear to be particularly reliable as a means of subjective retrieval of images. In [30] Markey reported that there were wide disparities in the keywords that different individuals assigned to the same image. Similar results were reported from studies of the usefulness of assigned keywords in answering user queries in picture libraries. These limitations mean that retrieval of images has to rely on the knowledge and experience of the staff. At the present stage of CBIR development, it is meaningless to ask whether CBIR techniques perform better or worse than manual indexing. Potentially, CBIR techniques have a number of advantages over manual indexing. They are inherently quicker, cheaper, and completely objective in their operation. Another limitation of manual indexing is the difficulty to anticipate the retrieval cues future searchers will actually use [22]. As observed above, in contrast with the situation in text retrieval, where index language effectiveness has been the subject of intensive study for more than thirty years, there is little hard evidence on the effectiveness of visual information retrieval systems of any kind.

2.2 Characteristics of Image Queries

Image retrieval types have been classified into three levels of increasing complexity by Eakins [31]:

Level 1: The retrieval by primitive features such as color, texture, shape or the spatial location of image elements. Examples of such queries might include “find pictures with a yellow disc the at top”, or most commonly “find images similar to this one”. This level of retrieval uses features (such as a particular hue and lightness of yellow) that are both objective and directly derivable from the images themselves. Its use is largely limited to specialized applications such as trademark registration, identification of drawings in a design archive, or color matching of fashion accessories.

Level 2: The retrieval by derived, sometimes known as logical features, involving some degree of logical inference about the identity of the objects depicted in the image. It can conveniently be further divided into: retrieval of objects of a given type (e.g. “find images of a SUV”); retrieval of individual objects or persons (“find a picture of the Chicago Skyline”). In the first example above, some prior understanding is necessary to identify an object as a SUV which is a type of car; in the second example, one needs the knowledge that a given individual structure has been given the name “the Eiffel tower”. Search criteria at this level, are usually still reasonably objective.

Level 3: The retrieval by abstract attributes, involving a significant amount of high-level reasoning about the meaning and purpose of the objects or scenes depicted. Again, this level of retrieval can be subdivided into: retrieval of named events or types of activity (e.g. “find pictures of Scottish folk dancing”); retrieval of pictures with emotional or religious significance (“find a picture depicting suffering”).

At present the most significant gap lies between levels 1 and 2. Many authors [32] refer to levels 2 and 3 together as semantic image retrieval, and hence the gap between levels 1 and 2 as ***semantic gap***. It should be noted that Eakins’ classification ignores the retrieval by associated metadata, such as who created the image, where, and when.

2.3 Content-Based Image and Video Retrieval

2.3.1 Primitive Features Used in Retrieval

The most common primitive features used in image classification and retrieval are color, texture and shape and are used by all current CBIR systems. A typical system allows users to formulate queries by submitting an example of the type of image being sought, though some offer alternatives such as selection from a palette or sketch input. The system then identifies those stored images whose feature values most closely match those of the query, and displays thumbnails of these

images on the screen. Some of the more commonly used types of features used for image retrieval are described below.

2.3.2 Color Retrieval

Several methods for retrieving images on the basis of color similarity have been described in the literature, but most are variations of the same basic idea. Each image added to the collection is analyzed to compute a color histogram which shows the proportion of pixels of each color within the image. The color histogram for each image is then stored in the database. During search, the user can either specify the desired proportion of each color (75% olive green and 25% red, for example), or submit an example image from which a color histogram is calculated. Either way, the matching process then retrieves those images whose color histograms most closely match those of the query. The matching technique most commonly used, histogram intersection, was first developed by Swain and Ballard [1991]. Variants of this technique are now used in a high proportion of current CBIR systems. Methods that improve the Swain and Ballard's original technique include the use of cumulative color histograms [33], combining histogram intersection with some element of spatial matching [34], and the use of region-based color querying [35].

2.3.3 Texture Retrieval

The ability to match texture can often be useful in distinguishing between areas of images with similar color (such as sky and sea, or leaves and grass). A variety of techniques has been used for measuring texture similarity; the best-established ones rely on comparison of values of second-order statistics calculated from query and stored images. Essentially, methods calculate the relative brightness of selected pairs of pixels from each image. Using these descriptors, it is possible to calculate measures of image texture such as the degree of contrast, coarseness, directionality and regularity [36], or periodicity, directionality and randomness [37]. The alternative methods of

texture analysis for retrieval include the use of Gabor filters [38] and fractals [39]. Texture queries can be formulated in a similar manner to color queries, by selecting examples of desired textures from a palette, or by supplying an example query image. The system then retrieves images with texture measures most similar in value to the query. An extension of the technique is the texture thesaurus developed by Ma and Manjunath [5], which retrieves textured regions in images on the basis of similarity to automatically-derived codewords representing important classes of texture within the collection.

2.3.4 Shape Retrieval

Unlike texture, shape is a fairly well-defined concept - and there is considerable evidence that natural objects are primarily recognized by their shape [40]. A number of features characteristic of object shape (but independent of size or orientation) are computed for every object identified within each stored image. Queries are then answered by computing the same set of features for the query image, and retrieving those stored images whose features most closely match those of the query. Two main types of shape features are commonly used - global features such as aspect ratio, circularity and moment invariants [41] and local features such as sets of consecutive boundary segments [42]. Alternative methods proposed for shape matching have included elastic deformation of templates [43, 44], comparison of directional histograms of edges extracted from the image [45, 46], and shocks, skeletal representations of object shape that can be compared using graph matching techniques. The queries to shape retrieval systems are formulated either by identifying an example image to act as the query, or as a user-drawn sketch [47, 48]. The shape matching of three-dimensional objects is a more challenging task - particularly where only a single 2-D view of the object in question is available. While no general solution to this problem is possible, some useful inroads have been made into the problem of identifying at least some instances of a given object from different viewpoints. One approach has been to build up a set

of plausible 3-D models from the available 2-D image, and match them with other models in the database [49]. Another approach is to generate a series of alternative 2-D views of each database object, each of which is matched with the query image [50].

2.3.5 Retrieval by Other Types of Primitive Features

One of the oldest-established means of accessing pictorial data is retrieval by its position within an image. Accessing data by spatial location is an essential aspect of geographical information systems and efficient methods to achieve this have been around for many years [51, 52]. Similar techniques have been applied to image collections, allowing users to search for images containing objects in defined spatial relationships with each other [53, 54]. Improved algorithms for spatial retrieval are still being proposed [55]. Spatial indexing is seldom useful on its own, though it has proved effective in combination with other cues such as color [34, 56] and shape [57].

Several other types of image features have been proposed as a basis for CBIR. Most of these rely on complex transformations of pixel intensities that have no obvious counterpart in any human description of an image. Most such techniques aim to extract features that reflect some aspect of image similarity, which a human subject can perceive, even if he or she finds it difficult to describe. The most well-researched technique of this kind uses the wavelet transform to model an image at several different resolutions. Promising retrieval results have been reported by matching wavelet features computed from query and stored images [58, 59]. Another method giving interesting results is retrieval by appearance. Two versions of this method have been developed, one for whole-image matching and one for matching selected parts of an image. The part-image technique involves filtering the image with Gaussian derivatives at multiple scales [60], and then computing differential invariants; the whole-image technique uses distributions of local curvature and phase [61]. The advantage of all these techniques is that they can describe an image at varying levels of detail (useful in natural scenes where the objects of interest may appear in a variety of guises),

and avoid the need to segment the image into regions of interest before shape descriptors can be computed. This is because, despite recent advances in techniques for image segmentation, it remains a challenging problem.

2.3.6 Retrieval by Semantic Image Features

The vast majority of current CBIR techniques are designed for primitive-level retrieval. However, some researchers have attempted to bridge the gap between level 1 and level 2 retrieval. One early system aimed at tackling this problem was GRIM-DBMS [62], designed to interpret and retrieve the line drawings of objects within a narrow predefined domain, such as floor plans for domestic buildings. The system analyzed object drawings, labeling each with a set of possible interpretations and their probabilities. These were then used to derive likely interpretations of the scene within which they appeared.

More recent research reports tend to concentrate on two problems. The first is scene recognition. It can often be important to identify the overall scene type depicted by an image, both because this in an important filter that can be used when searching, and because this can help in identifying specific objects present. One system of this type is IRIS [63], which uses color, texture, region and spatial information to derive the most likely interpretation of the scene, generating text descriptors which can be input to any text retrieval system. Other researchers have identified simpler techniques for scene analysis, using low-frequency image components to train a neural network [64], or color neighborhood information extracted from low-resolution images to construct user-defined templates [65]. The second focus of research activity is object recognition, an area of interest to the computer vision community for many years [66–68]. Techniques are now being developed for recognizing and classifying objects with database retrieval in mind. Such techniques are based on the idea of developing a model for each object class to be recognized, identifying image regions that might contain examples of the object, and building up evidence to

confirm or rule out the object's presence. The evidence typically includes both features of the candidate region itself (color, shape or texture) and contextual information such as its position and the type of background in the image. In contrast to these fully-automatic methods, there is a family of techniques that allow systems to learn associations between semantic concepts and primitive features from user feedback. The earliest such system was FourEyes from MIT [69]. This approach invites the user to annotate selected regions of an image, and then proceeds to apply similar semantic labels to areas with similar characteristics. The system is capable of improving its performance with further user feedback. Another approach is the concept of the semantic visual template introduced by Chang et al [70]. Here, the user is asked to identify a possible range of color, texture, shape or motion parameters to express his or her query, which is then refined using relevance feedback techniques. When the user is satisfied, the query is given a semantic label (such as "sunset") and stored in a query database for later use. Over time, this query database becomes a visual thesaurus, linking each semantic concept to the range of primitive image features most likely to retrieve relevant items.

2.4 Available CBIR Software

2.4.1 Commercial Systems

Despite the shortcomings of current CBIR technology, several image retrieval systems are now available as commercial packages, with demonstration versions of many others available on the Web. Some of the most prominent systems are described below.

Pixlogic: Pixlogic is a start-up company that has recently introduced a visual search engine that "automatically analyzes, indexes, and searches the contents of images and video files." The user can input either the image, or a selection, as an input to query for similar images. This system segments the image into regions and then uses color and shape descriptors to classify the objects

in the image such as a car, person, logo, etc. This system is still unable to assign semantic labels.

QBIC: IBM's QBIC system [71] is probably the best-known of all image content retrieval systems. It is available commercially either in standalone form, or as part of other IBM products such as the DB2 Digital Library. It offers retrieval by any combination of color, texture or shape - as well as by text keyword. The system extracts and stores color, shape and texture features from each image and calculates a similarity score between the query and each stored image.

Virage: Another well-known commercial system is the VIR Image Engine from Virage, Inc [72]. A high-profile application of Virage technology is AltaVista's AV Photo Finder, allowing Web surfers to search for images by content similarity. Virage technology has also been extended to the management of video data.

Excalibur: This product offers a variety of image indexing and matching techniques based on the company's own proprietary pattern recognition technology [73]. Its best-known application is probably the "Yahoo!" Image Surfer, allowing content-based retrieval of images from the World-wide Web.

2.4.2 Experimental Systems

A large number of experimental systems have been developed, mainly by academic institutions, in order to demonstrate the feasibility of new CBR techniques. Many of these are available as demonstration versions on the Web. Some of the best-known are described below.

Photobook: The Photobook system [43] from Massachusetts Institute of Technology (MIT) has proved to be one of the most influential of the early CBIR systems. Photobook characterizes images for retrieval by computing shape, texture and other appropriate features. This system has been successfully used in a number of applications, involving retrieval of image textures, shapes, and human faces, each using features based on a different model of the image. More recent versions of the system allow users to select the most appropriate feature type for the retrieval problem

at hand from a wide range of alternatives [74]. Although Photobook itself never became a commercial product, its face recognition technology has been incorporated into the FaceID package from Viisage Technology which is used by several US police departments.

Cypress: The Cypress CBR system is incorporated within the Berkeley Digital Library project.

VisualSEEk: The VisualSEEk system [56] was developed at Columbia University, New York. It offers searching by image region color, shape and spatial location, as well as by keyword. Users can build up image queries by specifying areas of defined shape and color at absolute or relative locations within the image.

WebSEEk: The WebSEEk system [75] aims to facilitate image searching on the Web. Web images are identified and indexed by an autonomous agent, which assigns them to an appropriate subject category according to associated text. Color histograms are also computed from each image. During search, users are invited to select categories of interest; the system then displays a selection of images within this category, which users can then search by color similarity. Relevance feedback facilities are also provided for search refinement. Further prototypes from this group include VideoQ [76], a video search engine allowing users to specify motion queries, and MetaSEEk [77], a meta-search engine for images on the Web.

MARS: The MARS (Multimedia Analysis and Retrieval System) project has been developed at the University of Illinois [78]. Relevance feedback is an integral part of the system, and according to the authors is the only way of capturing individual human similarity judgments at the present time. The system characterizes each object within an image by a variety of features, and uses a range of different similarity measures to compare the query and stored objects. User feedback is then used to adjust feature weights, and if necessary to invoke different similarity measures [79].

Infomedia: The Infomedia project [80] aims to facilitate full content search and retrieval of video by integrating speech and image processing. This system identifies video scenes (not just

shots) from analysis of color histograms, motion vectors, speech and audio soundtracks, and then automatically indexes these 'video paragraphs' according to the significant words detected from the soundtrack, text from the images and captions, and objects detected within the video clips. A query is typically submitted as speech input. Thumbnails of keyframes are then displayed with the option to show a sentence describing the content of each shot, extracted from spoken dialogue or captions, or to play back the shot itself. Many of the system's strengths stem from its extensive evaluation with a range of different user populations [81]. Its potential applications include TV news archiving, sports, entertainment and other consumer videos, and education and training.

Surfimage: The Surfimage system was developed at INRIA, France [82]. This system has a similar philosophy as the MARS system, using multiple types of image features that can be combined in different ways, and offering relevance feedback.

Netra: The Netra system uses color texture, shape, and spatial location information to provide region-based searching based on local image properties [83] utilizing an image segmentation technique.

Synapse: This system is an implementation of retrieval by appearance using whole image matching [61].

2.5 Current research trends

Most of current CBR approaches utilize an image segmentation scheme as an intermediate step, and then rely on the content of the segmented regions as well as their context within an image to obtain semantic information. Some of them use an explicit image segmentation where image is decomposed in a regular grid.

Mojsilovic and Rogowitz [10] attempt to link low-level image features directly to image semantics, while Zhu *et al.* [84] partition the image into equal size blocks and index the regions using a codebook whose entries are obtained from the block features. Wang *et al.* [85] also propose

a codebook based approach, whereby the codebook is used to segment the image based on the statistics of the region color and texture features. Their approach also attempts to take into account properties of the neighboring regions. Carson *et al.* [86] use a simple segmentation technique to segment an image into regions and extract their features. Each region is given a label called a blob-token. The authors attempt to find the association (co-occurrence) among the blob-tokens and the associated captions to index the image. Barnard, Forsyth *et al.* [87] extend blob idea even further by learning the joint distributions of image regions and associated words. Their approach is an example of multimodal data mining in which there is an attempt to effectively *match words and pictures*. Li and Wang [88] use a statistical modeling approach in which images of a given concept are regarded as the instances of a random process characterizing this concept. Their method utilizes 2D hidden Markov models to calculate a measure of association between the image and the textual description of a concept. Gao et. al. [89] address the framework for indexing and retrieval of images based on high-dimension feature extraction and discriminative classifier learning. This method attempts to describe the image content, using keywords from a predefined vocabulary, based on low-level features. To avoid errors associated with image segmentation proposed method performs the explicit segmentation by employing regular blocking and tokenization of all sub-blocks of training images. Then single and bigrams of neighboring blocks are computed and high dimensional Latent Semantic Analysis (LSA) matrix assembled. Image annotation is then treated as a multi-class text categorization, i.e., assigning multiple ranked labels to an image according to its closeness to concept models. In the method proposed by Feng et. al. [90], each image is partitioned into a set of rectangular regions and a real-valued feature vector is computed over these regions. The relevance model, which is a joint probability distribution of the word annotations and the image feature vectors, is computed using the training set. The word probabilities are estimated using a Multiple Bernoulli Relevance Model (MBRM), and the image feature probabilities using a non-parametric kernel density estimate. Images are then annotated using this model. The authors evaluated their algorithm using 260 concepts with a total of COREL 5000 images, where 4000

images were used for training, 500 for a validation set, and 500 for testing. Experimental results obtained 24 percent average precision and 25 percent average recall.

The above mentioned approaches have achieved some success for certain image types and certain semantic categories but, in spite of all this effort, the effectiveness of CBIR systems has not been satisfactory and they are still a long way from matching the performance of the human visual system (HVS).

A Major obstacle for the success of CBR systems, as pointed out by the authors in [91], is the unavailability of semantically meaningful image segmentation.

The volume of the research activity associated with CBIR techniques and systems continues to grow. Significant research problems addressed are: methods of segmenting images to distinguish objects of interest from their background (or alternatively, improved techniques for feature extraction which do not rely on segmentation), new paradigms for user interaction with CBIR systems, and better ways of representing human judgments of image similarity. Above all, there is a need to bridge the semantic gap, introducing a degree of automation to the processes of indexing and retrieving images.

Chapter 3

Semantic Labeling

The identification of appropriate semantic categories is of critical importance in the design of a CBR system, as it will ensure both the relevance and the feasibility of the solutions sought.

3.1 Selecting Semantic Categories

The goal of this thesis is to classify segments into semantic categories according to their features. However, these semantic categories have to be selected according to the manner in which humans classify images.

Recent perceptual experiments by Mojsilovic and Rogowitz [1] suggest a semantically based image similarity and retrieval model, and identified semantic categories that humans use for image classification. In their experiments, Mojsilovic and Rogowitz, uncovered the two major axes with the most fundamental categories: natural vs. man made axis, and more human-like vs. less human like axis. These axes could be conveniently named animate and inanimate, and they are adopted as the fundamental categories in our classification.

In this thesis, we propose a hierarchical classification that extends these abstract categories into a more specific ones. For example, the “natural” category can be divided into “vegetation,”

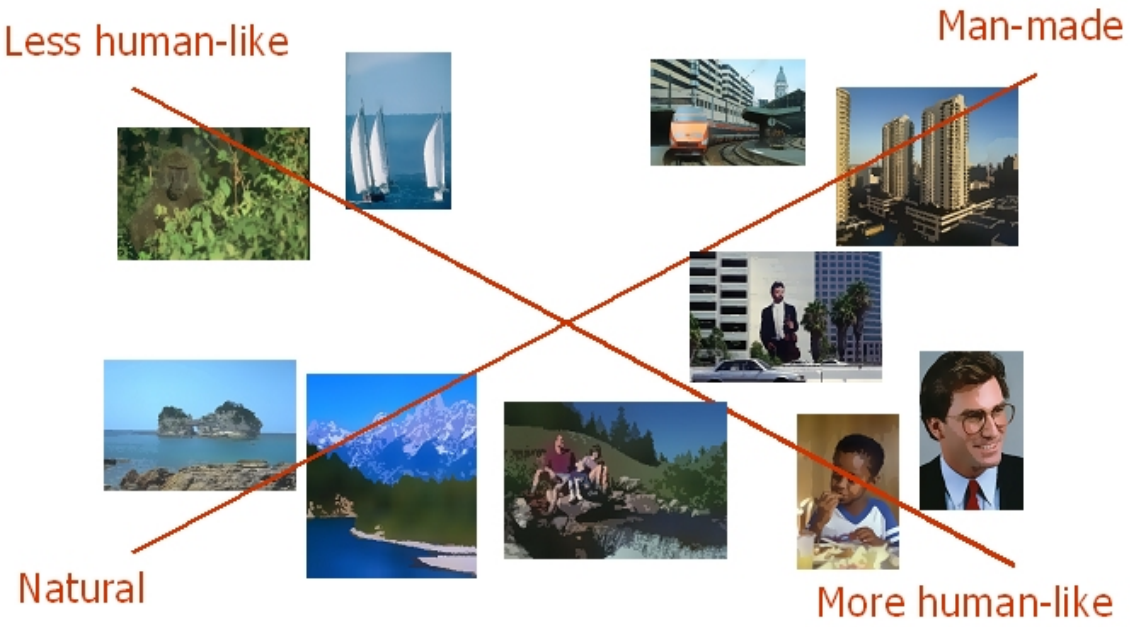


Figure 3.1: Fundamental semantic categories humans use in judging image similarity [1].

"sky," "water," "landform," etc., and the "vegetation" subcategory can be further divided into "grass," "forest," and "woods/bushes" categories. Continuing in such a manner, we would be able to construct a vocabulary of labels which can be associated with a particular segment. The primary reason for such a hierarchical vocabulary representation is that sometimes we will not be able to classify the segment into any of the more specific categories, however we still might be able classify it into a more abstract one.

Our experiments using unsupervised hierarchical clustering based on texture features only, presented us with promising results. We were able to discriminate between several categories. However, these experiments also revealed importance of color composition features. Our main research is focused at combining the color and texture feature descriptors.

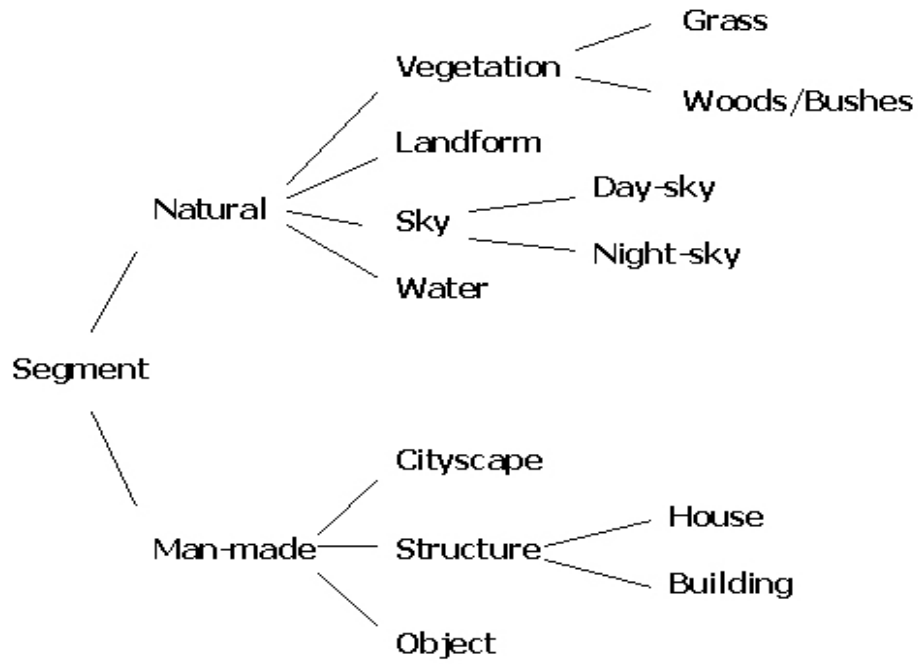


Figure 3.2: Hierarchical category representation - example.

3.2 Color Labeling

In addition to the semantic labels, we also want to attach a color name label to each segment/category. This color label would allow, for example, the users to refine their query by specifying a particular color within the semantic category, or to search directly by color composition. Thus, as we will see below, color can be used both directly and as an intermediate feature for extracting semantic labels.

The mechanism of color naming is still not very well understood. Due to the limited research efforts invested in this area by the engineering community, as a starting point, we have to rely on the research done in the field of neurophysiology [92] [93] Another drawback is the non-existence of color space suitable for the CBR applications. All of the existing color spaces were designed with intervene gamut matching or compression applications in mind. Even the CIE

L*a*b* and L*u*v* color spaces, that were designed with perceptual uniformity in mind are only locally perceptually uniform.

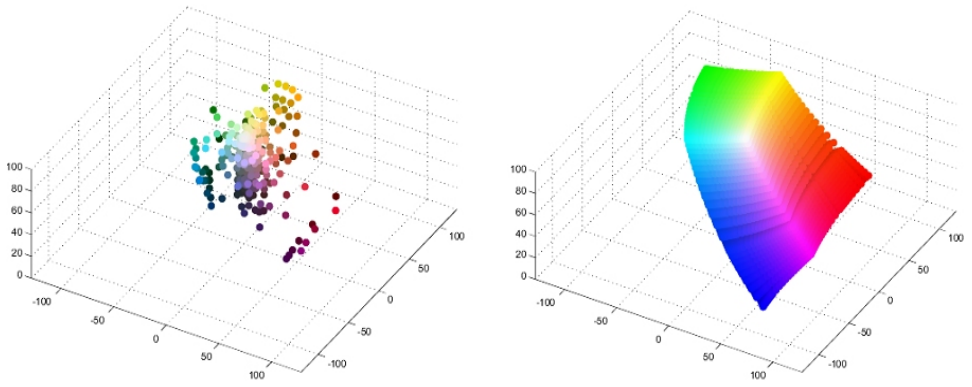
One of the most important studies of color categorization was done by Berlin and Kay [94], who examined about one hundred languages and uncovered similarities in the color vocabulary. They introduced a concept of basic color term and identified the following eleven basic colors: black, white, red, green, yellow, blue, brown, pink, orange, purple and gray. Later studies confirmed this hypothesis and indicated that prototypical colors play a crucial role in the internal representation of color categories, and that a membership in the color category seems to be represented relative to the prototype [95].

3.2.1 Proposed Approach

One possible approach for segment color labeling is to utilize National Bureau of Standards - NBS recommendation for color names, which defines 267 focal colors and associated names [96]. A sample color could then be compared to all 267 focal colors and the name of the closest foci assigned. Through perceptual experiments, Mojsilovic in [97] showed that in many cases this approach failed to match the color names assigned by subjects. Thus, she proposed a new computational model for color categorization and naming. Although it outperforms the simple closest focal color approach, this method is computationally intensive requiring several iterations for each of the foci.

As presented in Figure 3.3 (a), the shortcoming of the NBS focal colors is a poor sampling of the color space. A more effective way of assigning a color name to a segment would be to repeat the perceptual experiments in order to obtain the set of additional focal points (allowing several points to have the same name). Using this method, it could be possible to construct a simple lookup table to decrease the computation time.

Table 3.2.1 illustrates a color naming syntax proposed by the NBS, which utilizes English



(a) (b)

Figure 3.3: (a) NBS focal colors in L*a*b* color space (b) RGB gamut in L*a*b*.

terms to describe colors along dimensions of hue (primary and secondary), saturation and lightness.

Hue - primary	Hue - secondary	Saturation	Lightness	Achromatic
red	red	grayish	blackish	black
orange	brown	moderate	very dark	gray
brown	yellow	medium	dark	white
yellow	green	strong	medium	
green	blue	vivid	light	
blue	purple		very light	
purple	pink		whitish	
pink				
beige				
olive				
violet				

Table 3.1: Color naming syntax

Chapter 4

Segmentation

In this chapter, we review the perceptually-tuned multiscale color-texture segmentation developed by Chen and Pappas [18] The results of this algorithm form the basis of the thesis research. The purpose of this thesis will be to relate the image segments to semantic categories used in image labeling and retrieval.

Many Content-Based Image Retrieval (CBIR) systems rely on scene segmentation for retrieval [11] [12]. The segmentation of natural images is particularly difficult because, textures that appear uniform to the human eye exhibit non-uniform statistical characteristics due to effects of lighting, perspective, etc. Thus, the problem of combining spatial texture and color to obtain segmentations that are consistent with human perception is quite challenging. The key to addressing this problem is in combining perceptual models and principles about the processing of texture and color information with an understanding of image characteristics. Although significant efforts have been devoted to understanding perceptual issues in image analysis(*e.g.,* [98] [99]), relatively little work has been done in applying perceptual principles to complex scene segmentation(*e.g.,* [100]).

In [13] [14], Chen and Pappas presented an image segmentation algorithm that is based on spatially adaptive color and spatial texture features. The perceptual aspects of this algorithm were further developed in [15] [18], while in [16] authors proposed perceptual tuning of the algorithm

based on subjective tests.

4.1 Adaptive Perceptual Color-Texture Segmentation

The flow chart of the algorithm is shown in Fig. 4.1. The algorithm is based on two types of spatially adaptive features. One describes the local color composition, and the other the spatial characteristics of the grayscale component of the texture. These features are first developed independently, and then combined to obtain the overall segmentation.

The color composition features consist of the (spatially adaptive) dominant colors and associated percentages in the vicinity of each pixel. The use of spatially adaptive dominant colors reflects, on the one hand, the fact that the human visual system (HVS) cannot simultaneously perceive a large number of colors, and on the other, the fact that image colors are spatially varying. The spatially adaptive dominant colors are obtained using the adaptive clustering algorithm (ACA) for segmentation [101]. The color feature representation is as follows:

$$f_c(x,y,N_{x,y}) = \{(c_i(x,y,N_{x,y}), p_i(x,y,N_{x,y})), \\ i = 1,\ldots,M, p_i(x,y,N_{x,y}) \in [0,1]\} \tag{4.1}$$

where each of the dominant colors, $c_i(x,y,N_{x,y})$, is a three dimensional vector in *Lab* space and $p_i(x,y,N_{x,y})$ is the corresponding percentage. $N_{x,y}$ denotes the neighborhood around the pixel at location (x,y) and M is the total number of colors in the neighborhood. A reasonable choice is $M = 4$. Finally, a perceptual metric (OCCD) [97] is used to determine the similarity of two color feature vectors.

The spatial texture features describe the spatial characteristics of the grayscale component of the texture, and are based on a multiscale frequency decomposition such as the steerable pyramid [102] or the Gabor transform [103]. Such decompositions have been widely used as descriptions

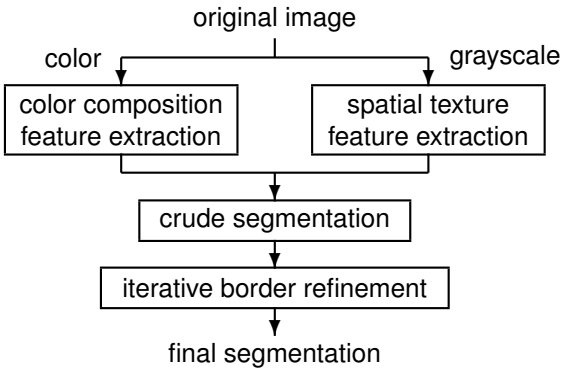


Figure 4.1: Schematic of segmentation algorithm

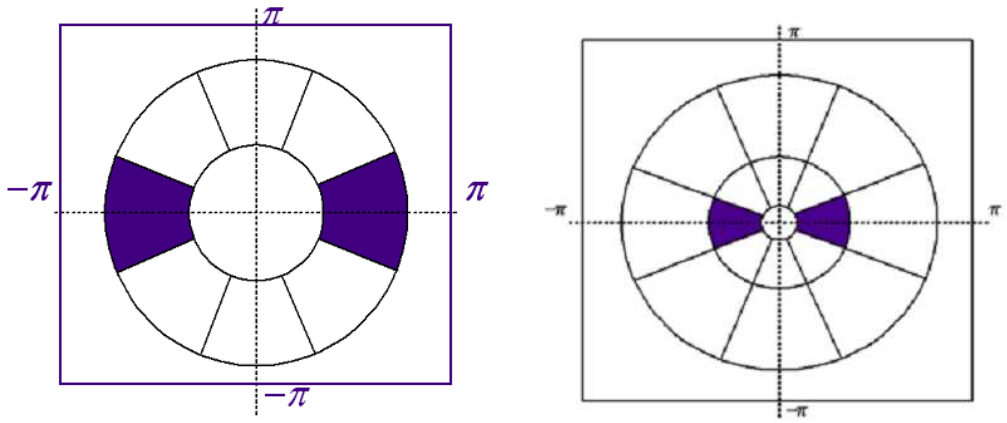
of early visual processing in mammals. The local median energy of the subband coefficients is a simple but effective characterization of spatial texture. Median operators tend to respond to texture within uniform regions and suppress responses associated with transitions between regions. In [14] Chen et al. used a one-level steerable filter decomposition with four orientations, as shown in Fig. 4.2 (left). The texture features consist of a classification of each pixel into one of the following categories: *smooth, horizontal, vertical, +45 -45* and *complex.*

The spatial texture feature extraction consists of two steps. First, pixels are classified into smooth and non-smooth categories. Then the non-smooth pixels are further classified into the remaining categories. Let $s_0(x,y)$, $s_1(x,y)$, $s_2(x,y)$, and $s_3(x,y)$ represent the subband coefficient at location (x,y) that corresponds to the horizontal, +45 vertical, and -45lope directions, respectively,

The $s_{\max}(x,y)$ is used to denote the maximum absolute value of the four coefficients, and $s_i(x,y)$ to denote the subband index that corresponds to that maximum. A pixel (x,y) is classified as smooth if the median of $s_{\max}(x',y')$ over a neighborhood of (x,y) is below a threshold T_0. In [15] this threshold was determined using a two-level K-means over the image. As shown in [16], this threshold can be determined by subjective tests. If the pixel is non-smooth, then it

is further classified as follows. We compute the percentage for each value (orientation) of the index $s_i(x', y')$ in the neighborhood of (x, y). If the maximum of the percentages is higher than a threshold T_1 (*e.g.,* 42%) and the difference between the first and second maxima is greater than a threshold T_2, (*e.g.,* 12%), then there is a dominant orientation in the window and the pixel is classified accordingly. Otherwise, the pixel is classified as complex. The first threshold ensures the existence of a dominant orientation and the second ensures its uniqueness. These thresholds were determined by subjective tests. The use of maximum is due to the fact that neighboring subband filters typically have significant overlap (*e.g.,* in the steerable filter decomposition) and the maximum carries significant information about the texture orientation.

The segmentation algorithm combines the color composition and spatial texture features to obtain segments of uniform color texture. This is done in two steps. The first relies on a multigrid region growing algorithm to obtain a crude segmentation. The segmentation is crude due to the fact that the estimation of the spatial and color texture features requires a finite window. The second uses an elaborate border refinement procedure, which progressively relies on the color composition features to obtain accurate and precise border localization.



(a) One-level Decomposition (b) Two-level Decomposition

Figure 4.2: Steerable Filter Frequency Response

4.1.1 Multiscale Feature Extraction

Since the HVS can perceive multiple scales at the same time, it is important that a computer-based segmentation algorithm be able to detect textures at different scales. Multiple scale analysis can also help capture an object in different perspectives as one uniform object.

A texture may be smooth at a finer scale, horizontal at a coarser scale, and smooth again at an even coarser scale. In such a case, the texture will be perceived as horizontal. Alternatively, if a texture is horizontal at one scale and vertical at another, then a human could detect both orientations. In such a case, it would make more sense to classify the texture as complex (given the above texture categories). Finally, if a texture is complex at one scale and horizontal at another, the horizontal orientation is more likely to dominate the human perception.

Based on the these observations, Chen et al. utilized the following rules for extending the one-level texture feature extraction method to multiple scales:

1. For each scale, use the texture extraction method described in the previous section.
2. If downsampling is performed in the multiscale decomposition, upsample the texture class images obtained at each scale to the original image size, so that the texture class images from all scales have the same size.
3. Combine the texture classes of different scales using the following rules:

- A pixel is classified as smooth only if it is classified as smooth at all of the scales.
- A pixel is classified as horizontal, vertical, +45 or -45 if all the scales are consistent, where classification in any given direction at one scale is consistent with a complex or smooth classification at another scale, but is not consistent with a classification in any other direction at another scale. Due to the crudeness of the texture classification, neighboring directions are also considered as consistent with each other.
- Pixels that do not satisfy the above conditions are classified as complex.

Thus, the complex category includes pixels that are classified as complex at some scales and smooth at the remaining scales, or pixels that have inconsistent classification at different scales.

4.1.2 Perceptual Tuning

Several key parameters of the segmentation algorithm were determined by subjective tests [17]. These include the threshold T_0 for the smooth/non-smooth classification and the thresholds necessary for determining if there is a dominant orientation (T_1 and T_2). Another important parameter is the threshold for the color composition feature similarity. The goal of the tests is to relate human perception of isolated (context-free) texture patches to the statistics of natural textures.

The parameter selections were based on a combination of texture statistics and how humans perceive textures. For more details on the subjective experiments,[1] refer to [16] [17]. Experimental results demonstrate that this perceptual tuning leads to significant improvements in segmentation performance.

4.2 Segmentation Results

Fig. 4.3 shows the segmentation results based on the algorithm described in this chapter. In all cases, the texture window size was 23×23 and spatial constraints beta was $\beta = 0.8$. More details can be found in [104].

[1] Available online at *http://peacock.ece.utk.edu/FeatureTest/*.

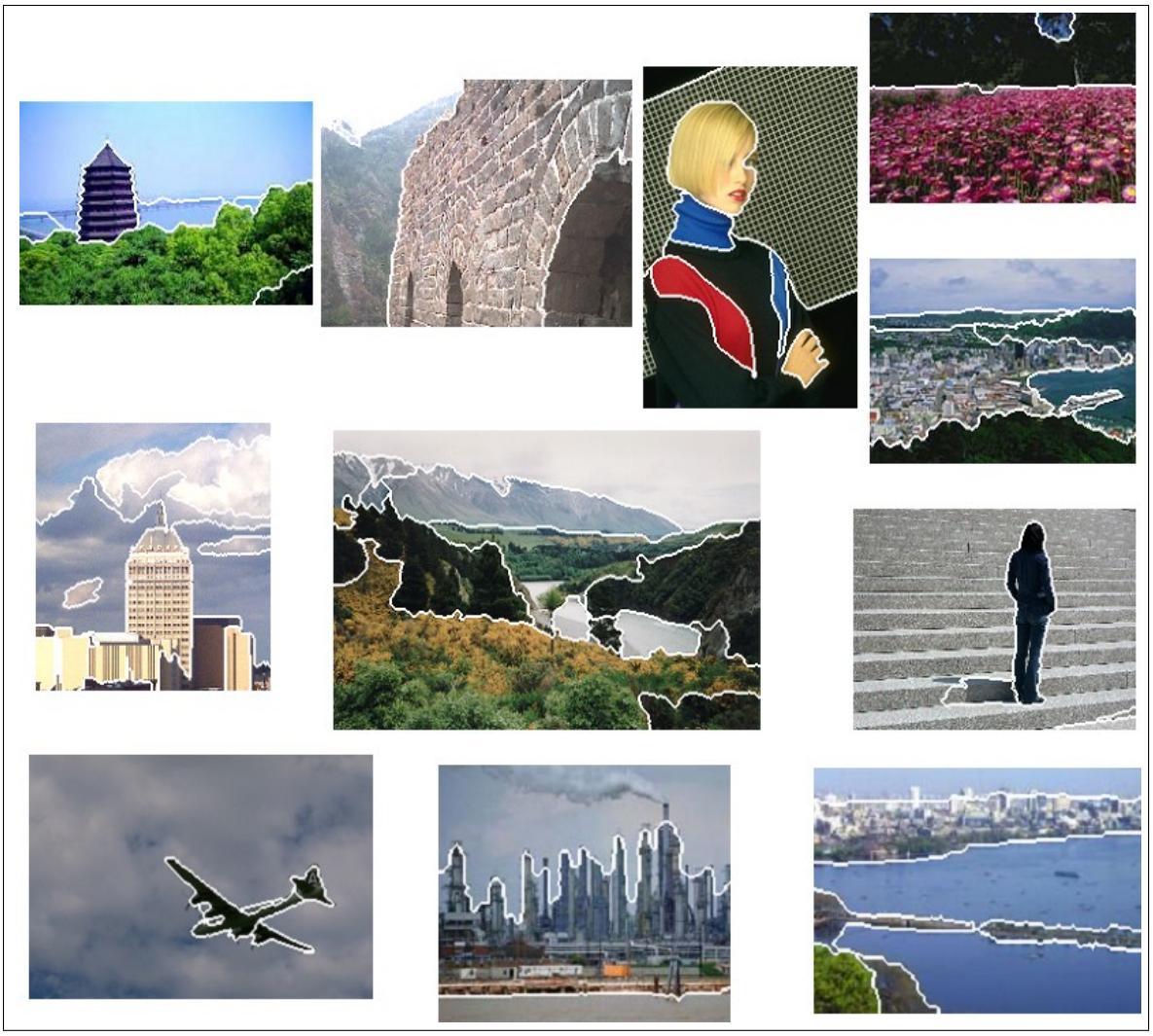


Figure 4.3: Segmentation Results

Chapter 5

Features for Classification

5.1 Color Features

5.1.1 What is Color

Color is a perceptual phenomenon, it is the human response to the particular wavelengths of light. Color is created by the interaction of three elements: an illuminant, an object, and an observer.

When an object appears to be of a given color, it is because the object transmits or reflects certain wavelengths of light. The human visual system associates particular combinations of visible wavelengths with different colors. The eyes contain two types of sensor cells, that are sensitive to light. The rods are essentially monochromatic, with a peak sensitivity at around the 510nm wavelength: They contribute to peripheral vision and allow us to see in relatively dark conditions, but typically they don't contribute to color vision. The sensation of color comes from the cones. Our eyes contain three different types of cones, each type sensitive to light of long, medium, and short wavelength.

5.1.2 Color Spaces

CIE 1931 Standard Observer (XYZ Color Space)

In 1931 the Commission Internationale De l'clairage (CIE) created a mathematical model that uses synthetic, imaginary primaries that represent the individual cone responses. This model converts continuous spectra that our eyes receive, into varying amounts of three different primaries. These primaries were labeled X, Y, and Z. Today CIE XYZ lies at the heart of all current implementations of color management; CIE XYZ, and its derivation CIE $L^*a^*b^*$, define light-independent, device and color independent spaces that software, device profiles, and drivers use when interpreting or translating color information.

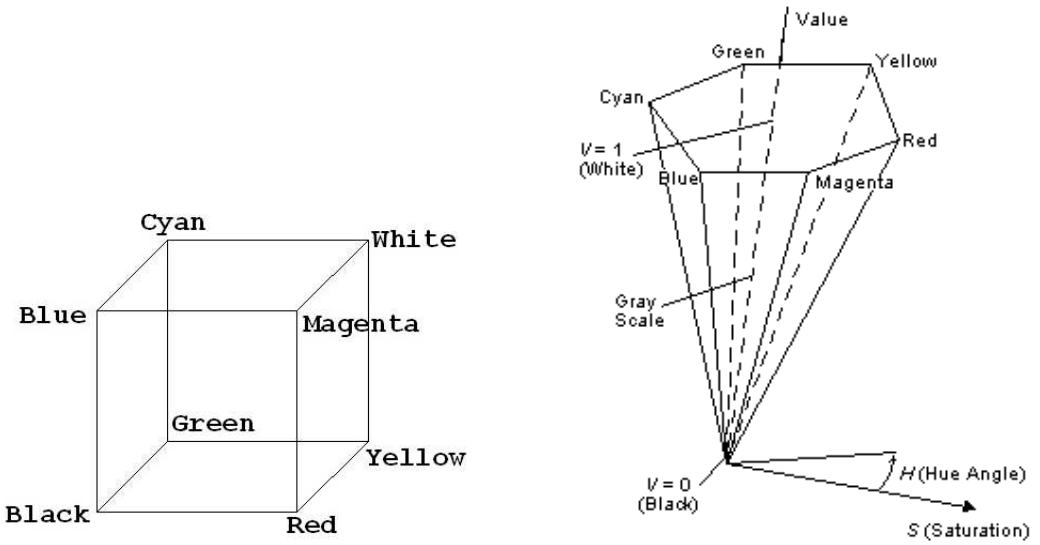
CIE $L^*a^*b^*$ Color Space

The CIE L*a*b* color space is derived from the CIE Standard Color Table by transforming the original X, Y and Z coordinates into the three new reference values of L^*, a^* and b^*. The objective of this transformation was to create a color space to aid the numerical classification of color differences. However, the CIE $L^*a^*b^*$ is only approximately perceptually uniform, and color differences are valid *only locally*. The L^* represents lightness, and its values run from 0 (black) to 100 (white). On each color axis, the values run from -128 to +128. On the a^* axis, positive values indicate amounts of red while negative values indicate amounts of green, and on the b^* axis, yellow is positive and blue is negative.

RGB Color Space

In 1931, the *Commission International de l'Eclairage* (CIE) standardized the primary colors at wavelengths $\lambda_R = 700nm$, $\lambda_G = 346.1nm$, $\lambda_B = 435.8nm$, which form a basis for the color monitors. As a result, the RGB color representation has become the standard for image storage. Each colored pixel is represented by three values $(R,G,B) \in [0,1]^3$. Hence, the RGB color space takes

the form of a cube of unit length, and represents all the colors that can be displayed on the CRT computer monitor.



(a) RGB Cube (b) HSV color space

Figure 5.1:

One of the limitations of the RGB Color Space is that it can not represent all the colors human can perceive. Figure 3.3 shows that RGB gamut occupies only part of the $L^*a^*b^*$ color space.

HSV Color Space

The HSV (hue, saturation, value) color space is a nonlinear transformation of the RGB color space, which is commonly used in the computer graphic applications. In it, the hue is represented by a circular region; Artists sometimes prefer to use the HSV color model over alternative models such as the RGB or CMYK, because of its similarities to the way humans tend to perceive color. The RGB and CMYK are additive and subtractive models, respectively, defining the color in terms of the combination of primaries, whereas the HSV encapsulates information about a color in terms that are more familiar to humans. Hue, the color type, ranges from 0-360°(but it is normalized

to $[0, 1]$ in many applications). Hue refers to the graduation of color within the optical spectrum, or visible spectrum, of the light. A particular hue value refers to a particular color within this spectrum, as defined by its dominant wavelength. Saturation ranges from 0 to 1, and lower the saturation of a color, the more ”grayness” is present and the more faded the color will appear. The value (lightness) of the color also ranges from 0 to 1.

5.1.3 Color Features in Image Retrieval

Color has been used extensively as a low-level feature for image retrieval [105] [106] [11] [107]. Many of the existing techniques are based on the color image histogram. For example: ImageRover, QBIC, MARS, and many other use variations of color-based histogram matching. A common characteristic of these methods is the global approach, where attempt is made to match the whole images by appearance. Even though such techniques have been successful in specialized settings, they have several significant shortcomings:

- Even in a coarsely quantized color space, histogram matching during retrieval in large databases involves a significant amount of computation.
- Histogram does not take into account the color composition of the image, *e.g.,* whether one color is concentrated in one corner of the image, or distributed throughout the image. In addition, histograms cannot discriminate between rapidly varying color patterns (*e.g.,* checkerboards), which the human eye may not even be able to perceive as separate colors, and solid colors that appear as small or large blobs around the image.
- Finally, histogram based methods do not take into consideration the fact that the human visual system can only perceive a few colors at a time.

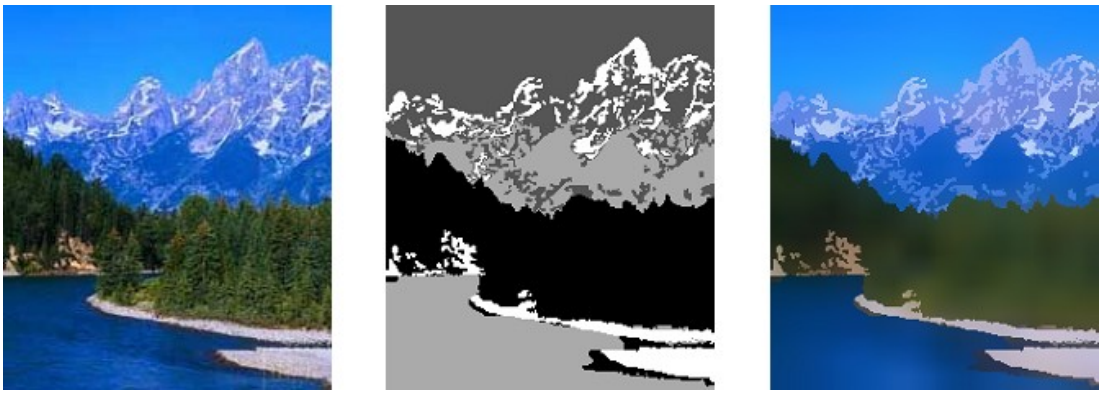
5.1.4 Spatially Adaptive Dominant Colors as Proposed Color Composition Features

Considering the shortcomings of the histogram based methods mentioned in the previous section, we are led to the conclusion that for an effective CBR system we need to obtain compact, spatially adaptive, segment level color features that incorporate knowledge of human perception. In this section we propose spatially adaptive dominant colors as features that satisfy such criterion.

An important characteristic of human color perception is that the human eye cannot simultaneously perceive a large number of colors [97], and that the number of colors that can be internally represented and identified in cognitive space is about thirty [108]. Using a small set of color categories (dominant colors) provides a compact and efficient representation, and more importantly, makes it easier to capture invariant properties in object appearance [109].

Existing approaches for extracting the dominant colors [97] [110] [111] [112] [113] [114], rely on the assumption that the characteristic colors of an image are relatively constant, *i.e.,* they do not change due to variations in illumination, perspective, etc. Consequently, the resulting color classification could be quite inadequate due to the lack of spatial adaptation and spatial constraints [101].

To handle a wide range of consumer produced images we are interested in (indoor, outdoor, landscapes, cityscapes, plants, animals, people, and man-made objects), we have to account for color and lighting variations in the scene. In addition, we have to take into consideration the adaptive nature of the human visual system [115], *i.e.,* humans perceive regions with spatially varying color as a single color. Furthermore, color perception could also depend on the surrounding colors [115], *i.e.,* an observer's notion of a blue, brown, or green color is highly dependent on the surrounding colors [116]. Conversely, the photometric description of a color that is perceived as blue, brown, or green could vary substantially with changing lighting conditions within the same image or across images or display devices.



(a) (b) (c)

Figure 5.2: Color image segmentation. (a) Original color image. (b) ACA color classes. (c) Locally averaged image.

In order to account for the spatially varying image characteristics and the adaptive nature of the HVS, we utilize the idea of ***spatially adaptive dominant colors.*** The proposed segment color composition feature representation consists of a limited number of locally adapted dominant colors and the corresponding percentage of occurrence of each color within a segment:

$$f_c(S) = \{(c_i, p_i), i = 1, \ldots, M, p_i \in [0,1]\} \tag{5.1}$$

where each of the dominant colors, c_i, is a three dimensional vector in *Lab* space, and p_i are the corresponding percentages. S stands for a particular segment, and M is the total number of dominant colors in the segment.

A particularly useful property of this approach is its robustness to the number of classes. This is because the gradual color adaptation makes it possible to use one color class to represent a wide range of similar colors, provided that they vary gradually over the image. That means that the same ACA color class might represent different colors in different segments. Thus, one of the

advantages of using the ACA to obtain spatially adaptive dominant colors is that we only need to specify the parameter K, which then determines the maximum number of dominant colors ($M \leq K$) in any given region of the image. Usually, a small number (*e.g.,* $K = 4$) is quite adequate.

The ACA segments the image into color classes, as shown in Fig. 5.1.4 (b). In the Fig. 5.1.4(c), each class is represented by the characteristic function $\mu^k(x,y)$, *i.e.,* a color that is equal to the average color of the pixels in its neighborhood that belong to that class [101].

5.2 Texture Features

5.2.1 What is Texture

The term "texture", although widely used, does not have a commonly accepted definition. Haindl [117] states that: "Texture is generally a visual property of a surface, representing the spatial information contained in object surfaces." A plethora of texture analysis methods proposed in literature can be classified into four major types [118]:

- Statistical (Co-occurrence Matrices [119], Autocorrelation Features),
- Geometrical (Voronoi tessellation Features [120], Structural [121] [122]),
- Model Based Methods (Random Field Models [123] [124] [125], Fractals [126] [127]),
- Signal Processing Methods (Spatial Domain Filters [128] [129], Fourier domain filtering [130], Gabor and Wavelet models [131] [132] [133])

5.2.2 Spatial Texture Features

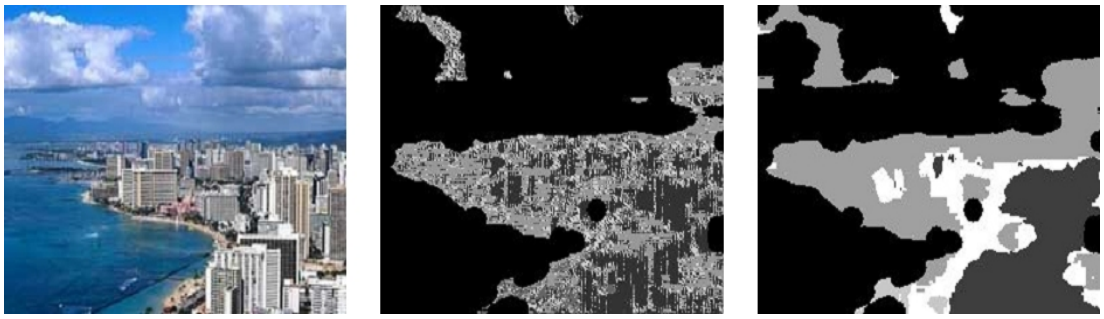
The spatial texture features we propose to use for the classification are based on the available subband coefficients already precomputed in the segmentation step. As mentioned in the Chapter

4, the spatial texture feature extraction relies on the steerable pyramid decomposition, which is a good approximation of the visual cortex. Such a decomposition can be designed to produce any number of the orientation bands. Furthermore, this approach utilizes the *local median energy* of the subband coefficients, where the energy is defined as the square of the coefficients. The advantage of the median filter is that it suppresses textures associated with transitions between regions, while it responds to texture within the uniform regions. The use of median local energy as a nonlinear operation also agrees with Graham [134] and Graham and Sutter [135] [136] who concluded that a nonlinear operator in texture segregation must have accelerating/expansive nature.

We use a steerable filter decomposition with four orientation subbands (horizontal, vertical, +45 -45. Most researchers have used four to six orientation bands to approximate the orientation selectivity of the HVS (*e.g.,* [137] [138]). However, since the images are fairly small, a one-level decomposition (lowpass band, four orientation bands, and highpass residue) is adequate.

A pixel is classified as smooth if there is no substantial energy in any of the four orientation bands. The next step is to classify the pixels in the non-smooth regions. As we mentioned above, it is the maximum of the four subband coefficients, $s_i(x,y)$, that determines the orientation of the texture at each image point. The texture classification is based on the local histogram of these indices. A median operation is used to increase the response to the texture within uniform regions and suppress the response due to textures associated with transitions between regions. If there is no dominant orientation, the pixel is classified as complex.

An example is presented in Figure. 5.2.2. The original color image is shown in (a), maxima of the subband coefficients are shown in (b), (the smooth regions are shown in black, and the nonsmooth regions are shown in different shades of gray representing the indices s_i of the subband coefficients with maximum energy). Part (c) shows the resulting texture classes, after the median operation (black denotes smooth, white denotes complex, and light gray denotes directional textures). The window for the median operation was 23×23. Two types of features were tested: maxima of subband coefficients and median of maxima [18]. The results indicate that a median of



(a) (b) (c)

Figure 5.3: (a) Original Image. (b) Maxima of Subband Coefficients. (c) Medians of Maxima.

maxima provides more successful classification of the segments.

5.3 Other Features

Although the main emphasis of this project is semantic classification of images based on color and spatial texture, other features such as position and shape may also be used to aid semantic segment classification.

Position of the segment within the image can distinguish among semantic categories that have similar color and texture characteristics. For example, sky and water both have blue dominant color and smooth spatial texture descriptors. However, sky is almost certain to be at the top of the image, while water is usually at the lower section.

Shape recognition is a problem that has received a considerable attention in recent years, and a vast number of techniques have been proposed. However, at this point in our CBIR system implementation, shape features might be of limited use since most of the semantic categories of our interest, such as sky, vegetation, trees/bushes, etc., lack a clearly defined shape.

Chapter 6

Segment Wide Feature Extraction

6.1 Color Texture Feature Selection

We now review the color-texture features that were developed for the adaptive perceptual segmentation algorithm proposed in [18]. These features can also be used for segment classification.

The segmentation approach [18] incorporates models of human perception and signal characteristics. It is based on two types of spatially adaptive features. The first provides a localized description of the color composition of the texture and the second models the spatial characteristics of its grayscale component.

The color composition feature exploits the fact that the HVS cannot simultaneously perceive a large number of colors. In addition, it accounts for the spatially varying image characteristics and the adaptive nature of the HVS. It thus consists of a small number of spatially adaptive dominant colors and the corresponding percent occurrence of each color in the vicinity of a pixel:

$$f_c(x,y,N_{x,y}) = \{(c_i,p_i), i = 1,\ldots,M, p_i \in [0,1]\} \tag{6.1}$$

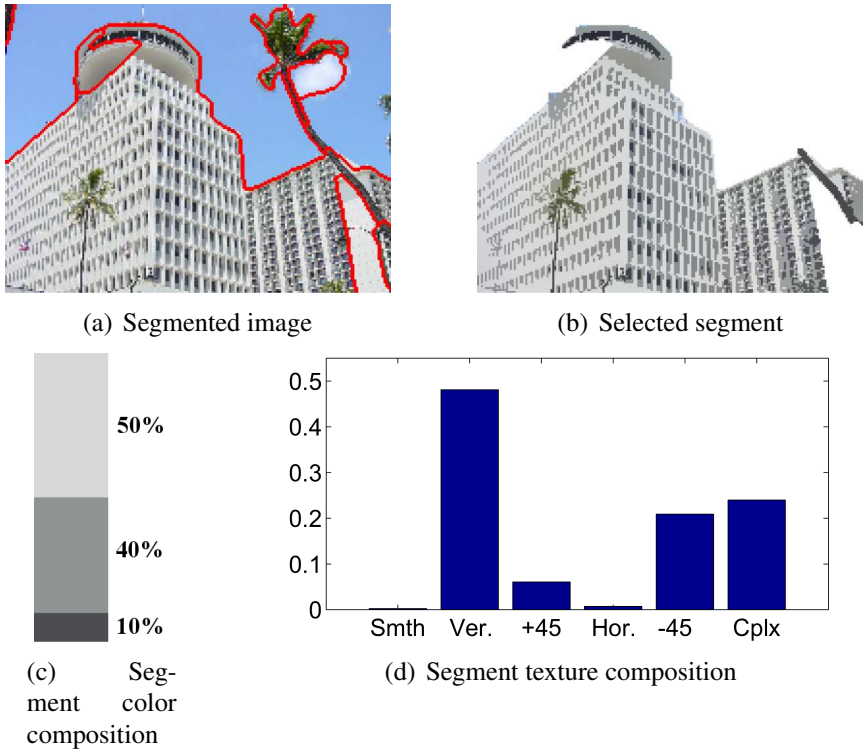
where c_i is a 3-D color vector and p_i is the corresponding percentage. $N_{x,y}$ denotes the neigh-

borhood of the pixel at (x,y) and M is the number of dominant colors in $N_{x,y}$; a typical value is $M = 4$. The spatially adaptive dominant colors are obtained using the adaptive clustering algorithm (ACA) [139]. The perceptual similarity between two color composition feature vectors is based on the "Optimal Color Composition Distance (OCCD)," which finds the optimal mapping between the color composition features of two segments and computes the average distance between them in the *CIE L*a*b** color space.

The spatial texture feature extraction is based on a multiscale frequency decomposition with four orientation subbands (horizontal, vertical, +45 -45. Here, we use a one-level steerable filter decomposition with four orientation subbands. The local energy of the subband coefficients is used as a simple but effective characterization of spatial texture. At each pixel location, the maximum of the four subband coefficients determines the texture orientation. A median filtering operation boosts the response to texture within uniform regions and suppresses the response resulting from to transitions between regions. Pixels are then classified into smooth and non-smooth classes, and non-smooth pixels are further classified on the basis of dominant orientation, as horizontal, vertical, +45 -45 and complex (i.e., no dominant orientation).

6.2 Segment Wide Color Texture Feature Extraction

Once the image has been segmented into regions, the goal is to relate their features to semantic concepts. This is done in two stages. First, we derive semantic labels at the segment level, and then we classify the entire image into categories. The key to bridging the gap between low-level image primitives and high-level semantics is the extraction of medium-level segment descriptors. These include region-wide color and texture features, as well as the segment location, size, and boundary shape. The semantic label for each segment can be extracted from a combination of such features, as well as the properties of the neighboring segments. The success of this approach, however, depends on having segments that are semantically meaningful. The methodology for obtaining



(a) Segmented image (b) Selected segment

(c) Segment color composition (d) Segment texture composition

Figure 6.1: Segment feature extraction

such segmentations was described in the previous sections and incorporates knowledge of human perception and image characteristics to segment scenes into perceptually uniform regions. In this section, we discuss the development of segment-wide color and spatial texture features. The segment features that we used to obtain the image segmentation are not necessarily the same as those that are most suitable for assigning a semantic label to a segment. Image segmentation requires a combination of local and global features, while region interpretation requires region-wide features.

Thus, for each segment, we recalculate the color composition and spatial texture features using only information from within the segment, that is, the local averages and medians are computed across and strictly within the segment. An example is shown in Fig. 6.1, where Fig. 6.1(a) shows a segmented image, Fig. 6.1(b) shows a selected segment, and Fig. 6.1(c) shows the color composition of the segment (dominant colors and percentages). The texture features of the seg-

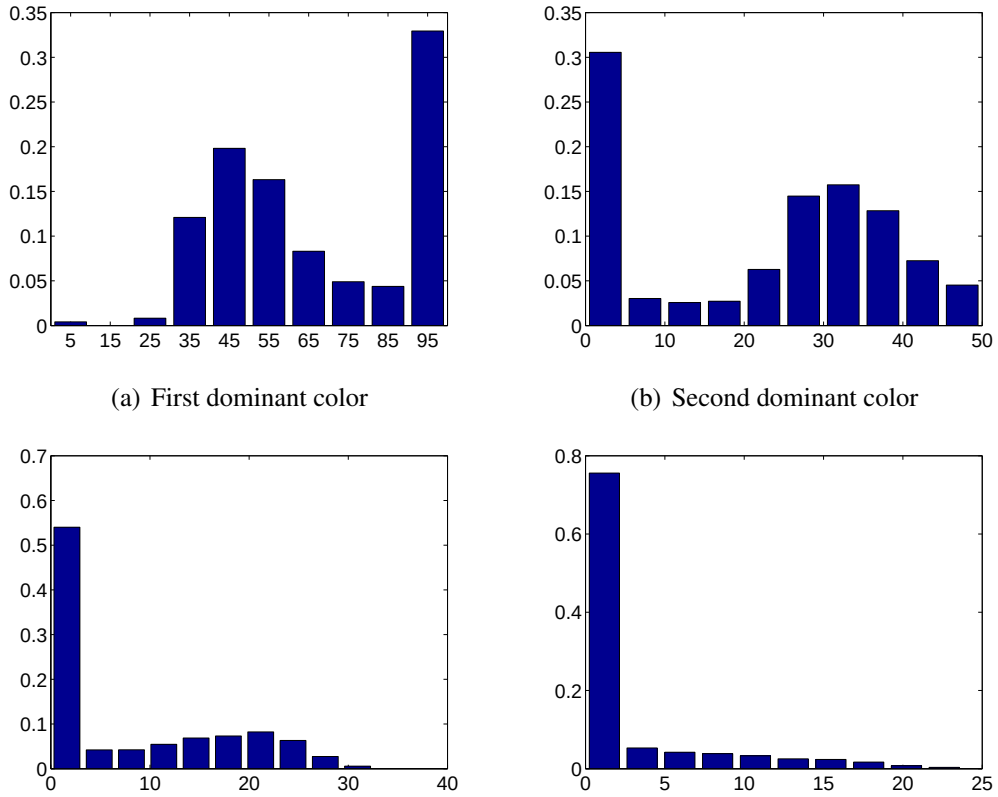
ment can be similarly described by the percentage of smooth, horizontal, vertical, +45 -45 and complex pixels as shown in Fig. 6.1(d).

6.2.1 Dominant Colors as Color Features

For this thesis, we considered two types of dominant color feature representations. The first type of color feature considered can be described by the L*a*b* coordinate of the dominant color with highest percentage (referred from now on as the first dominant color) and the L*a*b* color space coordinates difference between the first and second dominant color. This type of feature was motivated by our statistical analysis of dominant colors associated with segments in our database, which revealed that the great majority of segments could be described by first two dominant colors. The statistics of dominant colors is presented in Fig. 6.2(a)-(d) where each figure presents respectively, the histogram for each dominant color occupying certain percentage of the segment area. The horizontal axis represents the area percentage, while the vertical axis has been normalized and it could be said that it represents the probability of occurrence for a particular bin. By analyzing the data we can conclude that color content of a particular segment can be described by the two most dominant colors without any significant loss of information. Furthermore, by analyzing the L*a*b* distance among the dominant colors that in majority of cases the second dominant color is less than twenty units away from the first. This means that for a great majority of segments the second dominant color is similar to the first. The histogram of distances between dominant colors is presented in Fig. 6.3 where Fig. 6.3(a) presents distances between the first and second dominant colors and Fig. 6.3(b) between the first and third.

6.2.2 Perceptually Quantized Colors as Color Features

Another way to obtain a color feature representation is to use a perceptually quantized color space, whereby the color space is reduced to a set of several perceptually distinct categories. The moti-



(a) First dominant color

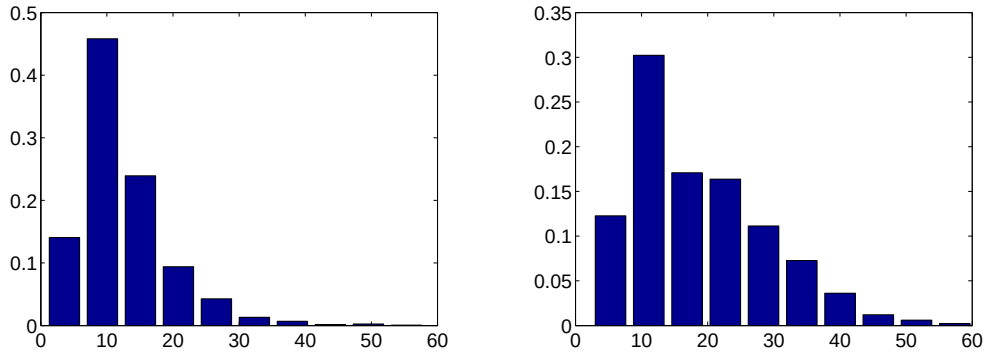
(b) Second dominant color

(c) Third dominant color

(d) Fourth dominant color

Figure 6.2: Statistics of dominant colors. The horizontal axis represents the percentage of the area that the dominant color occupies in a segment and the vertical axis represents the probability of occurrence for each bin.

vation for this type of color feature came as a natural extension our perceptual approach, and also as a attempt to handle an apparent asymmetry between the two types of features. As explained in Sec.6.1, the spatial texture feature consists of six labels and the corresponding percentages, while the color composition feature consists of up to four dominant colors (which take essentially a continuum of values) and the associated percentages. In order to reduce the dimensionality of the color composition features, we assign color names to the dominant colors of each region. The procedure for assigning color names can be found in [97]. The selected color names (labels) are consistent with a National Bureau of Standards recommendation for color names. The syntax con-



(a) Histogram of distances between first and second dominant color

(b) Histogram of distances between first and third dominant color

Figure 6.3: Distances between dominant colors in L*a*b* color space

tains color names for 267 regions in color space, and employs English terms to describe colors along the three dimensions of the color space: hue, lightness and saturation. There are seven discrete values for lightness, five discrete values for saturation, and a basic set of eleven prototypical hues, as shown in Table 3.2.1. Thus, if we assign labels based on hue only, we end up with 14 labels (and corresponding percentages) instead of a continuum of color values, which establishes a symmetry with the spatial texture features. The use of a limited number of colors is consistent with Boyntons study, which found that when people are asked to categorize colors, the number of perceptually distinguishable color categories is small. (See his 1989 paper Eleven colors which are almost never confused [140].) Finally, we should note that, in addition to facilitating semantic labeling, the color labels (either at the segment or at the image level) can also be used to allow queries on the basis of color composition.

6.3 Semantic Labeling

In [1, 10, 141], subjective experiments were conducted in order to identify important semantic categories that humans use for image organization and retrieval. For example, they have discovered

Natural				Man-made	Human
Vegetation	Sky	Landform	Water	Building/House	Face
Grass	Day-sky	Snow		Bridge	Person
Trees/bushes	Night-sky	Ground		Car	People
Forest	Sun	Mountain/Hill		Boat	
Flowers	Clouds			Airplane	
	Sunrise/set			Pavement	
				Other Manmade	

Table 6.1: Segment Labels

two important dimensions in human similarity perception: "natural" vs. "man-made," and "humans" vs. "non-human." In addition, certain cues, such as "sky," "water," "mountains," etc., were found to have an important influence in human image perception [142]. Thus, rather than trying to obtain a complete and detailed description of every object in the scene, this suggests that it may be sufficient to isolate segments of such perceptual significance, which in turn can be used to correctly classify an image into a given category (e.g., "natural," "man-made," "outdoor," etc.). Our first goal will be to assign labels to image segments. For this, we need to relate the segment features to semantic labels, but first, we must decide what the labels will be. To this end, we have assembled a vocabulary of labels consistent with the above findings, as well as those used in annotation of the NIST TRECVID 2003 development set [143]. The set of labels we selected is a subset of NIST lexicon. To describe the content of an image we use two types of labels, segment and scene labels. The segment labels describe the semantics of a particular segment (e.g., building, sky), while the scene labels describe the (higher-level) semantic content of the image (e.g., beach scene). The latter cannot be inferred from a particular image segment alone. The segment labels we chose are shown in Table 6.1, and are arranged in a hierarchical manner. Note that only leaf nodes are used in the annotation.

Chapter 7

Learning and Classification

7.1 Classification Setup

We performed several sets of experiments using approximately 3300 photographs. The majority of the images were obtained from the Corel Stock Photo Library. Additional images were obtained from a Key Photos Library and the investigators personal repository. The images in the database cover a variety of outdoor scenes, with a wide range of themes. The images were segmented using the adaptive perceptual color-texture image segmentation algorithm [18] described above, and the resulting segments were manually labeled to be used as the ground truth in supervised learning. Each segment was assigned exactly one label. Segments whose area was less than three percent of total image area were not considered. This resulted in approximately 13000 labeled segments, 80% of which were used for training and the rest for testing.

7.2 Supervised vs. Unsupervised Learning Techniques

For the training and classification we considered both unsupervised (clustering) and supervised learning techniques. Among unsupervised techniques we experimented with K-means, K-nearest

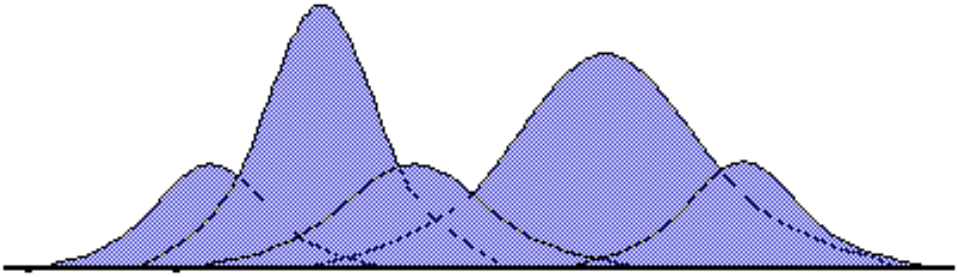


Figure 7.1: Arbitrary density approximated as a mixture of Gaussians

neighbors, agglomerative, and conglomerative clustering methods, while in the supervised learning experiments we used Gaussian Mixture Models (GMM) [144], Support Vector Machines (SVM) [145], and Linear Discriminant Analysis (LDA) [146]. It quickly became clear that supervised techniques are best suited for the problem at hand, primarily because of the complexity of the cluster configurations. Supervised techniques require the existence of ground truth for a large database of segments.

7.3 Gaussian Mixture Models

In the Gaussian Mixture Model approach, each label from the training set is represented as a probability density, and is parametrized as a mixture of Gaussian components. An individual multivariate Gaussian is represented as

$$f_k(\mathbf{x}) = \mathbf{N}(\mu, \sum) = \frac{1}{\sqrt{(2\pi)^d |\sum|}} exp(-\frac{1}{2}(\mathbf{x}-\mu)^T \sum^{-\mathbf{1}} (\mathbf{x}-\mu)) \tag{7.1}$$

where μ is the mean, $\sum$ is the covariance matrix, and $|\cdot|$ denotes the determinant. The mixture density is then represented as

$$f(\mathbf{x}) = \sum_{\mathbf{k=1}}^{\mathbf{K}} a_k f_k(\mathbf{x}) \tag{7.2}$$

where a_k represents the weight of each Gaussian component.

The mixture model is fitted to the training data using the Expectation-Maximization (EM) [144] algorithm. The EM algorithm is an iterative maximum likelihood (ML) technique that computes the most likely estimate of a distribution $p(\mathbf{X}|\theta)$ parametrized by θ.

$$\theta^* = arg\,max_\theta\, p(\mathbf{X}|\theta) = arg\,max_\theta \prod_{p=1}^{n} p(\mathbf{x_p}|\theta) \tag{7.3}$$

EM introduces a hidden variable the estimation of which simplifies the maximization of $p(\mathbf{X}|\theta)$. After initialization, the EM technique iterates between expectation and maximization steps to improve the parameter estimates. The expectation step computes the expected value of the hidden variable given the data and the current value of the parameters. The maximization step modifies the parameters in order to maximize the joint distribution of the data and the hidden variable. After each iteration, the likelihood function is guaranteed to increase, and thus, model convergence to local optimum is assured. Once the model has been determined, the membership of the test feature is determined using Bayes rule.

7.4 Support Vector Machines

SVM belongs to a class of supervised machine learning algorithms that are used in classification and pattern recognition. SVM maximizes the margin around hyperplanes separating the different classes. The SVM decision function is specified in terms of a subset of training samples, the support vectors. The support vectors are the elements of the training set that if removed would change the position of the dividing hyperplane. (They are thus critical elements of the training set). A boundary hyperplane is expressed as

$$\mathbf{w^T x} + \mathbf{b} = \mathbf{0} \tag{7.4}$$

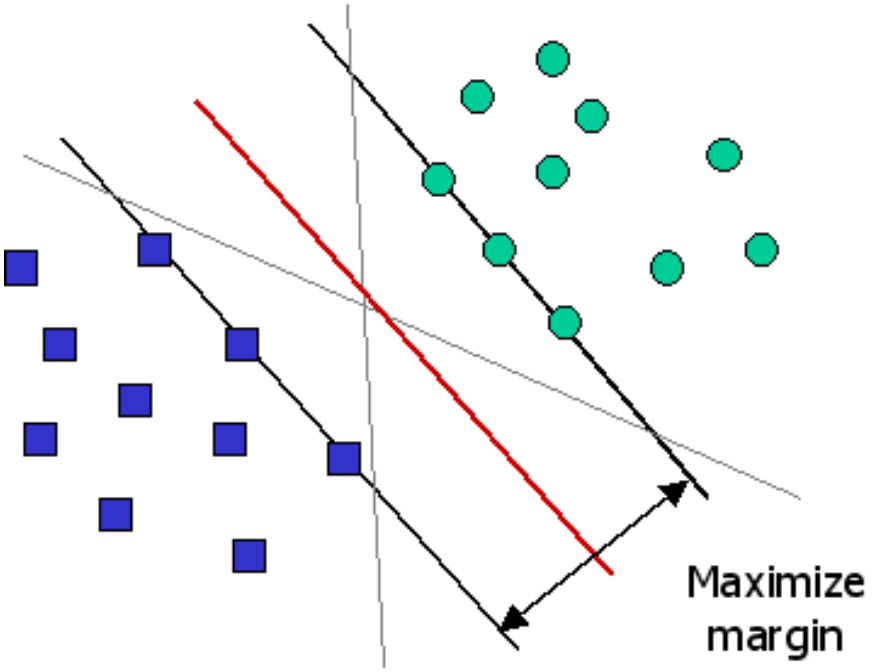


Figure 7.2: SVM as a maximum margin classifier

where $\mathbf{x}$ is a vector in a vector space, $\mathbf{w}$ is a weight coefficient vector and b is a bias term. The distance between a training vector $\mathbf{x_i}$ and the boundary, called margin, is expressed as follows:

$$\frac{|\mathbf{w^T x_i} + \mathbf{b}|}{\|\mathbf{w}\|} \tag{7.5}$$

The optimal boundary maximizes the minimum of 7.5, and the optimization function can be expressed as:

$$minimize : \mathbf{w^T w} \tag{7.6}$$

$$subject\ to : \mathbf{y_i}(\mathbf{w^T x_i} + \mathbf{b}) \geq \mathbf{1} - \xi_\mathbf{i} \tag{7.7}$$

where $\mathbf{y_i}$ is 1 if $\mathbf{x_i}$ belongs to one set and -1 if $\mathbf{x_i}$ belongs to the other set, and the ξ_i's are the slack variables that indicate tolerances of misclassification. This means that a training vector is allowed to exist in a limited region in the wrong side along the boundary.

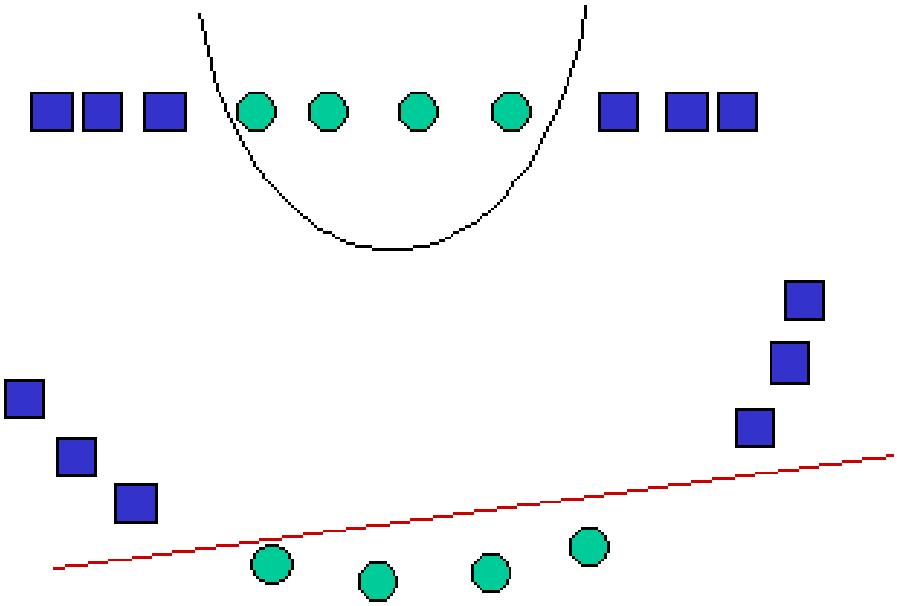


Figure 7.3: Kernel transformation

In the non-linear case, SVM can be extended by transforming the original data through a kernel function to a new space, usually of higher dimension, in which the data becomes linearly separable. The kernel function satisfies

$$\mathbf{K}(\mathbf{x},\mathbf{x}') = \mathbf{R}(\mathbf{x})^{\mathbf{T}}\mathbf{R}(\mathbf{x}') \tag{7.8}$$

where $\mathbf{R}$ is transformation to a higher dimensional space. The above equation indicates that the kernel function is equivalent to the distance between $\mathbf{x}$ and $\mathbf{x}'$ measured in the higher dimensional space transformed by $\mathbf{R}$. Note that the boundary in the transformed space is obtained as

$$\mathbf{w}^{\mathbf{T}}\mathbf{R}(\mathbf{x}) + \mathbf{b} = \mathbf{0} \tag{7.9}$$

The optimization function in the transformed space is also obtained by substituting $\mathbf{x_i}^{\mathbf{T}}\mathbf{x_j}$

with $\mathbf{K}(\mathbf{x_i}, \mathbf{x_j})$. This means that all of the calculation can be achieved by using $\mathbf{K}(\mathbf{x_i}, \mathbf{x_j})$ only, and we do not need to know what $\mathbf{R}$ or the transformed space actually is.

7.5 Linear Discriminant Analysis

LDA is a method that belongs to the class linear classifiers. It tries to find a subspace of projections such that samples from the different classes are well separated, or in other words to find directions that are useful for the data classification. The main idea of LDA is to find a directions which maximize the variance between the class means and at the same time minimize the variance within each class.

The measure of separation between the class means is defined as:

$$S_B = \sum_c N_c(\mu_c - \bar{\mathbf{x}})(\mu_c - \bar{\mathbf{x}})^T \tag{7.10}$$

where,

$$\mu_c = \frac{1}{N_c}\sum_{i \in c} x_i \tag{7.11}$$

$$\bar{\mathbf{x}} = \frac{1}{N}\sum_c N_c \mu_c \tag{7.12}$$

The measure of within class variance is defined as:

$$S_W = \sum_c \sum_{i \in c} (\mathbf{x}_i - \mu_c)(\mathbf{x}_i - \mu_c)^T \tag{7.13}$$

S_B is usually referred to as the "between the classes scatter matrix", while S_W is known as the "within the classes scatter matrix".

Using these measures we can obtain the objective function as:

$$J(\mathbf{w}) = \frac{\mathbf{w}^T S_B \mathbf{w}}{\mathbf{w}^T S_W \mathbf{w}} \tag{7.14}$$

where $\mathbf{w}$ is a transformation matrix. The optimization function can be maximized by taking the derivative of $J(\mathbf{w})$ with respect to $\mathbf{w}$ and setting it to zero.

$$\frac{d}{d\mathbf{w}} J(\mathbf{w}) = \frac{(2S_B\mathbf{w})\mathbf{w}^t S_w \mathbf{w} - (2S_w\mathbf{w})\mathbf{w}^t S_b \mathbf{w}}{(\mathbf{w}^t S_w \mathbf{w})^2} = 0 \tag{7.15}$$

which can be reduced to

$$\frac{\mathbf{w}^t S_w \mathbf{w}(S_b\mathbf{w})}{\mathbf{w}^t S_w \mathbf{w}} - \frac{\mathbf{w}^t S_b \mathbf{w}(S_w\mathbf{w})}{\mathbf{w}^t S_w \mathbf{w}} = 0 \tag{7.16}$$

$$S_b w - \frac{\mathbf{w}^t S_b \mathbf{w}(S_w\mathbf{w})}{\mathbf{w}^t S_w \mathbf{w})} = 0 \tag{7.17}$$

Note that

$$\frac{\mathbf{w}^t S_b \mathbf{w}(S_w\mathbf{w})}{\mathbf{w}^t S_w \mathbf{w}} = \lambda \tag{7.18}$$

and it can be concluded that the objective function $J(\mathbf{w})$ is maximized by solving the generalized eigenvalue problem

$$S_B\mathbf{w} = \lambda S_W \mathbf{w} \tag{7.19}$$

so that columns of an optimal $\mathbf{w}$ are the eigenvectors associated with largest eigenvalues.

If S_w is nonsingular, the nontrivial solution can be obtained from the standard eigenvalue problem

$$S_w{}^{-1} S_b \mathbf{w} = \lambda \mathbf{w} \tag{7.20}$$

However, S_w is often singular, or at the very least, ill-conditioned. For the singular and ill-conditioned cases, the generalized eigenvalue problem can be solved by applying the QZ fac-

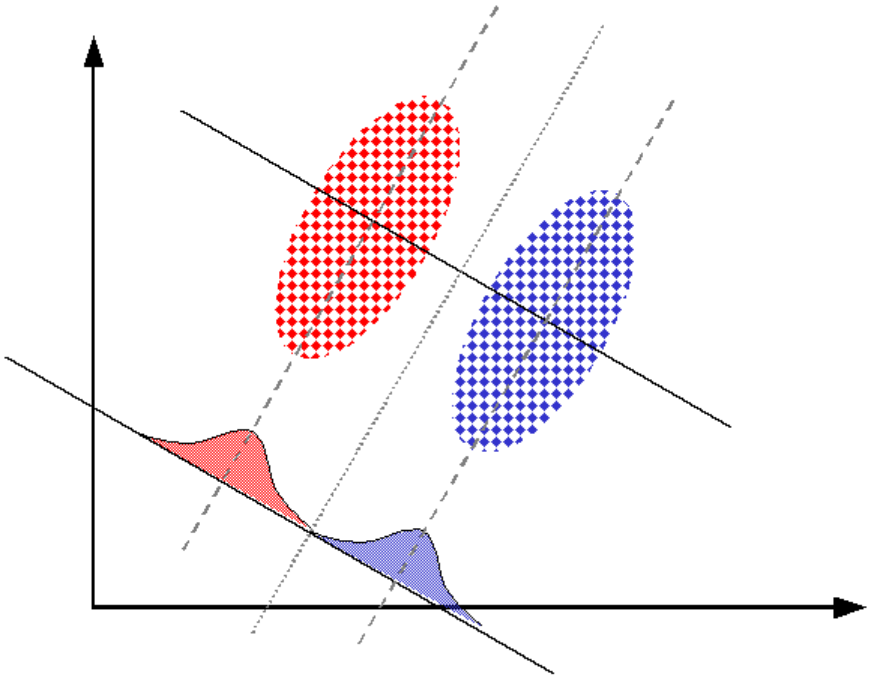


Figure 7.4: LDA example in two dimensions

torization [147] algorithm without any inversion of S_w or its submatrix. The QZ factorization is the generalization of QR (orthogonal-triangular) algorithm. Here Q and Z are both orthogonal matrices.

The QZ algorithm or *Generalized Schur Decomposition* attempts to find an alternative approach to the $\mathbf{A} - \lambda\mathbf{B}$ problem. The key step is to compute nonsingular matrices $\mathbf{Q}$ and $\mathbf{Z}$ such that

$$\mathbf{A_1} = \mathbf{Q^{-1}AZ} \tag{7.21}$$

and

$$\mathbf{B_1} = \mathbf{Q^{-1}BZ} \tag{7.22}$$

are each in the canonical form. It can easily be proved that the pencils $\mathbf{A} - \lambda\mathbf{B}$ and $\mathbf{A_1} - \lambda\mathbf{B_1}$ are equivalent and it follows that

$$\lambda(\mathbf{A},\mathbf{B}) = \lambda(\mathbf{A_1},\mathbf{B_1}) \tag{7.23}$$

since

$$\mathbf{Ax} = \lambda\mathbf{Bx} \Longleftrightarrow \mathbf{A_1y} = \lambda\mathbf{B_1y}, \qquad \mathbf{x} = \mathbf{Zy}. \tag{7.24}$$

The generalized eigenvalues of original problem are extracted from diagonal elements of matrices $\mathbf{A_1}$ and $\mathbf{B_1}$

$$\lambda_i = a_{ii}/b_{ii} \quad for \quad b_{ii} \neq 0 \tag{7.25}$$

and the corresponding eigenvectors can be extracted from the columns of $\mathbf{Z}$.

In the case of Linear Discriminant Analysis, the matrices $\mathbf{S_w}$ and $\mathbf{S_b}$ are positive semidefinite covariance matrices. Although the QZ algorithm can be used to solve symmetric semi-definite problem, it has a flaw of destroying both symmetry and definiteness. In this special case QZ algorithm can be improved by exploiting the special properties of covariance matrices. More stable and efficient algorithms can be devised by computing a matrix $\mathbf{X}$ [148] such that

$$\mathbf{A_1} = \mathbf{X^T AX} = diag(a_1, ..., a_n) \tag{7.26}$$

and

$$\mathbf{B_1} = \mathbf{X^T BX} = diag(b_1, ..., b_n) \tag{7.27}$$

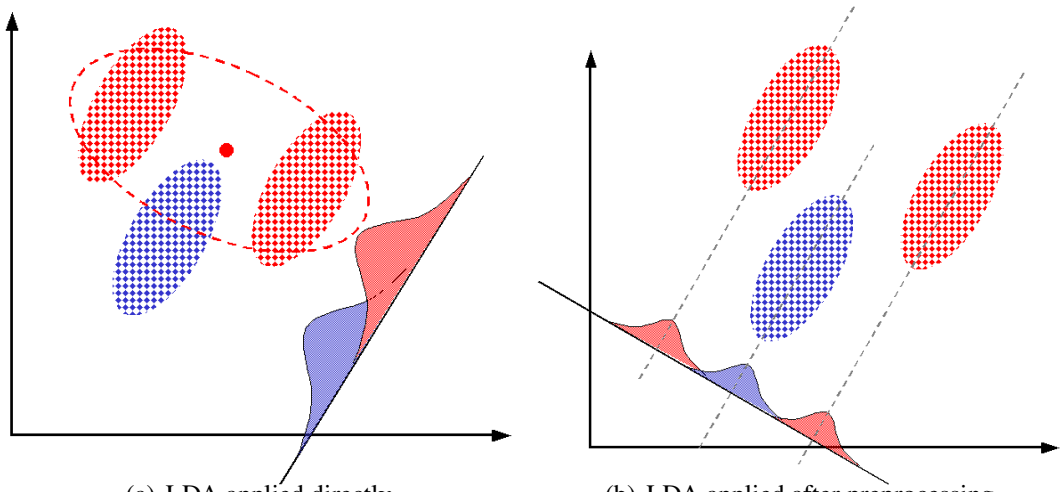
are both in canonical form. The generalized eigenvalues of the original problem are extracted as

$$\lambda_i = a_i/b_i \tag{7.28}$$

and the corresponding eigenvectors can be extracted from the columns of $\mathbf{X}$.

We should also note, that for LDA to work, the data for each class has to form a single cluster. Furthermore, although not a requirement, LDA assumes that the underlying class distribution can be approximated with a Gaussian.

It is reasonable to expect that a particular label may consist of more than one cluster. For



(a) LDA applied directly (b) LDA applied after preprocessing

Figure 7.5: LDA applied to the class consisting of multiple clusters

example, the label "water" may be represented by blue dominant color and smooth or horizontal texture, thus resulting in two clusters. In such a case, the mean and the covariance of the data set belonging to the label consisting of several disjoint clusters would be miscalculated resulting in suboptimal classification when LDA is applied. This case is represented in the Figure 7.5(a).

To deal with such cases, we experimented with applying the clustering algorithm to each category in the training set to create additional clusters before applying LDA. Note that misclassification among clusters that belong to the same label is not recorded as a classification error. The effect of applying the clustering algorithm prior the LDA is depicted in Figure 7.5(b).

The **K-means** is an algorithm for clustering N data points into K disjoint subsets. The **K-means** objective function, to be minimized, is represented as

$$J = \sum_{j=1}^{k} \sum_{i=1}^{n} \|x_i^{(j)} - \mu_j\|^2 \tag{7.29}$$

where $\|x_i^{(j)} - \mu_j\|^2$ is a chosen distance measure between a data point $x_i^{(j)}$ and the cluster center μ_j.

The **K-means** algorithm consists of a simple iterative procedure, where initially, the K

cluster centers are assigned at random. Data points are then assigned to the cluster whose centroid is closest to that point, and new cluster centers are calculated. These two steps are then alternated until a stopping criterion is met, usually when there is no further change in the assignment of the data points.

Although it converges only to a local minimum and the number of clusters must be determined before hand, we found that for our purposes **K-means** outperformed other unsupervised clustering techniques such as k-nearest neighbors and other types of agglomerative clustering.

We found that LDA is best suited for the problem at hand because it works better with noisy data and is less sensitive to data redundancies than the other two approaches. Furthermore, LDA is not critically dependent on the correct choice of a Kernel function like in the SVM approach. The GMM approach achieved lower recall and precision rates compared to LDA, due to inherent limitations of the mixture approach and the fact that EM algorithm is only locally convergent.

Chapter 8

Classification Results

8.1 Results and Discussion

We evaluated the performance of the proposed method using the standard measures that are used for search strategies in the literature. The **recall** is the ratio of the correctly labeled segments to the total number of relevant segments in the database (i.e., those with the particular label). The **precision** is the ratio of the correctly labeled segments to the total number of segments that the algorithm assigned to the particular label (both correct and incorrect). Both performance measures are expressed as percentages. Overall performance can be expressed as the accuracy over the whole database.

We experimented with the three supervised techniques described in the previous section, GMM, SVM, and LDA. Indeed, these are the dominant supervised techniques encountered in the literature. Since LDA turned out to be the most suited to our problem, we present the LDA experiments first.

The goal of our experiments was to identify the most suitable set of features for segment classification. We compared the classification performance of spatial texture and different color feature representations as described in Section 6.2 by applying the LDA on our labeled set of

segmented images. The classification is hierarchical; each level of hierarchy is handled as an exclusive multi-class classification problem.

In the first experiment, we used the perceptually quantized color features with 15 colors. Thus, each segment was represented by a 21 dimensional feature vector (six spatial textures and 15 colors). The results are shown in Figure 8.1, which shows the recall and precision rates for the most important semantic categories. We then evaluated the classification performance using only the most dominant color expressed in CIE L*a*b* coordinates. In this case, the dimension of the feature vector is three (three dimensions for color component). Figure 8.2 presents the results of this classification. Comparing the recall and precision rates of the two experiments it is clear that using only the most dominant color expressed as an exact coordinate in the L*a*b* color space outperforms the perceptual quantization approach. At first, this appears to be surprising, which is what led us to the statistical analysis presented in Chapter 6. Recall that the key conclusion of our analysis was that a great majority of segments either have only one dominant color or have the second dominant color similar to the first. To make sure that the result is not an artifact of the number of quantization levels, we experimented with different numbers of colors. Figure 8.3 shows the classification results when the number of quantization levels is reduced to eleven. Observe that there is no significant difference between the two quantization schemes. Decreasing the number of representative colors to fewer than 11 resulted in too coarse quantization of the feature vector and a decrease in performance of the classification algorithm. Note also that increasing the number of color quantization levels will not increase the classification performance. This is because increasing the number of quantization levels increases the feature dimensionality in such a way that it creates an artificial distance between perceptually similar colors. This statement has been confirmed by our experiments. We also tried including a second dominant color (in addition to first). This resulted in a modest gain in classification performance, as is evident from Figure 8.4, which shows the results of using two most dominant colors expressed in CIE L*a*b* coordinates. Here, the first dominant color is expressed as a coordinate in L*a*b* color space, while the second

dominant color is expressed as the difference to the first. Finally, we should also note that additional experiments verified that using a third dominant color does not increase performance, and that using a fourth dominant color actually reduces the classification ability of LDA. Figures 8.5 and 8.6 present the classification results using only spatial texture and spatial texture with eleven perceptually quantized colors, respectively. It can be concluded that spatial texture plays an important role in classification feature by itself or in combination with color, resulting in significant improvement in recall and precision rates.

Figure 8.7 and Figure 8.8 show that using spatial texture with the unquantized color value of the first dominant color (in L*a*b* color space) or first and second dominant color outperforms the use of all the dominant colors perceptually quantized. Here, the first dominant color is expressed as an L*a*b* coordinate, while second is expressed as a difference. Note that adding information about the second dominant color improves the classification even further. Figure 8.9 shows the effect of adding the segment position to the feature vector. It is expressed as the centroid normalized by the size of the image. As expected, adding position improves classification performance, especially for separating the sky and water categories. Finally, the last plot shown as the Figure 8.10 presents the result obtained by applying the K-means algorithm followed by LDA, in order to add within-class clusters. This yields a modest improvement in precision (6% on the average), while the recall remains the same.

We now examine the performance of the other two supervised classification techniques. Figures 8.11 and 8.12 show the classification results using the GMM and SVM techniques, respectively. In both cases, we used the texture features, the L*a*b* coordinates of the first and second dominant color, and position. As can be seen from a comparison of these graphs with the one in Figure 8.10, LDA outperforms the other two techniques.

We now turn to the comparison of the proposed approach with the existing literature. Direct comparison of performance on our data set would be difficult as we do not have access to implementations of the other techniques. Other problems are differences in number of semantics

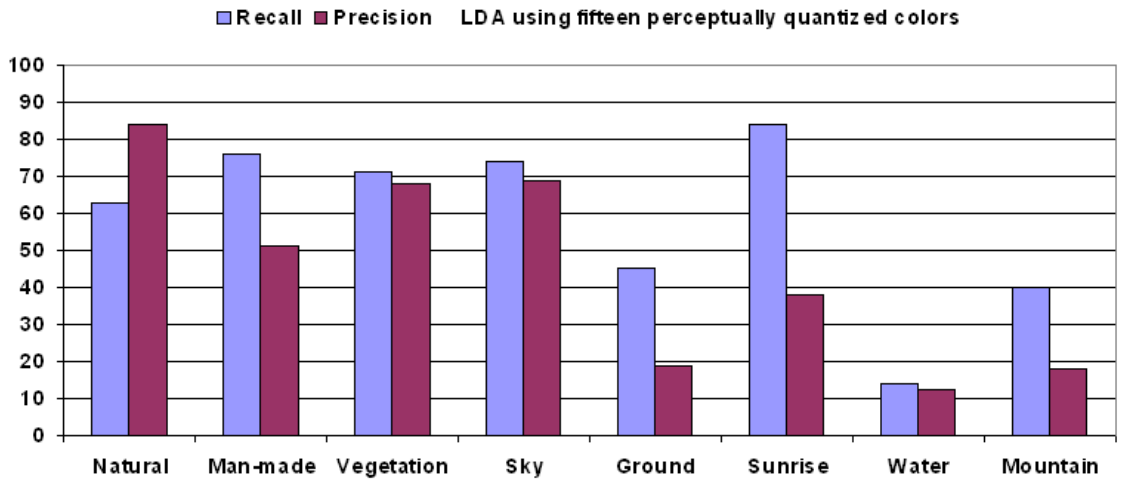


Figure 8.1: Classification results using LDA and fifteen perceptually quantized colors.

labels. The majority of approaches try to infer a larger set of semantic labels, while our aim is to classify the smaller set of significant labels with high precision and recall rates. Still, we give a brief description of the relevant published results for a variety of techniques.

Blobworld [86], which proposes a model that calculates the co-occurrence of blobs and words, achieves a mean precision rate of seven percent and a mean recall of 11 percent. In [149], as in the above mentioned approaches, the sky, water, and people labels have high retrieval rates, the grass label has moderate, while the others are significantly lower. The overall average recall rate is 24 percent and the average precision rate is 14 percent, for a total of 69 concepts. As noted in one of the authors' comments, the principal reason for that is the unavailability of semantically meaningful segmentations.

Reference [85] is an example of an approach that uses a smaller set of labels. The reported rates are quite high, approximately 75 percent recall and 75 percent precision at the optimal operating point, which are comparable to our approach. However, it is important to note here they do not use images containing man-made objects, but rather images containing natural scenes only. Other non-segmentation based algorithms intend to infer the presence of the labels in the image

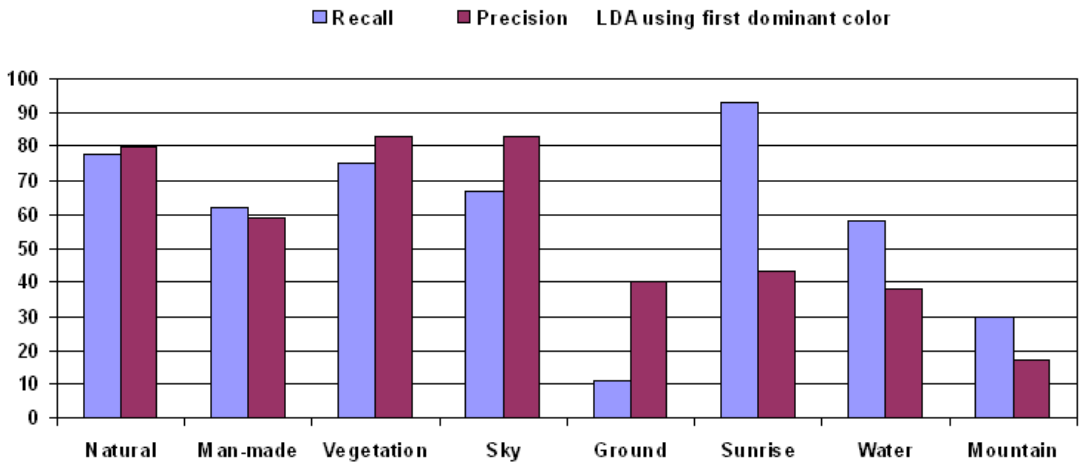


Figure 8.2: Classification results using LDA and first dominant color.

using sophisticated statistical and pattern recognition methods, but although they handle a large vocabulary of labels, their performance is still unsatisfactory.

For example [150] was evaluated on a set of 260 labels, but their average precision rate is 10 percent, while the average recall rate is 9 percent. Although their classification rates are high for the sky, water, people, sun categories, the others are significantly lower. For the 69 concepts with best retrieval rates their average precision rate increases to 33 percent, and the recall rate to 37 percent.

For the LSA based method in [89], the authors evaluated their algorithm using 374 concepts with a total of COREL 5000 images, where 4500 images were used for training and 500 for testing. Each image was uniformly segmented into 16 by 16 pixels grid. Experimental results obtained 25 percent average precision and 27 percent average recall with 133 concepts detected.

In the MBRM method [90], the authors evaluated their algorithm using 260 concepts with a total of COREL 5000 images, where 4000 images were used for training, 500 for a validation set, and 500 for testing. Experimental results obtained 24 percent average precision and 25 percent average recall with 122 concepts detected.

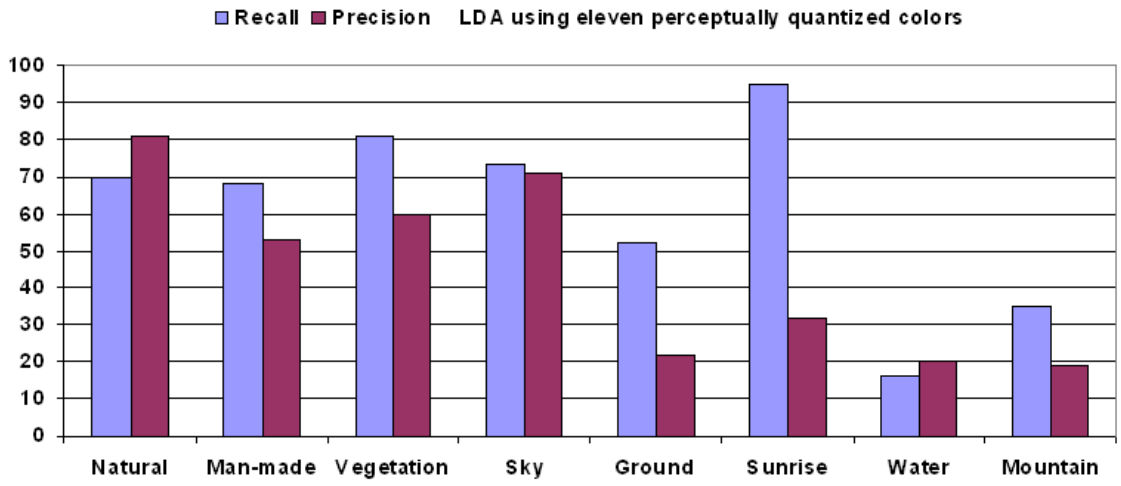


Figure 8.3: Classification results using LDA and eleven perceptually quantized colors.

As we discussed above, the classification techniques presented in this paper are primarily aimed at identifying a limited number of important semantic categories that humans use for image organization and retrieval. Overall, based on the results we have presented in the paper, we believe that the proposed techniques compare favorably to the literature. This includes the approaches we reviewed above [86, 89, 90, 150–152] as well as other references such as [153–155].

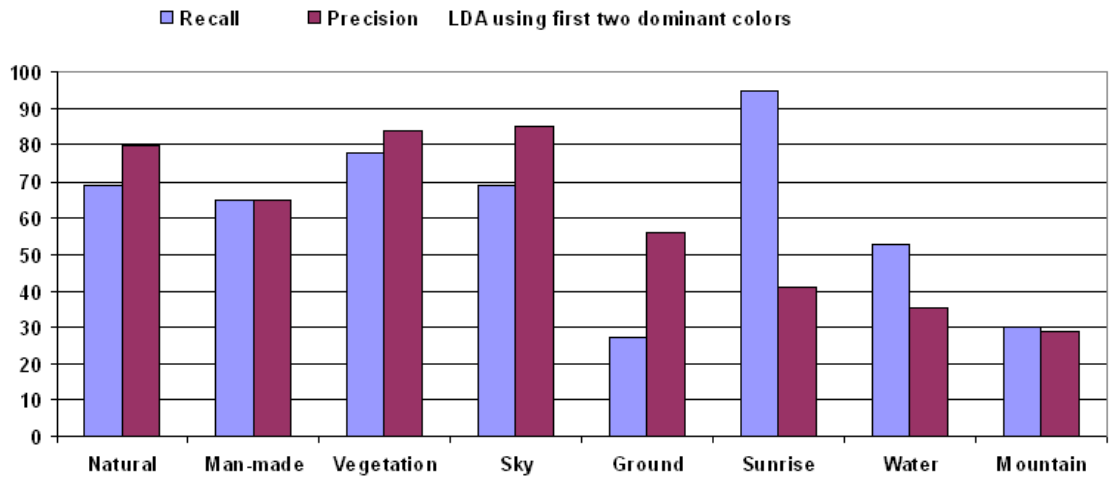


Figure 8.4: Classification results using LDA and first and second dominant color.

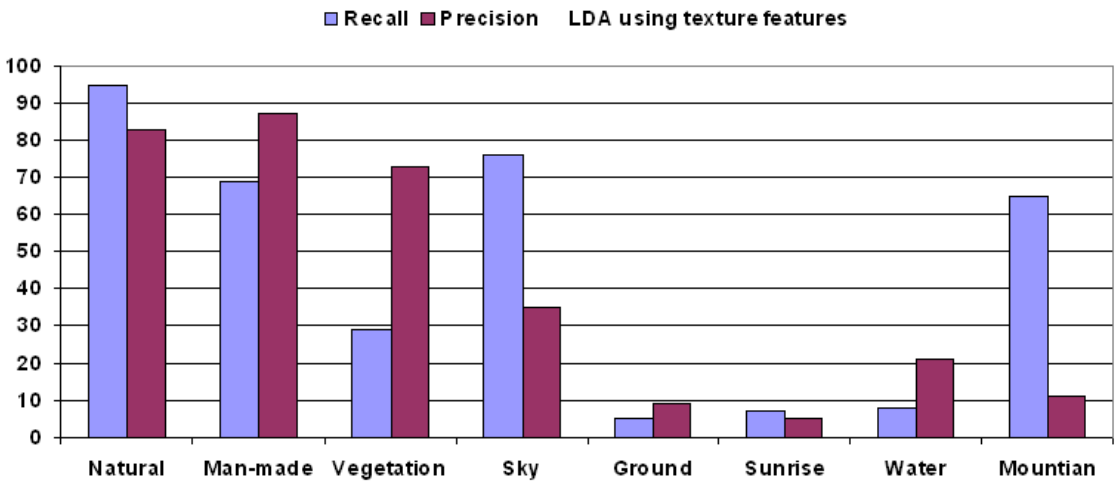


Figure 8.5: Classification results using LDA and texture features.

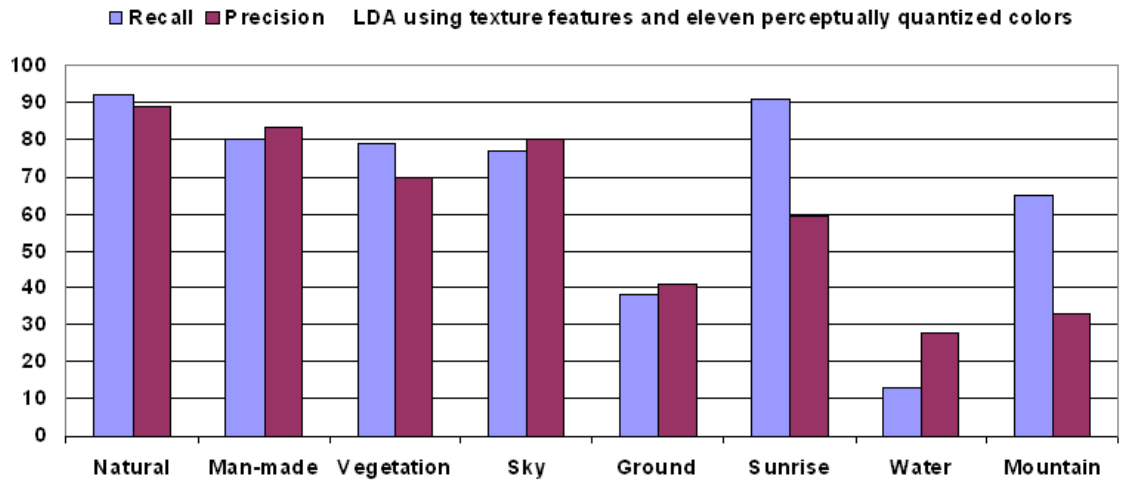


Figure 8.6: Classification results using LDA and eleven perceptually quantized colors and texture features.

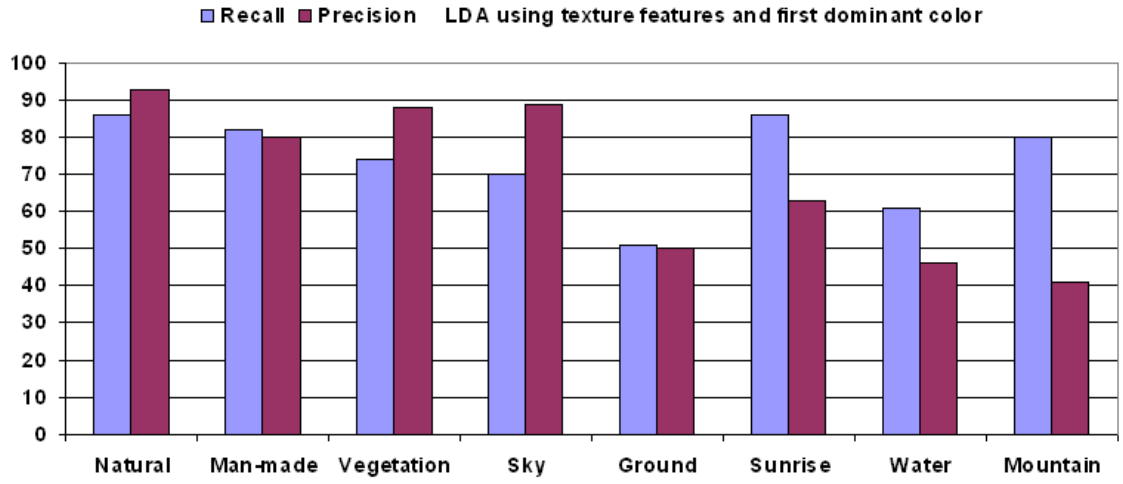


Figure 8.7: Classification results using LDA and first dominant color and texture features

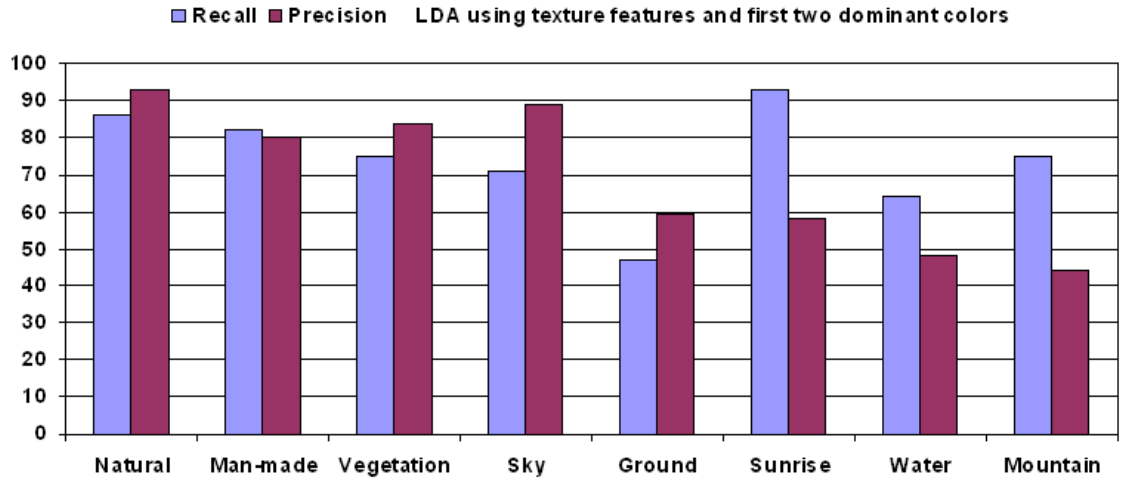


Figure 8.8: Classification results using LDA and first and second dominant color and texture features

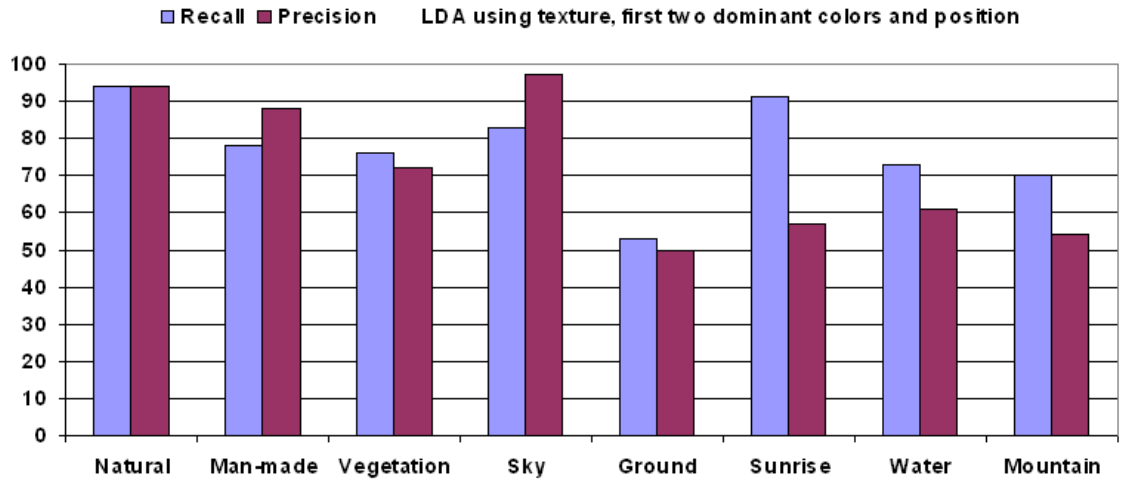


Figure 8.9: Classification results using LDA and first and second dominant color, texture features and position

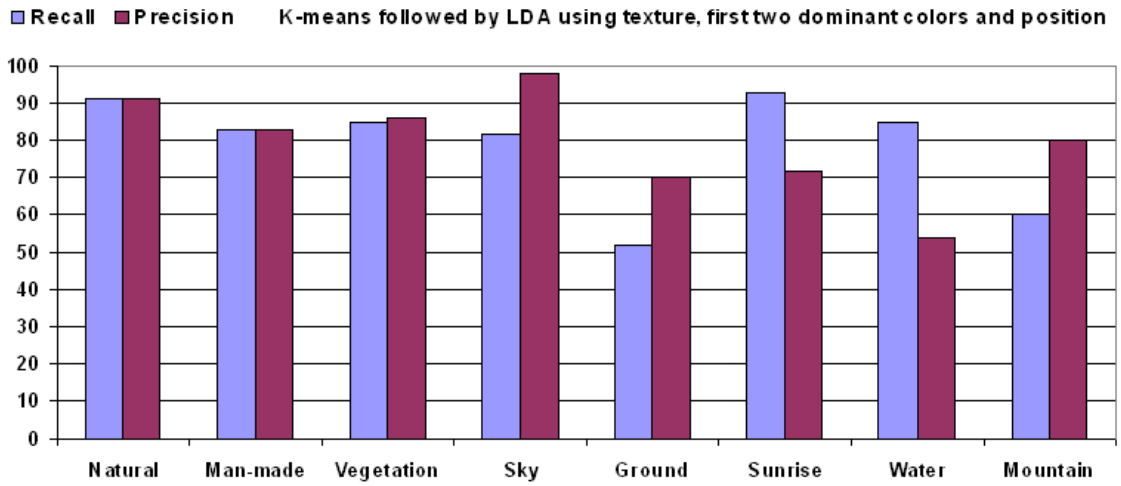


Figure 8.10: Classification results using LDA and first and second dominant color, texture features, position and K-means preprocessing

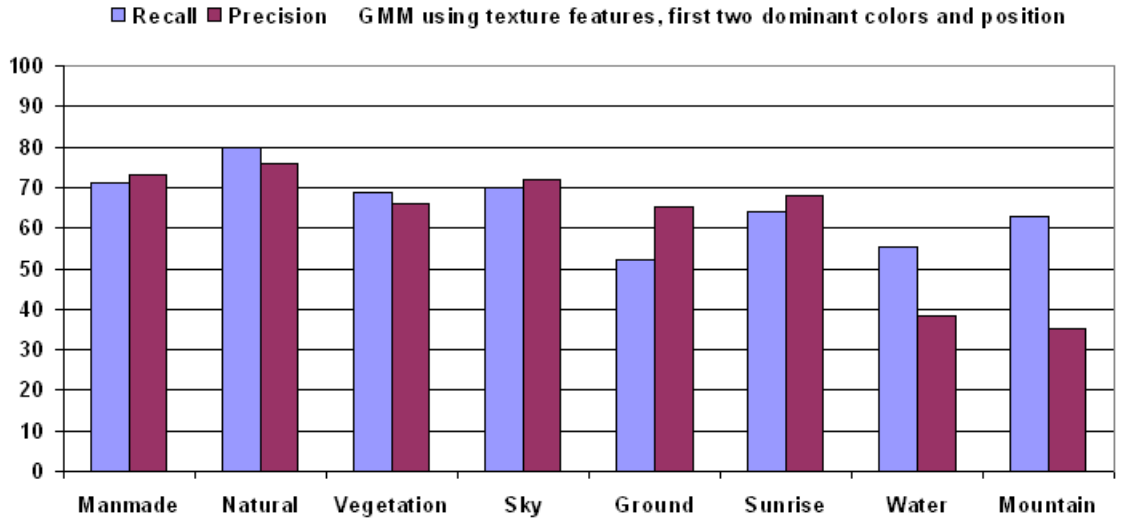


Figure 8.11: Classification results using the GMM approach with texture features, the L*a*b* coordinates of the first and second dominant colors, and position

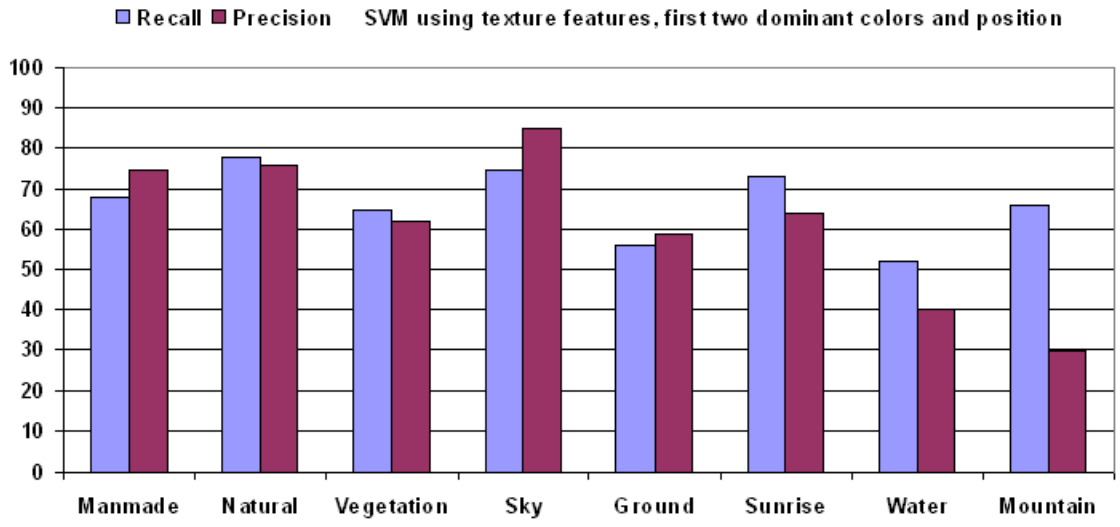


Figure 8.12: Classification results using the SVM approach with texture features, the L*a*b* coordinates of the first and second dominant colors, and position

Chapter 9

Classification and Feature Evaluation in Terms of Human Segmentations

9.1 Introduction

In the previous chapters we showed how the feature selection and the performance of the classification algorithms were based on the segment statistics. In this chapter we investigate the dependence of the segment statistics on the segmentation algorithm. For this, we extract and compare statistics of the segment features obtained using the Chen et al. algorithm to those that correspond to human segmentations. Our findings indicate that although segmentations of the same image by different humans appear to be quite different, the resulting statistics are consistent. Moreover, the statistics are similar to those obtained when the automatic segmentation algorithm is used.

9.2 Segment Statistics for Natural Images

In this section we compare the segment statistics that are obtained from automatic and human segmentations. Figure 4.3 shows several images segmented using the algorithm in Ref. 18. Figure 9.1

shows two images from the UCB database segmented by different humans. Observe that there are substantial differences in the segmentations. Martin *et al.* [156] have shown that, if we allow for mutual refinement, the segmentations by different humans are consistent. What is of more interest to us is whether the segment statistics for the human segmentations are significantly different to those obtained from the human segmentations.

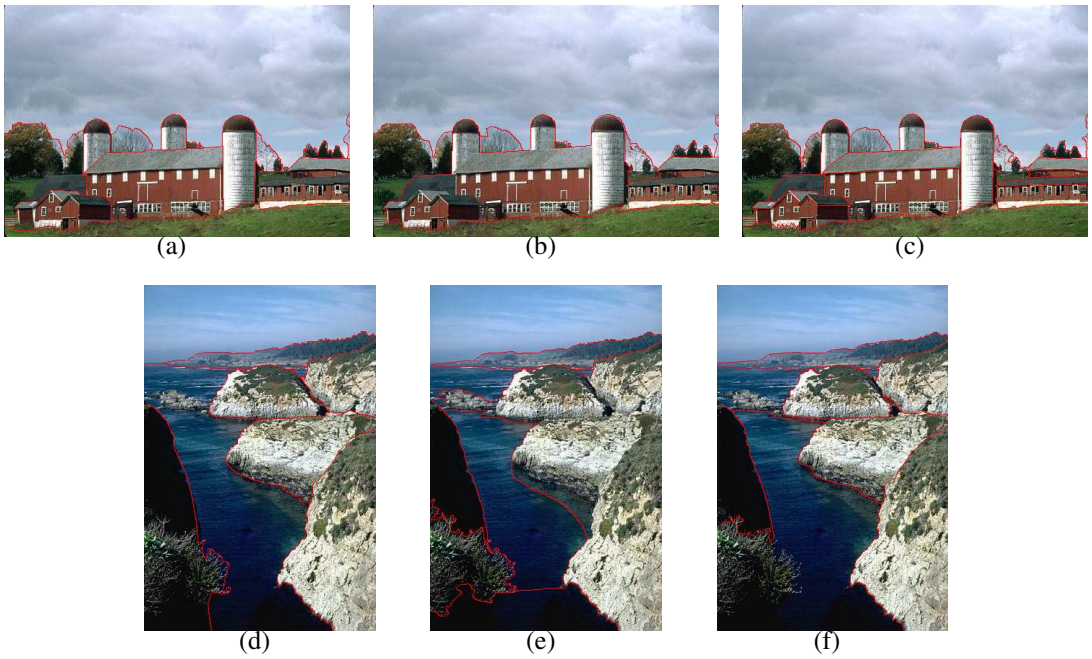


Figure 9.1: Human Segmentations

Automatic segmentations were performed on a image database containing approximately 3300 photographs. The majority of the images were obtained from the Corel Stock Photo Library. Additional images were obtained from a Key Photos Library and the investigators' personal repository. The images in the database cover a variety of outdoor scenes, with a wide range of themes. The images were segmented using the algorithm in Ref. 18 described above. The resulting segments were manually labeled to be used as the ground truth for collecting statistics and supervised learning. Each segment was assigned exactly one label. Segments whose area was less than three percent of total image area were not considered.

Human segmentations were obtained from the University of California at Berkeley (UCB) segmentation database [156]. Martin et. al. constructed this segmentation database by selecting the a representative (RGB) images from the Corel image database, which is widely used in computer vision. Selected images were images of a natural scenes that contain at least one discernible object. Images that are inappropriate for the task of recognition, such as photographs of reflections of neon signs on wet concrete sidewalks, or photographs of marble textures were excluded.

Subjects performed the segmentation task using the computer, and in order to collect segmentations from a wide range of people, authors developed a Java application that subjects can use to divide an image into segments, where a segment is simply a set of pixels. Subjects can then split or merge resulting segments, as well as transfer pixels between any two existing segments.

In order to preserve only the variation among human segmentations of an image due to different perceptual organizations of the scene, and minimize variation among human segmentations due to different interpretations of the task and experimental setup, the instructions were made intentionally vague in an effort to cause the subjects to break up the scene in a ”natural” manner. Subjects were instructed to divide each image into pieces, where each piece represents a distinguished thing in the image. The instructions to the human subjects made no attempt to restrict or encourage the use of any particular type of cues. The number of resulting segments was left up to the subjects. The subjects were also provided with several example segmentations of simple, unambiguous images as a visual description of the task.

Images were assigned to subjects randomly with a bias towards images that had been segmented by some other subject. In addition, no subject segmented the same image twice, no image was segmented by more than 5 people, and no two images were segmented by exactly the same set of subjects.

We selected a subset of a database containing approximately segmented 400 images, with approximately 1600 labeled segments. Each image was segmented by an average of five subjects. There was a total of 30 subjects. For reasons that we will explain in Section 6.3, we did not consider

segments containing humans or animals. As with the automatic segmentations, the segments were manually labeled; each segment was assigned exactly one label.

The statistics for automatic segmentations using the Chen *et al.* algorithm [18] are shown in the left column of Fig. 9.2, which shows histograms for the first, second, third, and fourth dominant color across all segments in the database. The horizontal axis represents the percentage of the area that the corresponding dominant color occupies in a segment, while the vertical axis represents the probability of occurrence for each bin. Based on these statistics, the great majority of segments could be described by the first two dominant colors with very little loss of information. We also looked at the distance between the dominant colors. The histogram of the distance between the first and second dominant colors is shown at the top left of Fig. 9.3, and the distance between the first and third is shown at the right. Observe that, in the majority of cases, the second dominant color is less than twenty units away from the first. Here we should note that colors with $L^*a^*b^*$ distance of 10 units are quite similar. This means that for a great majority of segments the second dominant color is similar to the first. This explains why using the $L^*a^*b^*$ value of the first dominant color gives better classification results than using the 15 quantized colors, and also why adding the second dominant color improves performance by only a small amount.

The corresponding statistics for the human segmentations are shown in Figures 9.2 and 9.3. Note that, with the exception of the 100% bin for the first dominant color, the statistics are quite similar. A similar observation holds for the distribution of the distance between the first and second dominant colors and the distance between the first and third dominant colors. Thus, we can safely assume that ignoring the second and third dominant color can be safely ignored in segment classification. Finally, we should point out that the two sets of statistics are quite similar, even though they obtained over different sets of images.

9.3 Classification Results

In addition to the segment statistics, it is interesting to investigate whether the performance of the classification algorithm can improve if human segmentations are used instead of the automatic ones, or if it degrades due to the fact that the features we are using are matched to the segmentation algorithm and do not work with the human segments. For this, we compared the classification performance of the spatial texture and color composition features described in Section 6.2 on both human and automatic segmentations. We applied LDA to the same labeled sets of segmented images that we used for the segment statistics in Section 9.2. In both cases we used 80% of the segments for training and the rest for testing.

We evaluated the performance of the classification techniques using the standard measures that are used for evaluating search strategies in the literature. The **recall** is the ratio of the correctly labeled segments to the total number of relevant segments in the database (i.e., those with the particular label). The **precision** is the ratio of the correctly labeled segments to the total number of segments that the algorithm assigned to the particular label (both correct and incorrect). Both performance measures are expressed as percentages. Overall performance can be expressed as the accuracy over the whole database.

The results are shown in Fig. 9.4(b) for the segment classification using spatial texture features, first two dominant colors and position, and Fig. 9.4(b) using same set of features and the K-means preprocessing. We should note that the "Sunrise/Sunset" category was omitted from the human segmentations results because of insufficient number of samples for classification.

Comparing the recall and precision rates from the two experiments we can see that on the average they are approximately the same. Experiments with different sets of features (one most dominant color, 15 quantized colors, etc.) and resulted in the same conclusion. We also verified that using a third dominant color does not increase performance, and that using a fourth dominant color actually reduces the classification ability of LDA.

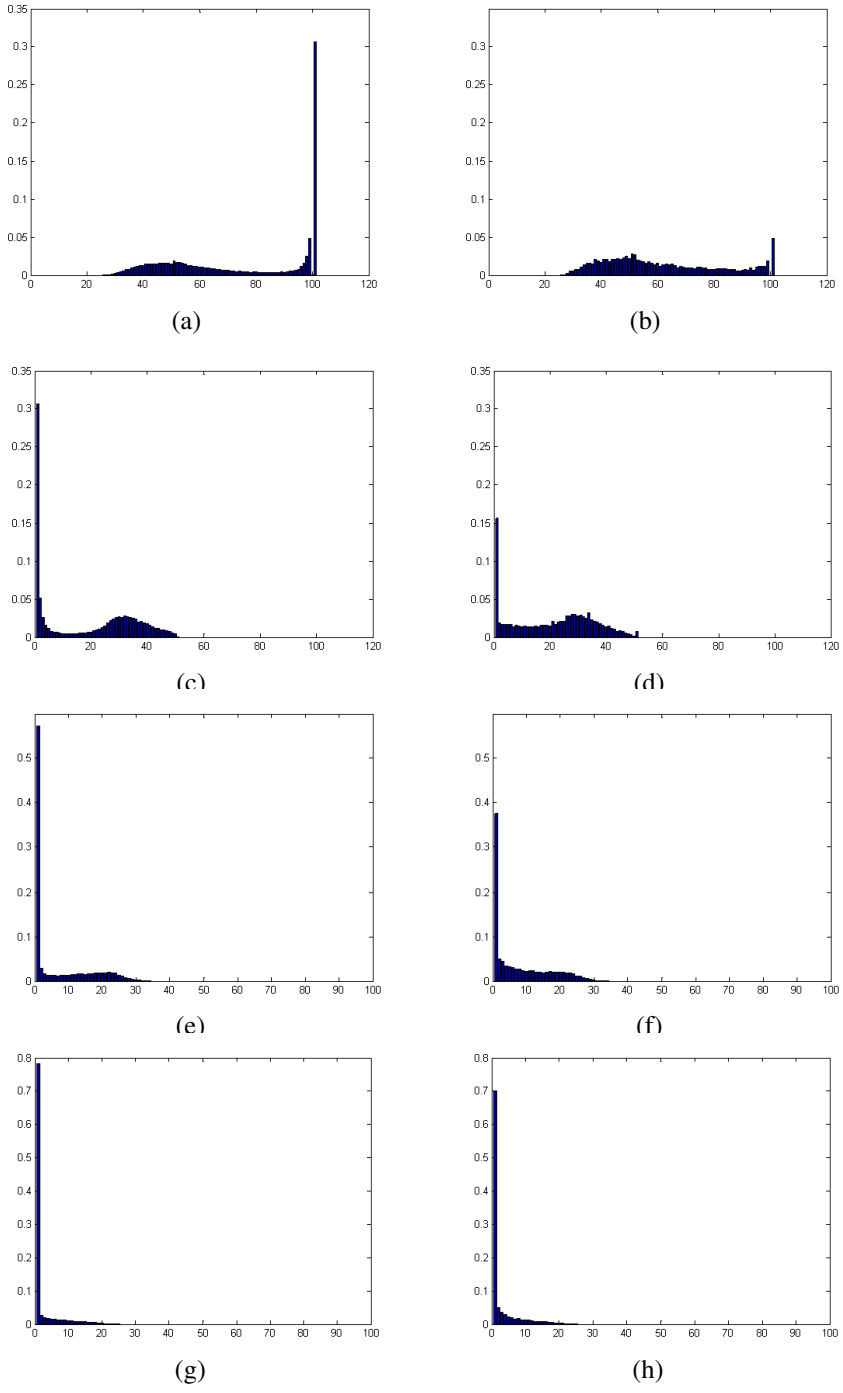


Figure 9.2: Statistics of dominant colors. Left column: automatic segmentations. Right column: human segmentations. The horizontal axis represents the percentage of the area that the dominant color occupies in a segment and the vertical axis represents the probability of occurence for each bin.

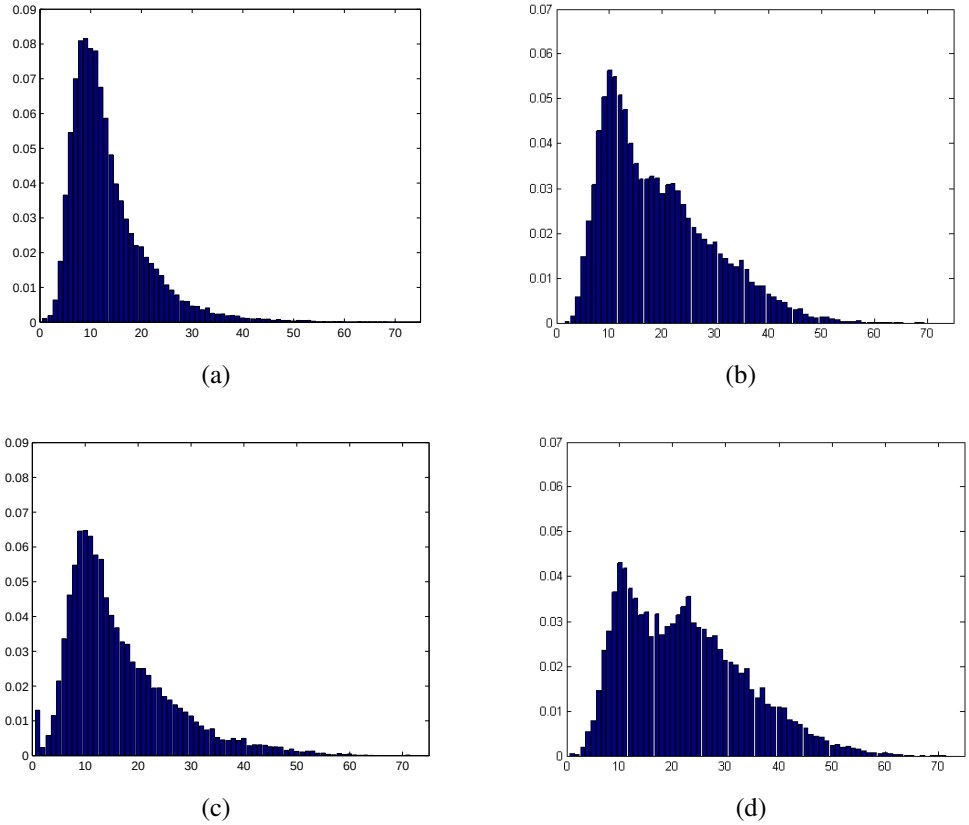
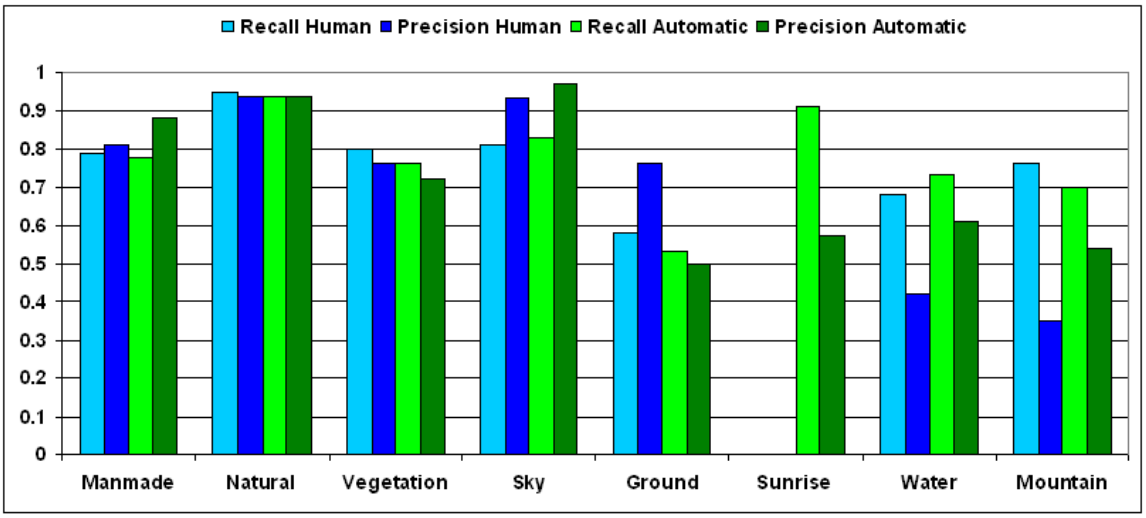
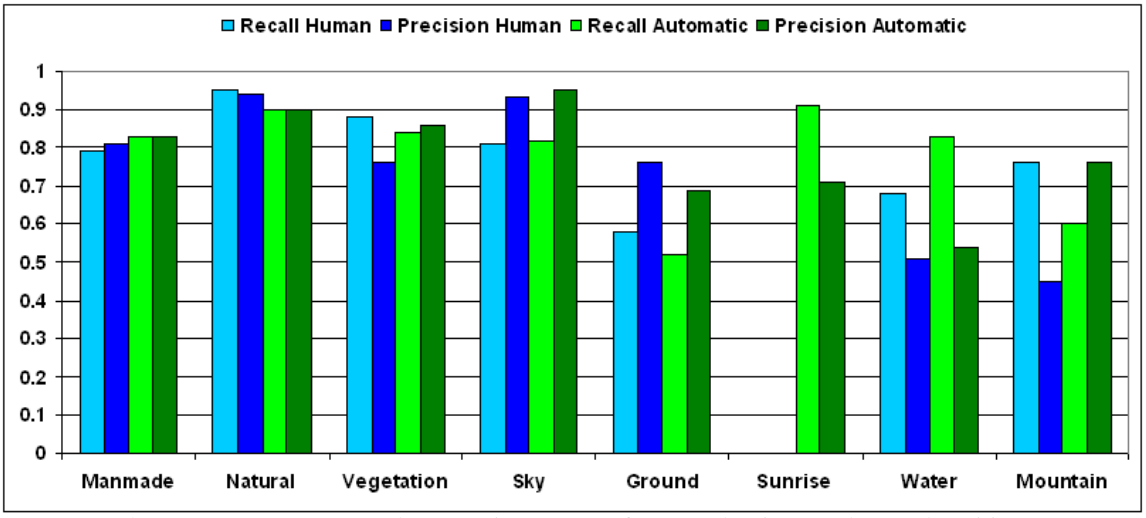


Figure 9.3: Distances between dominant colors in *L*a*b** color space



(a) LDA using texture, first two dominant colors and position



(b) K-means preprocessed LDA using texture, first two dominant colors and position

Figure 9.4: Comparison of classification results using human and automatic segmentations

Chapter 10

Summary, Conclusions, and Future Work

This dissertation addresses important aspects for Content Based Image Retrieval: derivation of medium level descriptors through spatially adaptive color texture segmentation that divides the image into perceptually uniform regions (segments) and extraction and representation of segment wide features for classification.

Background material is presented in Chapter 2. In addition to discussion of user needs for semantic indexing, an overview of Content Based Retrieval highlights the state of the art in the field. Important concepts of semantic gap is introduced in greater detail.

In Chapter 3, we present an overview of perceptually tuned adaptive color-texture segmentation algorithm recently proposed by Chen and Pappas. This segmentation algorithm combines the color composition and spatial-texture features to obtain segments of uniform texture. Several critical parameters of the texture features and this segmentation algorithm were determined by subjective tests.

Low level features used are introduced in Chapter 4. This chapter presents the color, texture and other features in greater detail.

Semantic label selection is presented in Chapter 5.

Chapter 6, describes how proposed features were extracted from segments and their repre-

sentation. Segment statistics of dominant colors is also presented in this chapter.

Verification database and ground truth generation were presented in Chapter 7. In addition, this chapter presents clustering, classification and pattern recognition algorithms used in classification, their tradeoffs and numerical implementations.

Experimental results were presented in Chapter 8. Recall and precision rates obtained by using the different feature representations and applying the classification techniques were presented and compared to the literature.

Feature Evaluation in Terms of Human Segmentations and statistics of human segmentations were presented in Chapter 9.

In Conclusion, we presented a new approach for semantic classification that utilizes perceptual models for image segmentation and classification. The main innovations of the proposed approach are the use of an algorithm that produces perceptually uniform segments and the selection of perceptually-motivated region-wide color and texture features. The features of these regions are used as medium level descriptors and are the key to bridging the gap between low-level image primitives and high-level image semantics. Our results indicate that the proposed approach offers significant performance improvements over the existing literature.

10.1 Future Work

In closing, we suggest possible future research directions based on the work presented here. First we consider extensions of the current classification scheme onto extraction of semantic labels at the image (scene) level. Then we present some thoughts on development of novel classification techniques and possible image query types.

10.1.1 Extracting Semantic Labels at the Image (Scene) Level

The classification of segments can be further extended to full semantic scene classification. This can be achieved by incorporating the relative location and relationship of the various segments within the image. A number of approaches can be used for this, including Bayesian inference, layout models, region adjacency graphs, entropy models, and Hidden Markov Models. For example, by exploiting the layout and mutual relationship among segments we expect that we will be able to classify the image as an indoor or outdoor scene. Similarly, we should be able to identify marine scenes, landscapes, garden scenes (bright colors), etc. A marine scene that also contains sand may be further classified as a beach scene. Such an approach could then be coupled with the existing face/person detection algorithms to yield a rich vocabulary of labels describing a particular image.

10.1.2 Development of Novel Classification Techniques

The primary technique that we have used so far for training and classification is the Linear Discriminant Analysis (LDA) method. We have tried a number of other techniques, but LDA is so far the best. LDA belongs to the class of linear classifiers, which try to find a subspace projection such that samples from the different classes are well separated, i.e., to find directions in the data space that facilitate data classification. This is done by finding a direction that maximizes the variance between the class means, and at the same time minimizes the variance within each

class. LDA is fast, has the ability to handle non-Gaussian cluster distributions, and the discriminatory dimensions are expressed as vectors. In addition, when each semantic class consists of two or more clusters, K-means followed by LDA can be used to separate the different clusters that belong to the same class. This provides lot of room for improvement since we do not care about the overlap of the clusters that belong to the same class. Based on such observations, we believe that by posing the optimization function in a slightly different way we may be able to achieve large gains in classification ability. Finally, we also plan to further improve the performance of segment classification by developing boundary shape features, as well as exploiting the properties of the neighboring segments.

10.1.3 Image Query Types

The proposed approach is designed for queries based on a set of predefined labels. However, the approach can be extended to query by example. As the proposed techniques will be able to extract semantic labels from an image, query by example will require the extraction of labels for both the query image and the images in the database. A simple matching of labels will then complete the query. Furthermore since low level descriptors are recorded in the XML (extended markup language) file structure, it could be also possible for the user to query an image with the particular shade of color or texture, as well as full low level query based on color and texture only.

10.1.4 Need for Benchmarking Image Retrieval Databases and Ontologies

During the course of this thesis research, we found it to be very difficult to directly compare retrieval and precision performance with the literature. The main difficulty is the lack of precisely defined sets of ontologies (formal description or specification of the concepts and relationships that can exist for an agent or a community of agents) for the semantic concepts used in image analysis and retrieval. Often, different researchers use the same label in a different meaning. For

example the definition of cityscape can sometimes be quite lenient and left at the discretion of the investigator. Ontological commitments by researchers would allow the use of shared vocabulary in a coherent and consistent manner.

Benchmarking results are further complicated by the lack of a common test database. Although the majority of researchers use the Corel Stock Photo Library, there are credible doubts about the usability and bias of this database. For example, sometimes several images of the same subject are taken from a slightly different angle or are similar in some other way.

Such shortcomings have also been encountered in the related field of video retrieval, and work is already under way to improve benchmarking. The goal of the TRECVID workshops, which are sponsored by the National Institute of Standards and Technology (NIST), is to facilitate research in information retrieval by providing a large collection of test data, uniform scoring procedures, and a forum for researchers interested in comparing their results. Such efforts were met with great anticipation and widespread acceptance among researchers. Unfortunately, similar benchmarks and procedures are not yet available in the image analysis and retrieval field.

References

[1] A. Mojsilovic and B. Rogowitz, "A psychophysical approach to modeling image semantics," in *Human Vision and Electronic Imaging VI* (B. E. Rogowitz and T. N. Pappas, eds.), vol. Proc. SPIE Vol. 4299, (San Jose, CA), pp. 470–477, Jan. 2001.

[2] D. Petkovic, "Query by image content," in *Storage and Retrieval for Image and Video Databases IV* (I. K. Sethi and e. Jain, R C, eds.), vol. Proc. SPIE Vol. 2670, (San Jose, CA), Jan. 1996.

[3] A. e. a. Kitamoto, "Similarity retrieval of noaa satellite imagery by graph matching," in *Storage and Retrieval for Image and Video Databases* (W. R. Niblack and e. Jain, R C, eds.), vol. Proc. SPIE Vol. 1908, (San Jose, CA), pp. 60–73, Jan. 1993.

[4] A. Soffer and H. Samet, "Retrieval by content in symbolic image databases," in *Storage and Retrieval for Image and Video Databases IV* (I. K. Sethi and e. Jain, R. C., eds.), vol. Proc. SPIE Vol. 2670, (San Jose, CA), pp. 144–155, Jan. 1996.

[5] M. W. Y. and B. S. Manjunath, "A texture thesaurus for browsing large aerial photographs," *Journal of the American Society for Information Science*, vol. 49, no. 7, pp. 633–648, 1998.

[6] L. C. S. et. al., "S-stir: Similarity search through iterative refinement," in *Storage and Retrieval for Image and Video Databases VI* (I. K. Sethi and e. Jain, R. C., eds.), vol. Proc. SPIE Vol. 3312, (San Jose, CA), pp. 250–258, Jan. 1998.

[7] K. T., "Database architecture for content-based image retrieval," in *Image Storage and Retrieval Systems* (A. A. Jambardino and e. Niblack, W. R., eds.), vol. Proc. SPIE Vol. 1662, (San Jose, CA), pp. 112–123, Jan. 1992.

[8] S. Santini and R. C. Jain, "The graphical specification of similarity queries," *Journal of Visual Languages and Computing*, vol. 7, pp. 403–421, 2005.

[9] B. E. Rogowitz and L. Treinish, "Data visualization: the end of the rainbow," *IEEE Spectrum*, pp. 52–59, Dec. 1998.

[10] A. Mojsilović and B. E. Rogowitz, "Semantic metric for image library exploration," *IEEE Trans. Multimedia*, vol. 6, pp. 828–838, Dec. 2004.

[11] Y. Rui, T. S. Huang, and S.-F. Chang, "Image retrieval: Current techniques, promising directions and open issues," *J. Visual Communication and Image Representation*, vol. 10, pp. 39–62, Mar. 1999.

[12] W. M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain, "Content-based image retrieval at the end of the early years," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 22, pp. 1349–1379, Dec. 2000.

[13] J. Chen, T. N. Pappas, A. Mojsilovic, and B. E. Rogowitz, "Adaptive image segmentation based on color and texture," in *Proc. Int. Conf. Image Processing (ICIP-02)*, vol. 2, (Rochester, NY), pp. 789–792, Sept. 2002.

[14] J. Chen, T. N. Pappas, A. Mojsilovic, and B. E. Rogowitz, "Perceptual color and texture features for segmentation," in *Human Vision and Electronic Imaging VIII* (B. E. Rogowitz and T. N. Pappas, eds.), vol. Proc. SPIE Vol. 5007, (Santa Clara, CA), pp. 340–351, Jan. 2003.

[15] J. Chen, T. N. Pappas, A. Mojsilovic, and B. E. Rogowitz, "Image segmentation by spatially adaptive color and texture features," in *Proc. Int. Conf. Image Processing (ICIP-03)*, vol. 1, (Barcelona, Spain), pp. 1005–1008, Sept. 2003.

[16] J. Chen, T. N. Pappas, A. Mojsilovic, and B. E. Rogowitz, "Perceptually-tuned multiscale color-texture segmentation," in *Proc. Int. Conf. Image Processing (ICIP-04)*, (Singapore), pp. 921–924, Oct. 2004.

[17] J. Chen and T. N. Pappas, "Experimental determination of visual color and texture statistics for image segmentation," in *Human Vision and Electronic Imaging X* (B. E. Rogowitz, T. N. Pappas, and S. J. Daly, eds.), vol. Proc. SPIE Vol. 5666, (San Jose, CA), pp. 227–236, Jan. 2005.

[18] J. Chen, T. N. Pappas, A. Mojsilovic, and B. E. Rogowitz, "Adaptive perceptual color-texture image segmentation," *IEEE Trans. Image Processing*, vol. 14, pp. 1524–1536, Oct 2005.

[19] S. Ornager, "Image retrieval: theoretical and empirical user studies on accessing information in images," in *proceedings of the 60th ASIS Annual Meeting*, vol. 34, pp. 202–211.

[20] M. Markkula and E. Sormunen, "Searching for photos journalists practices in pictorial ir," in *The Challenge of Image Retrieval research workshop*, (Newcastle upon Tyne), Feb. 1998.

[21] P. G. B. Enser and C. G. McGregor, "Analysis of visual information retrieval queries," in *British Library Research and Development Report*, vol. 6104.

[22] E. P. G. B., "Pictorial information retrieval," *Journal of Documentation*, vol. 51, no. 2, pp. 126–170, 1995.

[23] L. H. Keister, "User types and queries: impact on image access systems," in *Challenges in indexing electronic text and images* (e. Fidel, R et al., ed.), vol. proceedings of the 55th ASIS Annual Meeting, pp. 7–22.

[24] L. Armitage and P. G. B. Enser, "Analysis of user need in image archives," *Journal of Information Science*, vol. 23, no. 4, pp. 278–299, 1997.

[25] H. McCorry and I. O. Morrison, *Report on the Catechism project*. National Museums of Scotland, 1995.

[26] J. Sledge, "Points of view," in *Multimedia Computing and Museums: selected papers from the 3rd International conference on Hypermedia and Interactivity in Museums* (e. Bearman, D, ed.), vol. ICHIM95 / MCN95, (San Diego, California), pp. 335–346, Oct. 1995.

[27] J. H. E. van der Starre, "Ceci nest pas une pipe: indexing of images," in *Multimedia Computing and Museums: selected papers from the 3rd International conference on Hypermedia and Interactivity in Museums* (e. Bearman, D, ed.), vol. ICHIM95 / MCN95, (San Diego, California), pp. 267–277, Oct. 1995.

[28] S. Shatford Layne, "Some issues in the indexing of images," *Journal of the Amercan Society of Information Science*, vol. 45, no. 8, pp. 583–588, 1994.

[29] P. Constantopulos and M. Doerr, "An approach to indexing annotated images," in *Multimedia Computing and Museums: selected papers from the 3rd International conference on Hypermedia and Interactivity in Museums* (e. Bearman, D, ed.), vol. ICHIM95 / MCN95, (San Diego, California), pp. 278–298, Oct. 1995.

[30] K. Markey, "Access to iconographical research collections," *Library Trends*, vol. 37, no. 2, pp. 154–174, 1988.

[31] J. P. Eakins, "Automatic image content retrieval are we getting anywhere?," in *Proceedings of Third International Conference on Electronic Library and Visual Information Research* (e. Milton Keynes, ed.), vol. 3, pp. 123–135, 1996.

[32] G. V. N. and R. V. V., "Content-based image retrieval systems," *IEEE Computer*, vol. 28, no. 9, pp. 18–22, 1995.

[33] M. Stricker and M. Orengo, "Similarity of color images," in *Storage and Retrieval for Image and Video Databases III* (W. R. Niblack and e. Jain, R C, eds.), vol. Proc SPIE 2420, pp. 381–392, 1995.

[34] M. Stricker and A. Dimai, "Color indexing with weak spatial constraints," in *Storage and Retrieval for Image and Video Databases IV* (I. K. Sethi and e. Jain, R C, eds.), vol. Proc SPIE 2670, pp. 29–40, 1996.

[35] C. C. S., "Region-based image querying," in *Proceedings of IEEE Workshop on Content-Based Access of Image and Video Libraries*, (San Juan, Puerto Rico), pp. 42–49, 1997.

[36] H. e. a. Tamura, "Textural features corresponding to visual perception," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 8, no. 6, pp. 460–472, 1978.

[37] F. Liu and R. W. Picard, "Periodicity, directionality and randomness: Wold features for image modelling and retrieval," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 18, no. 7, pp. 722–733, 1996.

[38] B. S. Manjunath and W. Y. Ma, "Texture features for browsing and retrieval of large image data," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 18, pp. 837–842, 1996.

[39] L. M. e. a. Kaplan, "Fast texture database retrieval using extended fractal features," in *Storage and Retrieval for Image and Video Databases VI* (I. K. Sethi and e. Jain, R C, eds.), vol. Proc SPIE 3312, pp. 162–173, 1998.

[40] I. Biederman, "Recognition-by-components: a theory of human image understanding," *Psychological Review*, vol. 94, no. 2, pp. 115–147, 1987.

[41] W. e. a. Niblack, "The qbic project: querying images by color, texture and shape," *IBM Research Report*, no. RJ-9203, 1993.

[42] R. Mehrotra and J. E. Gary, "Similar-shape retrieval in shape data management," *IEEE Computer*, vol. 28, no. 9, pp. 57–62, 1995.

[43] P. A. et. al., "Photobook: tools for content-based manipulation of image databases," *International Journal of Computer Vision*, vol. 18, no. 3, pp. 233–254, 1996.

[44] A. e. a. del Bimbo, "Image retrieval by elastic matching of shapes and image patterns," in *Proceedings of Multimedia*, pp. 215–218, 1996.

[45] A. K. Jain and A. Vailaya, "Image retrieval using color and shape," *IEEE Transation on Pattern Recognition*, vol. 29, no. 8, pp. 1233–1244, 1996.

[46] D. e. a. Androutsas, "Image retrieval using directional detail histograms," in *Storage and Retrieval for Image and Video Databases VI*, vol. Proc SPIE 3312, pp. 129–137, 1998.

[47] B. B. e. a. Kimia, "A shock-based approach for indexing of image databases using shape," in *Multimedia Storage and Archiving Systems II* (e. Kuo, C C J et al, ed.), vol. Proc SPIE 3229, pp. 288–302, 1997.

[48] Y. Chan and S. Y. Kung, "A hierarchical algorithm for image retrieval by sketch," in *First IEEE Workshop on Multimedia Signal Processing*, vol. Proc SPIE 3229, pp. 564–569, 1997.

[49] J. L. Chen and C. C. Stockman, "Indexing to 3d model aspects using 2d contour features," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, (San Francisco), pp. 913–920, 1996.

[50] D. S. et. al., "Viewpoint-invariant indexing for content-based image retrieval," in *IEEE International Workshop on Content-based Access of Image and Video Databases*, vol. CAIVD98, (Bombay, India), pp. 20–30, 1998.

[51] M. e. a. Chock, "Database structure and manipulation capabilities of the picture database management system picdms," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 6, no. 4, pp. 484–492, 1984.

[52] N. e. a. Roussopoulos, "An efficient pictorial database system for psql," *IEEE Transactions on Software Engineering*, vol. 14, no. 5, pp. 639–650, 1988.

[53] S. K. e. a. Chang, "An intelligent image database system," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 5, pp. 681–688, 1988.

[54] S. K. Chang and E. Jungert, "Pictorial data management based upon the theory of symbolic projections," *Journal of Visual Languages and Computing*, vol. 2, pp. 195–215, 1991.

[55] G. V. N. and R. V. V., "Design and evaluation of algorithms for image retrieval by spatial similarity," *ACM Transactions on Information Systems*, vol. 13, no. 2, pp. 115–144, 1995.

[56] S. J. R. and C. S. F., "Querying by color regions using the visualseek content-based visual query system," in *Intelligent Multimedia Information Retrieval* (e. Maybury, M T, ed.), (Menlo Park, CA), pp. 23–41, 1997.

[57] Y. T. e. a. Hou, "A content-based indexing technique using relative geometry features," in *Image Storage and Retrieval Systems*, vol. Proc SPIE 1662, pp. 59–68, 1992.

[58] C. E. E. a. Jacobs, "Fast multiresolution image querying," in *Proceedings of ACM SIGGRAPH 95*, vol. Proc SPIE 1662, (Los Angeles, CA), pp. 277–286, 1995.

[59] K. C. Liang and C. C. J. Kuo, "Implementation and performance evaluation of a progressive image retrieval system," in *Storage and Retrieval for Image and Video Databases VI* (I. K. Sethi and e. Jain, R C, eds.), vol. Proc SPIE 3312, pp. 37–48, 1998.

[60] S. Ravela and R. Manmatha, "Retrieving images by appearance," in *IEEE International Conference on Computer Vision (IICV98)*, (Bombay, India), pp. 608–613, 1998.

[61] S. Ravela and R. Manmatha, "On computing global similarity in images," in *IEEE Workshop on Applications of Computer Vision (WACV98)*, (Princeton, NJ), pp. 82–87, 1998.

[62] F. Rabbitti and P. Stanchev, "Grim dbms: a graphical image database management system," in *Visual Database Systems* (e. Kunii, T, ed.), (Elsevier, Amsterdam), pp. 415–430, 1989.

[63] T. e. a. Hermes, "Image retrieval for information systems," in *Storage and Retrieval for Image and Video Databases III* (W. R. Niblack and e. Jain, R C, eds.), vol. Proc SPIE 2420, (Elsevier, Amsterdam), pp. 394–405, 1995.

[64] A. e. a. Oliva, "Real-world scene categorization by a self-organizing neural network," *Perception*, vol. 26, no. 19, 1997.

[65] A. L. Ratan and W. E. L. Grimson, "Training templates for scene classification using a few examples," in *IEEE Workshop on Content-Based Access of Image and Video Libraries*, (San Juan, Puerto Rico), pp. 90–97, June 1997.

[66] R. A. Brooks, "Symbolic reasoning among 3-d models and 2-d images," *Artificial Intelligence*, vol. 17, pp. 285–348, 1981.

[67] J. H. Connell and J. M. Brady, "Generating and generalizing models of visual objects," *Artificial Intelligence*, vol. 31, no. 2, pp. 159–183, 1987.

[68] T. M. Strat and M. A. Fischler, "Context-based vision: recognizing objects using information from both 2-d and 3-d imagery," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 10, pp. 1050–1065, 1991.

[69] T. Minka, "An image database browser that learns from user interaction," *MIT Media Laboratory Technical Report*, no. 365, 1996.

[70] S. F. e. a. Chang, "Semantic visual templates: linking visual features to semantics," in *IEEE International Conference on Image Processing (ICIP98)*, (Chicago, Illinois), pp. 531–535, 1998.

[71] M. e. a. Flickner, "Query by image and video content: the qbic system," *IEEE Computer*, vol. 28, no. 9, pp. 23–32, 1995.

[72] A. e. a. Gupta, "The virage image search engine: an open framework for image management," in *Storage and Retrieval for Image and Video Databases IV*, vol. Proc SPIE 2670, pp. 76–87, 1996.

[73] J. Feder, "Towards image content-based retrieval for the world-wide web," *Advanced Imaging*, vol. 11, no. 1, pp. 26–29, 1996.

[74] R. W. Picard, "A society of models for video and image libraries," *IBM Systems Journal*, vol. 35, pp. 292–312, 1996.

[75] S. J. R. and C. S. F., "An image and video search engine for the world-wide web," in *Storage and Retrieval for Image and Video Databases V* (I. K. Sethi and e. Jain, R C, eds.), vol. Proc SPIE 3022, pp. 84–95, 1997.

[76] S. F. e. a. Chang, "Videoq: an automated content based video search system using visual cues," in *ACM Multimedia 1997*, (Seattle, WA), pp. 313–324, 1997.

[77] M. e. a. Beigi, "Metaseek: a content-based meta-search engine for images," in *Storage and Retrieval for Image and Video Databases VI* (I. K. Sethi and e. Jain, R C, eds.), vol. Proc SPIE 3312, pp. 118–128, 1998.

[78] T. e. a. Huang, "Multimedia analysis and retrieval system (mars) project," in *Digital Image Access and Retrieval: 1996 Clinic on Library Applications of Data Processing* (P. B. Heidorn and e. Sandore, B, eds.), vol. Proc SPIE 3312, (Urbana-Champaign, IL), pp. 101–117, 1997.

[79] Y. e. a. Rui, "Relevance feedback techniques in interactive content-based image retrieval," in *Storage and Retrieval for Image and Video Databases VI* (I. K. Sethi and e. Jain, R C, eds.), vol. Proc SPIE 3312, pp. 25–36, 1998.

[80] H. D. e. a. Wactlar, "Intelligent access to digital video: the informedia project," *IEEE Computer*, vol. 29, no. 5, pp. 46–52, 1996.

[81] M. G. e. a. Christel, "Multimedia abstractions for a digital video library," in *ACM Digital Libraries 97* (R. B. Allen and e. Rasmussen, E, eds.), (New York, NY), pp. 21–29, 1997.

[82] C. e. a. Nastar, "Surfimage: a flexible content-based image retrieval system," in *ACM Multimedia 98*, (Bristol, UK), 1998.

[83] W. Y. Ma and B. Manjunath, "NeTra: A toolbox for navigating large image databases," in *Proc. Int. Conf. Image Processing (ICIP-97), vol. III*, (Santa Barbara, CA), pp. 568–571, Oct. 1997.

[84] A. R. L. Zhu, A. Zhang and R. Srihari, "Keyblock: An approach for content-based image retrieval," in *Proceedings of ACM Multimedia*, pp. 157–166, 2000.

[85] W. Wang, Y. Song, and A. Zhang, "Semantics retrieval by content and context of image regions," in *In Proc. of the 15th International Conference on Vision Interface*, 2002.

[86] H. G. C. Carson, S. Belongie and J. Malik, "Blobworld: Image segmentation using using expectation-maximization and its application to image querying," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 24, no. 8, pp. 1026–1038, 2002.

[87] K. Barnard, P. Duygulu, D. Forsyth, N. de Freitas, D. Blei, and M. Jordan, "Matching words and pictures," *Journal of Machine Learning Research*, vol. 3, pp. 1107–1135, 2003.

[88] J. Li and J. Z. Wang, "Automatical linguistic indexing of pictures by a statistical modeling approach," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 25, pp. 1075–1088, Sept. 2003.

[89] D. H. W. S. Gao and C. H. Lee, "Automatic image annotation through multi-topic text categorization," in *International Conference on Acoustics, Speech and Signal Processing, (ICASSP 2006)*, vol. 2, (Toulouse, France), pp. 377–380, May 2006.

[90] R. M. S. L. Feng and V. Lavrenko, "Multiple bernoulli relevance models for image and video annotation," in *Conference on Computer Vision and Pattern Recognition, (CVPR 2004)*, vol. 2, pp. 1002–1009, 2004.

[91] D. A.Forsyth and J. Ponce, *Computer Vision - A Modern Approach*. Prentice-Hall, 2002.

[92] P. Kay and C. K. McDaniel, "The linguistic significance of the meaning of basic color terms," *Language*, vol. 54, no. 3, pp. 610–646, 1978.

[93] J. E. Cairo, *The neurophysiological basis of basic color terms*. State University of New York at Binghamton, 1977.

[94] B. Berlin and P. Kay, *Basic Color Terms: Their Universality and Evolution*. Berkeley: University of California, 1969.

[95] J. M. Lammens, *A computational model of color perception and color naming*. PhD thesis, Univ. of Buffalo, June 1994.

[96] K. Kelly and D. Judd, "The ISCC-NBS color names dictionary and the universal color language (the ISCC-NBS method of designating colors and a dictionary of color names)," *NBS Circular 553*, Nov. 1 1955.

[97] A. Mojsilović, "A computational model for color naming and describing color composition of images," *IEEE Trans. Image Processing*, vol. 14, pp. 690–699, May 2005.

[98] T. P. Minka and R. W. Picard, "Interactive learning using a society of models," *Pattern Recognition*, vol. 30, pp. 565–581, Apr. 1997. Special issue on Image Databases.

[99] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 22, no. 8, pp. 1154–1160, 1998.

[100] M. Mirmehdi and M. Petrou, "Segmentation of color textures," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 22, pp. 142–159, Feb. 2000.

[101] T. N. Pappas, "An adaptive clustering algorithm for image segmentation," *IEEE Tranactions Signal Processing*, vol. 40, pp. 901–914, Apr 1992.

[102] E. P. Simoncelli and W. T. Freeman, "The steerable pyramid: A flexible architecture for multi-scale derivative computation," in *Proc. ICIP-95, vol. III*, (Washington, DC), pp. 444–447, Oct. 1995.

[103] J. G. Daugman and D. M. Kammen, "Pure orientation filtering: A scale invariant image-processing tool for perception research and data compression," *Behavior Research Methods, Instruments, and Computers*, vol. 18, no. 6, pp. 559–564, 1986.

[104] J. Chen, *Perceptually-Based Color and Texture Features for Image Segmentation and Retrieval*. Ph.D. thesis, Northwestern University, Evanston, IL, Dec 2003.

[105] M. Swain and D. Ballard, "Color indexing," *Int. Journal of Computer Vision*, vol. 7, no. 1, pp. 11–32, 1991.

[106] W. E. M. F. E. G. D. P. W. Niblack, R. Berber and P. Yanker, "The qbic project: Quering images by content using color, texture, and shape," in *Storage and Retrieval for Image and Video Databases*, vol. Proc. SPIE, Vol. 1908, (San Jose, CA), pp. 173–187, 1993.

[107] V. V. V. B. S. Manjunath, J.-R. Ohm and A. Yamada, "Color and texture descriptors," *IEEE Trans. Circuits Syst. Video Technology*, vol. 11, pp. 703–715, June 2001.

[108] G. Derefeldt and T. Swartling, "Color concept retrieval by free color naming," *Displays*, vol. 16, pp. 69–77, 1995.

[109] S. N. Yendrikhovskij, "Computing color categories," in *Human Vision and Electronic Imaging V* (B. E. Rogowitz and T. N. Pappas, eds.), vol. Proc. SPIE Vol. 3959, (San Jose, CA), Jan. 2000.

[110] A. B. Y. Linde and R. M. Gray, "An algorithm for vector quantizer design," *IEEE Transactions on Communications*, vol. 28, pp. 84–95, Jan 1980.

[111] W. Y. Ma, Y. Deng, and B. S. Manjunath, "Tools for texture/color based search of images," in *Human Vision and Electronic Imaging II* (B. E. Rogowitz and T. N. Pappas, eds.), vol. Proc. SPIE, Vol. 3016, (San Jose, CA), pp. 496–507, Feb. 1997.

[112] A. Mojsilović, J. Kovačević, J. Hu, R. J. Safranek, and S. K. Ganapathy, "Matching and retrieval based on the vocabulary and grammar of color patterns," *IEEE Trans. Image Processing*, vol. 1, pp. 38–54, Jan. 2000.

[113] D. Comaniciu and P. Meer, "Robust analysis of feature spaces: Color image segmentation," in *IEEE Conf. Computer Vision and pattern Recognition*, (San Juan, Puerto Rico), pp. 750–755, June 1997.

[114] Y. Deng, S. Kenney, M. S. Moore, and B. S. Manjunath, "Peer goup filtering and perceptual color image quantization," in *Proc. IEEE Int. Symposium on Circuits and Systems VLSI*, vol. 4, (Orlando, FL), pp. 21–24, June 1999.

[115] P. K. Kaiser and R. M. Boynton, *Human Color Vision*. Washington, DC: Optical Society of America, 1996.

[116] J. J. MaCann, *Color Perception: philosophical, psychological, artistic and computational perspectives, ch. Simultaneous Contrast and Color Constancy: Signatures of Human Image Processing*. Oxford University Press, 2000.

[117] M. Haindl, "Texture synthesis," *CWI Quarterly*, pp. 305–331, 1991.

[118] M. Tuceryan and A. K. Jain, "Texture analysis," in *Handbook Pattern Recognition and Computer Vision* (C. H. Chen, L. F. Pau, and P. S. P. Wang, eds.), ch. 2, pp. 235–276, Singapore: World Scientific, 1993.

[119] K. S. R. M. Haralick and I. Dinstein, "Texture features for image classification," *IEEE Trans. Syst, Man and Cybern.*, vol. 3, no. 6, pp. 610–621, 1973.

[120] M. Tuceryan and A. K. Jain, "Texture segmentation using voronoi polygons," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 12, no. 2, pp. 211–216, 1990.

[121] H. Voorhees and T. Poggio, "Detecting textons and texture boundaries in natural images," in *First International Conf. on Computer Vision*, (London, UK), pp. 250–258, 1987.

[122] D. Blostein and N. Ahuja, "Shape from texture: Integrating texture-element extraction and surface estimation," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 11, pp. 1233–1250, Dec 1989.

[123] G. C. Cross and A. K. Jain, "Markov random field texture models," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 5, pp. 25–39, 1983.

[124] R. Chellappa and S. Chatterjee, "Classification of textures using gaussian markov random fields," *IEEE Trans. on Acoustic, Speech, and Signal Processing,*, vol. 33, pp. 959–963, 1985.

[125] F. Cohen and D. Cooper, "Simple parallel hierarchical and relaxation algorithms for segmenting noncausal markovian random fields," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 9, pp. 195–219, 1987.

[126] B. B. Mandelbrot, *The Fractal Geometry of Nature*. Freeman, 1983.

[127] A. Pentland, "Fractal-based description of natural scenes," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 9, pp. 661–674, 1984.

[128] J. Malik and P. Perona, "Preattentive texture discrimination with early vision mechanisms," *Journal of the Optical Society of America A*, vol. 7, pp. 923–932, 1990.

[129] M. Unser and M. Eden, "Nonlinear operators for improving texture segmentation based on features extracted by spatial filtering," *IEEE Trans. Syst., Man, Cybern.*, vol. 20, pp. 804–815, 1990.

[130] J. M. Coggins and A. K. Jain, "A spatial filtering approach to texture analysis," *Pattern Recognition Letters*, vol. 3, pp. 195–203, 1985.

[131] M. Unser, "Texture classification and segmentation using wavelet frames," *IEEE Trans. Image Processing*, vol. 4, pp. 1549–1560, Nov. 1995.

[132] A. K. Jain and F. Farrokhnia, "Unsupervised texture segmentation using gabor filters," *Pattern Recognition*, vol. 24, pp. 1167–1186, 1991.

[133] G. V. de Wouwer, P. Scheunders, and D. Van Dyck, "Statistical texture characterization from discrete wavelet representations," *IEEE Trans. Image Processing*, vol. 8, pp. 592–598, Apr. 1999.

[134] N. Graham, "Non-linearities in texture segregation," in *CBIA Foundation Symposium* (G. R. Bock and e. J. A. Goode, eds.), vol. 184, (New York, NY), pp. 309–329, 1994.

[135] N. Graham and A. Sutter, "Spatial summation in simple(fourier) and complex(non-fourier) texture channels," *Vision Research*, vol. 38, pp. 231–257, 1998.

[136] N. Graham and A. Sutter, "Normalization: contrast-gain control in simple(fourier) and complex(non-fourier) pathways of patter vision," *Vision Research*, vol. 40, pp. 2737–2761, 2000.

[137] A. B. Watson, "The cortex transform: Rapid computation of simulated neural images," *Computer Vision, Graphics, and Image Processing*, vol. 39, pp. 311–327, 1987.

[138] S. Daly, "The visible differences predictor: an algorithm for the assessment of image fidelity," *Digital Images and Human Vision*, vol. 39, pp. 179–206, 1993.

[139] T. N. Pappas, "An adaptive clustering algorithm for image segmentation," *IEEE Trans. Signal Processing*, vol. SP-40, pp. 901–914, Apr. 1992.

[140] R. M. Boynton, "Eleven colors that are almost never confused," in *Human Vision, Visual Processing, and Digital Display* (B. E. R. and, ed.), vol. Proc. SPIE Vol. 1077, pp. 322–+, Aug. 1989.

[141] B. E. Rogowitz, T. Frese, J. R. Smith, C. A. Bouman, and E. Kalin, "Perceptual image similarity experiments," in *Human Vision and Electronic Imaging III* (B. E. Rogowitz and T. N. Pappas, eds.), vol. Proc. SPIE, Vol. 3299, (San Jose, CA), pp. 576–590, Jan. 1998.

[142] A. Mojsilovic and B. Rogowitz, "Capturing image semantics with low-level descriptors," in *Proc. Int. Conf. Image Processing (ICIP-01)*, (Thessaloniki, Greece), pp. 18–21, Oct. 2001.

[143] C. Y. Lin, B. L. Tseng, and J. R. Smith, "Video collaborative annotation forum: Establishing ground-truth labels on large multimedia datasets." NIST TRECVID, 2003.

[144] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal Royal Statistical Society*, vol. 39, no. 1, pp. 1–21, 1977.

[145] V. N. Vapnik, *Statistical Learning Theory*. Wiley, 1998.

[146] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*. Wiley-Interscience, 2nd ed., Oct. 2000.

[147] C. B. Moler and G. W. Stewart, "An algorithm for generalized matrix eigenvalue problems," *Society for Industrial and Applied Mathematics (SIAM) Journal of Numerical Analysis*, vol. 10, pp. 241–256, 1973.

[148] G. Golub and C. V. Loan, *Matrix Computations*. The Johns Hopkins University Press, 1996.

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

[150] J. Jeon, V. Lavrenko, and R. Manmatha, "Automatic image annotation and retrieval using cross-media relevance models," in *SIGIR*, pp. 119–126, ACM, 2003.

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

[152] W. Wang, Y. Song, and A. Zhang, "Semantics retrieval by content and context of image regions," in *Proc. 15th Int. Conf. on Vision Interface*, (Calgary, Canada), pp. 17–24, May 2002.

[153] J. Pan, H. J. Yang, P. Duygulu, and C. Faloutsos, "Automatic image captioning," in *ICME*, June 2004.

[154] J. Li and J. Z. Wang, "Automatic linguistic indexing of pictures by a statistical modeling approach," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 25, pp. 1075–1088, Sept. 2003.

[155] J. Z. Wang, J. Li, and G. Wiederhold, "SIMPLIcity: Semantics-sensitive Integrated Matching for Picture LIbraries," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 23, pp. 947–963, Sept. 2001.

[156] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *ICCV*, (Vancouver, Canada), July 2001.