INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

Bell & Howell Information and Learning
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
800-521-0600

AN IMPROVED APPEARANCE-BASED APPROACH TO IMAGE RETRIEVAL AND CLASSIFICATION

Fadi Beyrouti

Department of Electrical Engineering
McGill University

February 1999

A Thesis submitted to the Faculty of Graduate Studies and Research
in partial fulfilment of the requirements for the degree of
Master of Engineering

Abstract

This thesis describes the design and implementation of a content-based image retrieval system that improves on classical appearance-based methods of retrieval. The motivation is to build a system capable of retrieving complex scenes in unconstrained environments. In this work, we use principal components analysis as a parameterization of the images. The analysis, however, is not performed on the level of the intensity values of image pixels as in traditional approaches, but on the level of two intermediate representations. We use the magnitude of the Fourier Transform as a relevant and stable comparison feature to retrieve complex scenes, and zero crossings across various scales for well-framed images. We show through experimental results that the inclusion of suitable intermediate representations in the system renders more reliable responses, and allows relaxing the constraints about the nature of the images used.

This work also extends beyond normal content-based image retrieval systems. Given an unknown image query, our system does not merely return the closest images to the query, but further enhances the response by classifying the image as belonging to one of several classes comprising the database. Two methods of classification are used: the k nearest neighbors method, and a Bayesian classification method. In the latter case, we re-map the basis calculated by principal components analysis into another basis whose vectors are optimal for class discrimination. These vectors are called the most discriminating features. Experimental results are presented showing how the resulting system can be used to return a confidence measure in the decisions it takes, allowing for subsequent refining of the search.

Résumé

Cette thèse décrit l'élaboration et la réalisation d'un système de récupération d'images par contenu qui améliore les performances des systèmes traditionnels. Le but est de construire un système capable de rechercher des images complexes dans des environnements généraux. Le système utilise la méthode d'analyse des composantes principales pour paramétriser les images. Contrairement aux méthodes traditionnelles qui appliquent l'analyse sur les intensités absolues des pixels, le notre le fait sur des représentations intermédiaires des images. A ce sujet, deux approches sont envisagées: la magnitude de la transformée de Fourier pour les images complexes, et une méthode de détection des contours d'objets sur plusieurs échelles pour des images contenant des objets bien encadrés. Avec plusieurs expériences, nous démontrons que l'inclusion de représentations intermédiaires adéquates dans un système donne des résultats plus fiables, et permet un choix plus libre d'images à inclure dans un système de récupération.

L'amélioration du système de récupération ne s'arrête pas ici, mais s'étend au delà du retrait des images les plus proches d'une requête. La réponse du système inclu davantage d'information sur l'appartenance de l'image de requête, lui attribuant une classe parmi celles prédéterminées, formant la base complète. Deux méthodes de classification sont utilisées: la méthode des k plus proches voisins, et celle de la classification Bayesienne. Dans ce dernier cas, les vecteurs formant la base de l'analyse des composantes principales sont transformés pour obtenir une autre base de vecteurs, idéale pour séparer les différentes classes. Ces vecteurs sont appelés les caractéristiques les plus discriminatoires. Des résultats expérimentaux démontrent comment l'algorithme peut être utilisé pour générer une mesure de confiance dans les décisions prises, qui permettra un raffinement ultérieur de la requête.

Acknowledgements

First and foremost, I would like to thank my supervisor, Professor Frank Ferrie, to whom I will always be indebted for his support on a financial, moral, and technical level. His encouragement throughout this degree has helped me enormously to overcome the long and tough aspects of this work.

Next I would like to express my gratitude to two of my friends and colleagues at the Artificial Perception Laboratory, Tal Arbel and Gilbert Soucy. Their help and cooperation has been a major factor in the way this work has turned up to be. Thank you also for proof reading this thesis, and for your feedbacks and insights in the field of computer vision.

I would also like to thank all of members of the Artificial Perception Laboratory in the Centre for Intelligent Machines. Specifically, I thank, Franco Callari, Peter Whaite, Pierre Tremblay, and Stephen Benoit. I learned so many things from all these people, and my vision of the intellectual way of thinking has been influenced by all of them.

Special thanks to my friends, "The Ducks", whose presence all these years have made my long stay at McGill much more pleasant.

Finally, I am indebted beyond words to my family, and to Alia. I consider myself very lucky to be a part of your life.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

Introduction

The goal of a vision system is to be able to infer the state of the world that it sees. Of course, inferring the world, depending on the context, could have various meanings; such as finding the best parameters of a model that describes a set of data (Whaite and Ferrie 1993), or locating the contours of some objects in a real scene, or even simpler tasks like finding the histogram of intensities in an image. Ultimately, however, given the data (measurements) it is provided with, a vision system would like to recognize the objects that constitute its visual world. The data that influence a system's behavior can come in a multitude of forms, ranging from three-dimensional raw data of the objects, to two-dimensional silhouettes, to grey-scale intensity images of real objects (like cars, faces, trees, etc.) embedded in different kinds of backgrounds. If we closely analyze the human visual system, we notice that, from raw data consisting of electromagnetic waves, a human is capable of perceiving the world and its constituents with striking immediacy. Immediate also is the recognition of the great number of objects of different shapes, sizes, colors, and other properties that exist in the world. For these and various other reasons, object recognition has been an important research field among vision scientists, from the early days of computer vision until now; and has always found a multitude of industrial and medical applications.

The rapid development of multimedia information technologies has opened the door to a research area within computer vision called *content-based image retrieval*. By content-based image retrieval we mean being able to analyze images by their inner properties, and consequently infer the existence of similar contents in other images (Niblack, Barber, Equitz, Flickner, Glasman, Petkovic and Yanker 1993, Kelly and Cannon 1994). The word *content*

refers to pictorial attributes that can be extracted by image analysis methods. Content-based image retrieval is very tightly related to classical object recognition. There are two main differences however. The first difference is that in classic object recognition the main goal is to be able to understand clearly what the physical components of the visual world are, while the emphasis in image retrieval is on indexing, which is the assignment to each image of a synthetic descriptor facilitating its retrieval, whether related to the physical objects present in the image or not. The second difference lies in the fact that usually content-based image retrieval systems include very large databases of images, and hence massive amounts of data. Being able to retrieve images from large databases in a reasonable time therefore becomes a big challenge for researchers in this field. With the emergence of many applications requiring near real-time responses, it is becoming crucial to perform fast image retrieval. Methods have to be thought of to overcome the computational expense associated with large data sets. The borderline between indexing and recognition in images remains, however, very subtle.

The operations performed by most content-based image retrieval systems are summarized in Figure 1.1. The input to the system specifies a query through a user interface. The query is represented according to some description. On the other side, descriptions of all the images in the database are found through the indexing of relevant features. The query description is then compared to the database ones to find an answer space which is converted by the user interface to an output consisting of the retrieved images.

An image retrieval system consists of several phases of operation:

(i) The query phase: express the information retrieval problem in a query "language" (e.g., query by example, query by sketch, etc.).

(ii) The indexing phase: extract from the image the properties that are relevant to the image retrieval process.

(iii) The organizational phase: organize the image indexes such that the speed of the retrieval of images is optimized.

(iv) The retrieval phase: extract from the database, the images that best match the query.

(v) The feedback phase: refine the search when needed.

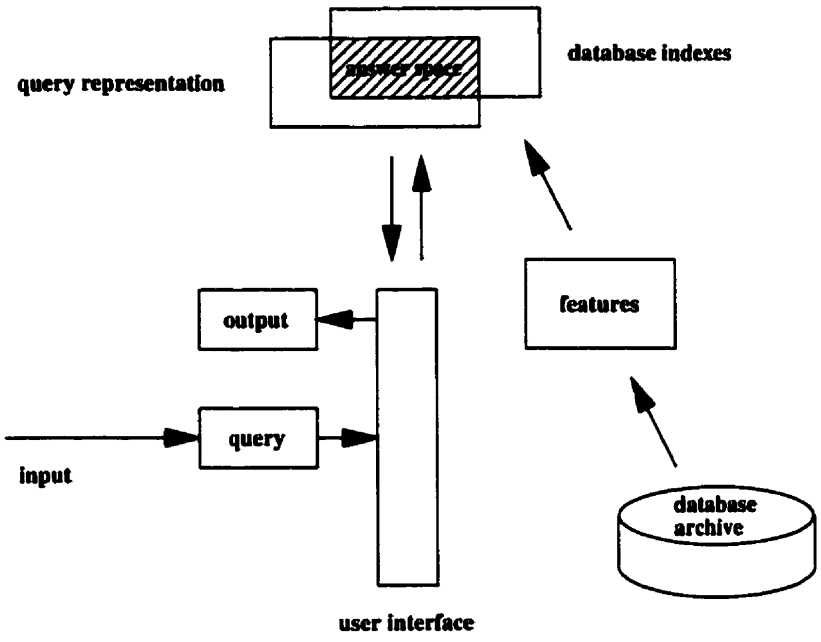


FIGURE 1.1. Information retrieval schema

In later chapters we will discuss in detail the requirements of the above phases. In this thesis, we concentrate on the indexing phase and the retrieval phase. We build a system that performs two separate but related tasks.

Given a query represented by an unknown example image,

(i) *We "classify" this image as belonging to one of several classes of images previously labeled.*

(ii) *We extract from a large database of images, those which are closest to our query image.*

The process consists of an off-line part and and on-line part and works as follows: off-line, we manually classify the images present in the database as belonging to one of several classes of images, where the classes convey semantic information about the contents of the images. This information is compatible with the human perception of structure (e.g., beach scenes, human faces, playing cards, fire works, etc.). Next, we seek a parameterization of our images such that each image is represented by a small number of parameters. To perform this step successfully, we need a training set consisting of a small set of sample images which we choose from the database. Of course, the training set has to be representative of the

whole database, something that is not a trivial matter. Once we have found the parameters for all the images in the database, we are done with the off-line part. On-line, we process our query image which does not necessarily belong to the original database. We calculate its parameters (coefficients) with respect to the basis model found in the off-line stage, then we classify the image by using both the traditional *k nearest neighbors* (knn) method, and a probabilistic method. Having calculated the k nearest neighbors of the image, our system returns these images as the images that best match our query. With the exception of the manual classification of the database images, our system is fully automatic and doesn't require human interference. Furthermore, it is designed to work on real images that include complex data.

1. Problem Definition

> *The primary goal of this research is to design a content-based image retrieval system that improves on the performance of classical appearance methods. The improvements in question are concerned with (i) Generality: the system has to be able to perform well in generalized environments like natural scenes. (ii) Robustness: the system has to be less sensitive than classical content-based systems to factors like the absolute illumination level, camera angle, translation, and rotation.*

Appearance-based indexing methods have several advantages, the two most important ones being their fast response time, and their ability to deal with real images. However, the systems built so far suffer from a number of deficiencies, such as their sensitivity to external factors like the ones mentioned above; and therefore their inability to work well unless the environment is well constrained. The motivation for this work comes from the need to address these problems. Having this focus in mind, we design a content-based image retrieval system that returns the closest images in the database to a given example image. We show that this system returns more accurate results than classical methods, while still being fast and automatic (not requiring human interference). Furthermore, the system we build stretches beyond normal image retrieval. In addition to returning the closest matches to the query, the system also returns a "classification" of the query. More specifically, if the images in the database are separated into several classes, then the system indicates the

score of each class with respect to the query. This opens the door for refining the search for best matches by "throwing away" the unlikely classes and comparing only a subset of the whole database.

2. Overview of the Approach

The application of this work is twofold: similarity retrieval by content and image classification. Images are compressed by a parameterization using *Principal Components Analysis*, where the representation of the images is reduced to just a small number of coefficients in an optimal fashion. This parameterization allows the system to compare images in a fast and accurate manner, two basic requirements of practical systems. Principal components analysis is a well-known method that is widely used in applications such as content-based face recognition (Sirovich and Kirby 1987, Kirby and Sirovich 1990, Turk and Pentland 1991), face tracking, and recognition of other objects. The above approaches mainly consist of finding a set of features that characterize the variations between images. As a result, these features are considered the "best coordinate system for image compression". Comparison between images is then accomplished in this lower dimensional space, where principal components analysis is applied on the level of the absolute intensities of the images. The approach we use here is somewhat similar. However, the important distinction of our methodology is that we replace analyzing the absolute intensities of the images (which do not convey semantic information about the objects in the scenes) by analyzing stable intermediate representations of the images. Following this concept, we compare the images based on their salient information, like the shape of the objects and their frequencies. It should be noted that salient information is very context dependent, while natural images are composed of diverse and complex objects. Therefore, it becomes very important to define the constraints the environment provides us with in a proper manner. Most of the content-based image retrieval systems so far have over-constrained the environments that they work in. We try to avoid this as much as possible and propose a system which works under different lighting conditions, camera angles, and is invariant to rotation and translation. The content-based image retrieval framework we use in this work is based on the following concepts.

2.1. The Indexing Phase.

(i) For scene analysis purposes where our images are considered to be complex, we calculate the *Fourier Transform* of all images, and then base our comparison on the magnitude of the Fourier transform rather than on the images themselves. The Fourier transform is particularly suitable for scene analysis because it captures the frequencies of a particular signal. Therefore, two similar scenes are expected to have very similar frequency characteristics, regardless of different camera angles and the amount of lighting.

(ii) For well framed images, where the environment is much more constrained than in the scene analysis case, we compare images based on their scale space salient features. More specifically, we find the zero-crossings of the images under different scales. By doing so, we remove the effect of the variations in the lighting conditions. Furthermore, by extracting such features, we compare objects on the basis of their shape characteristics rather than on intensity values of the pixels, adding semantic meaning to the image comparison process. Although intermediate representations have been used previously in image matching (Huttenlocher, Lilien and Olson 1996), content-based image retrieval of real images has widely relied on comparing images based on their absolute pixel intensities.

(iii) Principal Components Analysis (PCA) is used to optimally parameterize the interesting attributes in images.

2.2. The Retrieval Phase.

(i) The closest images to the query image are calculated in the lower-dimensional space, and returned as the best matches to the query.

(ii) For image classification, we use two different methods to assign our query image to a class among several. The first method is the classical *k nearest neighbors*, which selects among all the classes comprising the database, the one that has the highest score in the closest neighbors. This decision rule is attractive because no prior assumptions about the underlying distribution of the data are assumed. This type of decision rule is called non-parametric. The other method we use is a parametric decision rule: assuming that the prior distributions of the classes comprising the database are multivariate normal distributions, and that the parameters (mean and

covariance) of the distributions are estimated from the data (images), we calculate the posterior probabilities that the given item belongs to each of the database classes. We apply Bayes' rule on the distribution of the data to calculate this probability.

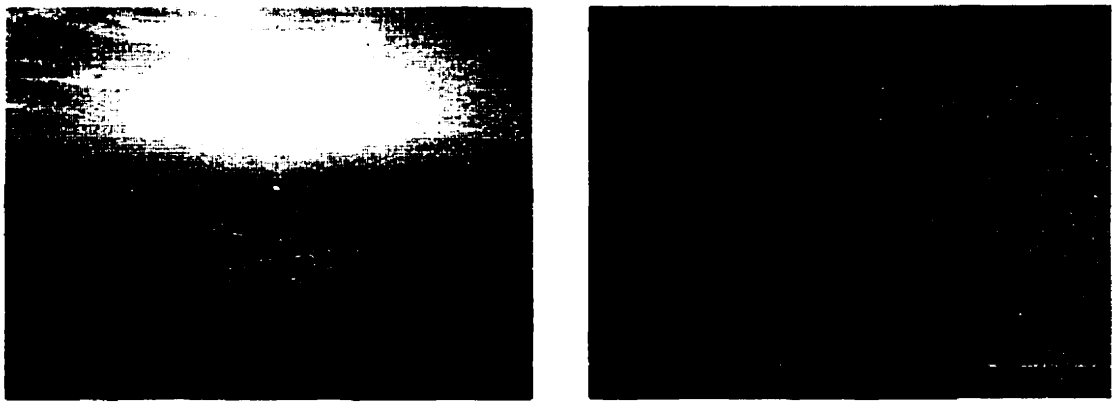
(iii) If we want our system to be able to generalize well, that is, to classify correctly those images that do not belong to the training set, then we must make sure that the classes which constitute the database are as far apart as possible. That way we minimize the confusion of a query belonging to a certain class. Therefore, we re-map the basis calculated by principal components analysis into another basis whose vectors are optimal for the purposes of discrimination between classes. These vectors are called the *Most Discriminating Features* (Wilks 1963), and are introduced by Swets and Weng (1996) as well as by Belhumeur, Hespanha and Kriegman (1996) as an application to image retrieval. In this work, the vectors are used as the new basis vectors for probabilistic scene classification.

In our context, the input data we are provided with are grey scale images, where the images are represented by pixels, and each pixel has a grey scale value ranging from 0 for black to 255 for white. The images we consider could be any type, but we treat two types of images in a different fashion. Those which are comprised of an object of interest with a very simple background, and those which comprise various objects hard to separate from each other. In this thesis, we call the first type of images *well framed*, and we call the second type *scenes*. Figure 1.2 shows an example of both a scene and a well framed image.

3. Organization of the Thesis

Many techniques have been proposed to efficiently solve the image retrieval problem in large databases. In Chapter 2, we present an overview of the various strategies used to tackle this problem and that have been introduced over the past decade. We will focus our attention on content-based schemes, which use information such as color statistics, patterns (textures), shapes, spatial relations of objects, and combinations of these.

The purpose of Chapter 3 is to review the theory behind principal components analysis, and look at it from two different points of view, the model fitting view, and the statistical view. We investigate whether principal components analysis is optimal for image classification. We show that while the basis it calculates is optimal in its expressive power, it



(a) Scene (b) Well Framed

FIGURE 1.2. A scene and a well framed image

might not be optimal for discriminating between different classes. Therefore, we seek a new basis that satisfies this requirement based on multidimensional discriminant analysis. We describe the theoretical foundations behind multidimensional discriminant analysis. This analysis results in the calculation of the Most Discriminating Features mentioned in point (iii) above. Furthermore, we discuss the reasons why for probabilistic classification purposes, this approach is preferred to principal components analysis.

Principal components analysis has been used successfully before in applications where the domains of images are well constrained. The main reason for introducing constraints is that such systems compare images based on their absolute intensities, with the resulting limitations: i) the sensitivity to the camera position, i.e., small disturbances in camera position might lead to large changes in the output of principal components analysis, ii) the sensitivity to small changes in the lighting conditions, which makes the comparison of two similar scenes with different lighting conditions a very difficult task, and, iii) the sensitivity to translation and rotation, something which could be problematic for applications involving unconstrained scenes. We discuss all of these issues in detail in Chapter 4 and conceive ways to deal with them.

The two most important phases in an image retrieval system are the indexing phase and the retrieval phase. While chapters 3 and 4 discuss issues concerning the indexing of images, Chapter 5 discusses the retrieval of these images. Specifically, we introduce the

two classification methods we use in the system, and discuss their differences as well as their relative advantages and disadvantages. Furthermore, we investigate how parametric uncertainty is used for the purpose of image classification, and discuss the conditions under which Bayes rule is used to return belief distributions. These distributions are applied conveniently to quantify the confidence of the system in the decisions it is taking. This results in a method that estimates, for each predetermined class of images, the conditional probability of the unknown query belonging to the class. In this case, each class is treated as a statistical distribution with a mean and a covariance.

In Chapter 6, we present experimental results that compare the performance of our proposed approach to classical principal components analysis methods. To investigate our system in detail, we analyze the various aspects of our method, and decide which strategy is more adequate under the given conditions of database specifications. In this thesis we consider two cases, complex scenes and well framed images. We illustrate that the two cases require different analysis methods. Specifically, we show that the use of the magnitude of the Fourier Transform as the basis of comparison for the analysis of complex scenes is advantageous with respect to classical appearance-based methods or with scale space feature analysis. On the other hand, if the database consists of well framed objects (faces for example), then we get better results when we index the database images and the query according to their features in scale space. The other set of experimental results concerns the relative performances of the two retrieval methods we use in this thesis. We analyze the differences between the two decision rules of classification, knn and Bayesian classification, and discuss the advantages and disadvantages of each. Finally, we indicate how our system paves the way for a complete content-based image retrieval system that is fast and stable.

We conclude in Chapter 7 with some general observations on our current work and points for future research.

4. Contributions

The contributions of this thesis consist of the following:

(i) The proposed approach to content-based image retrieval system extends the principal components analysis framework to more general classes of images. This is done in

order to handle situations where unreasonable constraints to the environment should be avoided.

(ii) To be able to get good results with complex images, the system developed applies principal components analysis on intermediate representations rather than on the images themselves. We use the Fourier transform as a relevant and stable comparison feature to retrieve general scenes, and we show that the inclusion of this representation improves the retrieval results considerably. If it can be assumed that the environment (image database) can be constrained so that all objects in the images are well framed, then the system uses zero crossings across different scales to represent the images.

(iii) The system extends beyond the normal content-based image retrieval system by including a quantification of the results. Therefore, the retrieval of "the closest images to the example image" is further enhanced by probabilistic Bayesian analysis whose result is a basis by which an external agent can assess the quality of the retrieval, decide whether it is good or not for the specific application, and therefore allow the search to be refined if possible.

(iv) The system paves the way for a more elaborate content-based image retrieval system. In fact, this work is part of general digital library project where the preliminary tests have demonstrated some promising results.

CHAPTER 2

Review of the Literature

1. Introduction

Traditionally, image retrieval from databases was based on textual annotations to the images. These textual annotations included fields like the date the picture has been taken, the photographer, and some descriptions of the contents of the images (e.g., church, courthouse, forest, etc.). However, it is very hard to address the contents of an image using textual query languages, the reason being the great amount of information present in real images, the richness of its structure, and the fact that the information content is based on two-dimensional entities rather than well defined patterns. Therefore content-based image retrieval methods became more popular in the last decade because they address images based on their peculiar informative structure. The most significant part of the retrieval process is the indexing part, which can be summarized as the assignment of "key words" from a description "language" to document entities to facilitate their retrieval. This is where most of the research has been concentrated. However, it is very hard to extract indices that are efficient and reliable for general classes of images. So far, present algorithms are only successful in dealing with limited classes where there is a small number of non-overlapping objects on a simple background. Furthermore, many systems are very sensitive to lighting conditions, as well as different views of the same object.

2. Indexing by Color and Gray Level Intensity

Color occupies an important role in image indexing because of the simplicity of automatically computing color features, and the fast processing of color queries. Furthermore,

color is a powerful feature in finding similar images. Some algorithms use color information as relevant features for image retrieval.

In the QBIC(*Query By Image Content*) system (Faloutsos 1994, Niblack, Barber, Equitz, Flickner, Glasman, Petkovic and Yanker 1993), color information is represented through color space conversion, quantization, and clustering. Specifically, the RGB space is quantized into K levels for each axis, which results in a K^3 number of cells. Next, the cells are transformed into *Super Cells* using the Mathematical Transform to Munsell (MTM), which results in a $K \times 1$ vector for each image. When querying for a new image, a difference histogram Z is calculated between the query image and all the images in the database, and a similarity measure is given by $\|Z\| = Z^T A Z$, where A is a symmetrical matrix with $A(i,j)$ representing how much colors i and j are similar.

In Ardizzone and LaCascia (1997) the RGB color space is more conveniently converted to a quantized HSV color space, then the 3-dimensional quantized HSV histogram is computed. Furthermore, to facilitate the retrieval, the histogram is reduced to a single variable histogram that is representative of the 3-dimensions. Color histograms of different images are compared using the Euclidean distance.

Color information is even more significant for video indexing, because it can help to segment objects in the scene. In the last few years several color based techniques have been proposed for video annotation (Swain and Ballard 1995, Zhang, Low, Smoliar and Wu 1995, Smith and Chang 1996, Ardizzone, LaCascia and Molinelli 1996). However, color by itself is not sufficient for image indexing because it conveys no information about the shape of the objects. For example two objects having the same color characteristics could be labeled similar even though they are very different in nature (e.g., red apples, and red cars). Therefore, more features should be included in a system to be able to perform effective queries and to eliminate false positive retrieval.

Other research uses just gray level information. Chang and Yang (1983) find the minimum number of pixel gray level changes to convert the image into a constant gray level image (or more generally into a k-gray levels image). The absolute minimum value is when the image is originally a constant gray level image, while the maximum is found when the image histogram is completely uniform, that is when the image is most informative. This very simple scalar measure is used as an index to the images in the database. It can be

further generalized by considering the objects in the image (which are extracted manually) where the analysis handles each object separately.

3. Indexing by Texture

Texture features have been proposed for content-based retrieval. A well-known content-based algorithm (Kelly and Cannon 1994), CANDID (Comparison Algorithm for Navigating Digital Image Database), includes texture features obtained by convolving the image with Laws' convolution kernels (Laws 1980). Probability density functions in the texture features are used as an index to the image, and image matching is performed by finding a suitable distance measure between probability density functions of respective images.

The most well known work concerning texture analysis is that done by Picard and Liu (1994). Each image is considered to be the product of three different phenomena: a harmonic one, a directional one, and a non-deterministic one corresponding to perceptual complexity. The three different fields are mutually orthogonal, that is they are (in theory) completely independent from one another. Since the harmonic field represents perceptual repetitiveness then it must be the dominant field in textured images. The algorithm developed exploits this observation, by comparing the periodicities of the query image with other images in the database. If the image is periodic, its harmonic peaks are compared with those of the stored images and the closest ones are retrieved. If the image is not periodic, the two other components (directionality and complexity) are evaluated using more complex types of queries (Picard and Liu 1994, Liu and Picard 1996).

Other algorithms use texture information to index images, most of the time including other types of features. Research on texture indexing can be found in Tamura, Mori and Yamawaki (1978), Francos, Meiri and Porat (1993), Rao and Lohse (1993), Manjunath and Ma (1996), LaCascia and Ardizzone (1996). There are a lot of disadvantages to texture based features however. First, it should be noted that these features work only in very constrained environments for specific applications. Second, most of the above algorithms assume that the image contains just one type of texture. Typical images usually contain different kinds of textures. Finally, texture features are useless in non-segmented images which are expected to have other non-textured objects. Dubuc and Zucker (1995) propose a formal complexity theory appropriate for separating different kinds of textures and curves

based on the calculation of the tangent and normal complexity at every point in a tangent map. Including this kind of framework in texture analysis could help to automate the process of texture segmentation to a large extent.

4. Indexing by Shape

Using features related to objects' shape to index images has received a lot of attention in computer vision. Algorithms developed are very diverse and often far apart in their approaches. It should be noted that the automatic definition of a shape by a computer may not at all correspond to what humans perceive as shape. Therefore the notion of shape similarity may be different as well. Two shapes can appear completely different to the computer although they represent the same object according to the human user, and vice versa. It is very hard to represent the human perception of similarity in a mathematical form. However, in order for an algorithm to be successful, human and computer judgments of similarity must be generally correlated.

Photobook is a set of interactive tools developed by Pentland, Picard and Sclaroff (1996) to index images by content. It consists of three separate parts: an appearance part (to be discussed later), a shape part, and a texture part which is the one mentioned in Section 3 (Picard and Liu 1994). The shape part treats the problem of shape similarity between two objects as an equilibrium equation. First the local curvature of the object is calculated, then feature points on the objects are chosen based on the highest curvatures. The feature points are then thought as attached to an elastic body by springs. The elastic body will deform subject to the force exerted by the springs; the equilibrium equation found is characteristic of the object's shape. The equations for two different images are compared for specific feature points. As a matter of fact for each object two matrices are built: K for stiffness and M for mass. Then the "mode shape vectors" φ_i are calculated so that they satisfy the equation,

$$K\varphi_i = \omega_i^2 M\varphi_i, \tag{2.1}$$

Each mode shape vector describes how the feature points on the objects are displaced. The similarity measure between two objects is found by computing the deformation energy

needed by the object to align with the query, the less the energy required, the greater the similarity. The problem with this approach is that it is only successful when dealing with objects with simple shapes on a clearly defined background like mechanical tools or silhouettes. If the images get complicated then the number of feature points needed grows enormously and the algorithm becomes computationally very expensive.

Another approach is used in Roadhouse and Kimia (1997), which is inspired from Blum's work (Blum 1973) and which represents the shape of objects as a growth process. The growth is specified by a history decomposed into a spatial element which is the skeleton of the object, and a temporal element or the dynamics of the evolution. The advantage of this description is that a qualitative approximation in the history domain produces a rich description of shape with only a limited number of parameters. The algorithm works well for two-dimensional silhouettes but doesn't address the problem of extracting the objects from real scenes.

Image contours have been used as well in shape-based retrieval approaches. Kliot and Rivlin (1997) index images based on geometric invariant features. Furthermore, the book by Mundy and Zisserman (1993) discusses the issue of invariance in computer vision in detail. Various object recognition systems that use geometric invariants are included in the book, like the Ponce and Kriegman system that locates objects from two-dimensional image contours. The advantage of these approaches is that they are able to retrieve images in situations in which part of the shape is missing or if the object is viewed from a different viewpoint. Again it can deal successfully with simple contours but not with complex scenes.

Several other works used shape-based indexing for image databases. Grosky et al. matched a chain code representation of objects for retrieval of images from a database of industrial parts (Grosky and Lu 1986, Mehrotra, Kung and Grosky 1990, Grosky and Mehrotra 1990, Grosky and Jiang 1994). Chang et al. focus on the automatic extraction of low-level visual features such as texture, color, and shape (Smith and Chang 1995, Chang and Smith 1995, Chang 1995). Fleck, Forsyth and Bregler (1996) combine color and texture properties as well as geometric constraints of the human body to identify naked people in pictures. In Kato, Turita, Otsu and Hirata (1992), and Hirata and Kato (1992), the algorithm extracts an abstracted edge map as the basic representation of shape and a measure of similarity to user drawn sketches for indexing. Del Bimbo et al. deformed

contours of a query shape to adhere to object boundaries and used the degree of deformation as the measure of similarity (Del-Bimbo and Pala 1997). Finally, Pernus et al. calculate global features from curvature information at different scales, like bending energy and the sum of absolute curvatures. A multiresolution vector is stored for each feature, and a binary tree classifier is used to speed up the search for the best match (Pernus, Leonardis and Kovacic 1994).

The major difficulty with shape-based features for indexing is the problem of locating the salient objects in the image. Automatically acquiring shape models from images with various objects that occlude each other is an open problem in computer vision. For now, a user has to select an appropriate object manually, or deal with constrained images with single objects of interest. Under such circumstances shape-based indexing is expected to perform pretty well. The problem is that various applications require fully automatic retrieval of very general pictures, and shape-based indexing can not tackle this problem as yet.

5. Indexing by Spatial Relations

Chang, Shi and Yan (1987) introduced an algorithm to describe the spatial contents of an image, based on the relative positions of the different objects composing the scene. The algorithm deals only with symbolic pictures, therefore assuming that the objects of the scene have been already recognized and labeled in an iconic fashion. This method has been exploited in image retrieval systems, where the problem of pictorial information retrieval becomes a problem of 2D subsequence matching. The efficiency of this method depends to a large extent on the two-dimensional string representations and on the string matching algorithm, therefore allowing variations of the original algorithm (Lee and Hsu 1991). The general idea reduces to projecting the 2D iconic image along the x and y axes once the objects have been segmented; then registering the relative positions of these objects on both axes. At this point we obtain two strings, one for the objects' spatial relations seen on the x axis and another one seen on the y axis. The query image is then represented in the same fashion, and from two one-dimensional substring matching processes, the algorithm verifies whether the query image is a subimage of any of the images in the database by transforming the problem into a graph matching problem (Costagliola, Tucci and Chang 1992) to simplify

the one-dimensional to two-dimensional substring matching. Other algorithms use centers of mass of minimum enclosing rectangles instead of the full objects (Jungert 1993) to simplify the representation process. The conversion to a graph representation is computationally expensive however, and in a working environment this leads to slow responses. Other researchers included additional improvements, like organizing the data in a hierarchical manner to reduce the number of matches in the database, as well as using geometric hashing schemes to store the data efficiently (Wu and Chang 1994). The main problem remains that the spatial relations methods cannot deal with real scenes directly. Different approaches have to be used to tackle this issue.

6. Indexing by Appearance

The limitations of shape-based indexing with real images have inspired a lot of researchers to address the problem from a different point of view. Images are not indexed by extracting the shape of the objects that are contained in them, but instead they are retrieved using a characterization of the visual appearance of objects. An object's visual appearance depends on several factors that are hard to define and especially hard to separate, like the three dimensional-shape of the object, its surface albedo, its texture, the viewpoint from which it is imaged, the lighting conditions, among other things. Appearance-based indexing treats the image (or a subimage) as a whole entity that represents more than just the sum of its parts. This is the main difference with shape-based indexing. Having said this, we believe that the object's shape is the main factor in characterizing its appearance. Therefore, the assumptions made in shape-based indexing are generally still valid when it comes to appearance-based indexing.

The pattern recognition community has used the statistical frameworks to perform useful systematic characterizations for different kinds of patterns, (Watanabe 1965, Fukunaga 1972). Later, Sirovich and Kirby (1987) and again Kirby and Sirovich (1990) used appearance-based methods in the context of face recognition. They represented the appearance of a face using a parametric eigen representation described in Chapter 3. The representation starts by considering the image as a fixed length vector, and then using a test set to calculate the "principal directions" of the data to be able to compress the images considerably to just a few coefficients. Image similarities are performed using an L_2

Norm (Euclidean distance). Turk and Pentland (1991) extended this research to be able to recognize and track faces that are not originally represented in the test set. The assumption made was that a face can be characterized by a small set of 2-D characteristic views. They called the principal directions *eigenfaces* because they describe the most significant variations in the face space, and because they are the eigenvectors of the sample test set of faces. This work has also been included as the appearance module of Photobook (Pentland et al. 1996) The algorithm performed quite well on faces and objects with well separated backgrounds but was not tested on more general classes of images.

Nayar, Murase and Nene (1996) find all possible appearance variations of a small set of objects. These variations define the visual workspace of the objects and are considered *appearance manifolds*, where each object has an appearance manifold parametrized by several variables, namely object pose and illumination. Given an unknown input image, it is first projected into the lower dimensional eigenspace, then the distance from the appearance manifolds indicates the identity of the object. Their work also extends beyond object recognition to include applications for object tracking and illumination planning. Unfortunately, the image indexing system deals only with a small number of objects. The cost of adding more objects and training the system under all poses and illumination variations becomes enormous.

Recently, the need has emerged for more accurate and robust measures of similarity than classical Euclidean distance or normalized correlation (Brunelli and Poggio 1993). Researchers started looking at the matching process from a probabilistic point of view. In Moghaddam, Pentland and Nastar (1996), and Moghaddam, Wahid and Pentland (1998), the eigenface analysis mentioned earlier is enhanced by substituting the L_2 norm as a similarity measure by a probability measure representing the a posteriori probability of a face belonging to some individual. Specifically, two types of classes are defined: intra-personal, Ω_I, representing the variations between pictures of the same person, and extra-personal, Ω_E, representing the variations between different people. The similarity measure is then expressed in terms of the probability,

$$S(I_1, I_2) = P(\Delta \in \Omega_I) = P(\Omega_I \mid \Delta), \tag{2.2}$$

where $P(\Omega_I \mid \Delta)$ is the a posteriori probability given by Bayes rule and Δ is the difference between the two images, that is $\Delta = I_1 - I_2$.

Ravela and Manmatha (1998) present a different description of appearance, based on local properties. The authors represent the local appearance of the intensity surface as responses to a set of Gaussian derivative filters. Queries are designed by users by choosing appropriate regions, matching is performed in parameter space as well as in coordinate space, and the best matches are displayed as an output. Their system has several advantages in that it deals with real images and neither requires segmentation nor training. The big disadvantage is that when dealing with local representations, the algorithm will slow down and ultimately break even if the database is moderately sized. The main difference with our work is that we use global properties of appearance with the application of principal components analysis to local representations.

Huttenlocher, Lilien and Olson (1996) use appearance based recognition to compare binary images. A major advantage of their method is that it uses the Hausdorff fraction as a salient feature to represent the images. This way, the comparison is performed on a subspace of intermediate representations of the images, which makes the approach more robust to occlusions, outliers, and variations in the intensity of the background pixels.

Other research on appearance representations includes Pun and Squire (1996) who use correspondence analysis instead of principal components analysis in order to determine the relevance of each feature to the indexing process. Ohba, Sato and Ikeuchi (1998) address the problem of indexing partially occluded objects by analyzing the RGB color space. Chandrasekaran, Manjunath, Wang, Winkeler and Zhang (1996) developed an algorithm to update the eigenspace representation as new images are included in the database, that way representing all the images in the test set. Nan, Dettmer and Shah (1997) extend appearance based methods to video images and specifically address the problem of lip-reading. Finally, Jacobs, Finkenstein and Salesin (1995) use the Haar wavelet decomposition of the image color space (Stollnitz, DeRose and Salesin 1995) to extract a small number of coefficients of largest magnitude to represent the images. The coefficients are organized in search arrays to optimize the speed of the search. Their algorithm is very efficient and it is easy to add new images to the database.

Appearance representations are probably not enough to be able to index accurately all the possible image situations, nor are they a replacement for shape representations. As a matter of fact, appearance-based indexing which includes properties external to the objects in an image is complementary to shape-based indexing whose properties are intrinsic to the objects in the image. Some way has to be found to exploit the advantages of both in the same algorithm. The focus of this thesis is to deal with this problem by extracting from the images features relevant to the shape of the objects in the images, and then analyzing them in an appearance-based fashion.

CHAPTER 3

The Ultimate Compression: Principal Components Analysis and the Most Discriminating Features

1. Overview of Principal Components Analysis

In this chapter we review the theoretical background of the method that we will use throughout this thesis. The method is called *Principal Components Analysis* (Pearson 1901) and has been used extensively by the pattern recognition community (Watanabe 1965, Fukunaga 1972). Principal components analysis (PCA) is very useful for a large class of problems that have to deal with objects represented by large amounts of data, where it considerably reduces the computational complexity of processing the data. In this thesis we apply principal components analysis to images or to properties of images. The general framework, however, has far broader potential applications. The explanation of the theory below will respect this broader view and is, therefore, not just constrained to images.

Consider an object in the world represented by a very large number of parameters that we call the *dimensions* of the object. Furthermore, consider a database that contains a large number of such objects, where all the objects have the same number of dimensions, say n. We can consider each object in the database as a point (or vector) in $n-$dimensional Euclidean space. Objects that are similar are expected to be close together (i.e., the Euclidean distance between them is small) in this very high dimensional space. Similarly, objects that are very different are expected to be far away from each other. We assume that there are meaningful structural relationships among the objects in the database.

In most cases of data analysis (including ours), the dimensionality n of the objects in the database is too large to be practical. That is, it slows down the process enormously. Therefore the need emerges for a representation with a much lower dimensionality without losing a great deal of information and without distorting the relationships between the original objects. Assuming that a relatively small number of features are sufficient to characterize this high dimensional data, principal components analysis provides a general framework for such requirements.

PROBLEM DEFINITION 1. *Let $\mathbb{R}^n$ be an n-dimensional Euclidean space, and let $\chi = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_p\}$ be a set of points belonging to $\mathbb{R}^n$. Find a subspace of $\mathbb{R}^n$ consisting of m vectors, where $m \ll n$, that will fit the p points belonging to χ in a manner that will minimize the error of the fit in the least squares sense.*

The mean of the set χ is,

$$\mu = \frac{1}{p}\sum_{i=1}^{p}\mathbf{x}_i. \tag{3.1}$$

If we subtract each point $\mathbf{x}_i$ from the mean μ we obtain,

$$\mathbf{y}_i = \mathbf{x}_i - \mu, \qquad i = 1, 2, .., p. \tag{3.2}$$

It turns out that the optimal subspace as defined in Problem Definition 1 is spanned by the eigenvectors of the matrix $\mathbf{C}$ corresponding to the m highest eigenvalues, where $\mathbf{C}$ is defined as follows,

$$\mathbf{C} = \sum_{i=1}^{p}\mathbf{y}_i\mathbf{y}_i^T = \mathbf{A}\mathbf{A}^T, \tag{3.3}$$

and $\mathbf{A} = [y_1, y_2, ..., y_p]$.

Let $\mathbf{u}_1, \mathbf{u}_2, ..., \mathbf{u}_m$ be the m normalized eigenvectors of $\mathbf{C}$ corresponding to the m highest eigenvalues. Then,

$$\lambda \mathbf{C} = \mathbf{C}\mathbf{u}. \tag{3.4}$$

Since $\mathbf{C}$ is real and symmetric, then $\mathbf{u}_1, \mathbf{u}_2, ..., \mathbf{u}_m$ are orthogonal, that is,

$$\mathbf{u}_i^T \mathbf{u}_j = \begin{cases} 0 & \text{if } i \neq j, \\ \|i\|^2 = \|j\|^2 & \text{if } i = j. \end{cases} \tag{3.5}$$

The projections of a point $\mathbf{y}_i$ with respect to the new basis is $\mathbf{z}_i$ where,

$$\begin{aligned} \mathbf{z}_i &= [z_{i1},\ z_{i2},\ ...,\ z_{im}]^T, \qquad i = 1, ..., n, \\ \text{where } z_{ik} &= \mathbf{y}_i^T . \mathbf{u}_k, \qquad k = 1, ..., m. \end{aligned} \tag{3.6}$$

The eigenvectors $\mathbf{u}_1, \mathbf{u}_2, ..., \mathbf{u}_m$ are called *the principal directions* (or principal axes) of $\mathbf{x}_1, ..., \mathbf{x}_n$, and the projections of the p original points onto these vectors are called *the principal components* of the points belonging to χ.

Now if we look at principal components analysis from a different angle and consider the points in χ as samples from a multivariate normal distribution (n-variate in our case) in a statistical sense, then we can think of the eigenvectors $\mathbf{u}_1, \mathbf{u}_2, ..., \mathbf{u}_m$ as the m major axes of an ellipsoid derived from this distribution, where the axes are estimated from the sample. According to this point of view, then, the matrix $\mathbf{C}$ represents the sample covariance of the distribution, and is therefore called *the covariance matrix*. By finding the principal directions it becomes possible to study the variations of the data around the mean of the distribution. The first principal direction (i.e., the one corresponding to the highest eigenvalue) is where the data varies the most around the mean, the second one is the second highest direction of variation, and so on. Figure 3.1 shows the first principal direction of a two-dimensional normal distribution. All in all, we have performed a transformation that is equivalent to translating the axes origin around the the center of gravity (or mean) of the points, and then rotating them so that they are aligned with the directions of highest variation. Throughout this thesis, the statistical point of view of principal components analysis is adopted, since probabilistic analysis of the data will be used in our retrieval

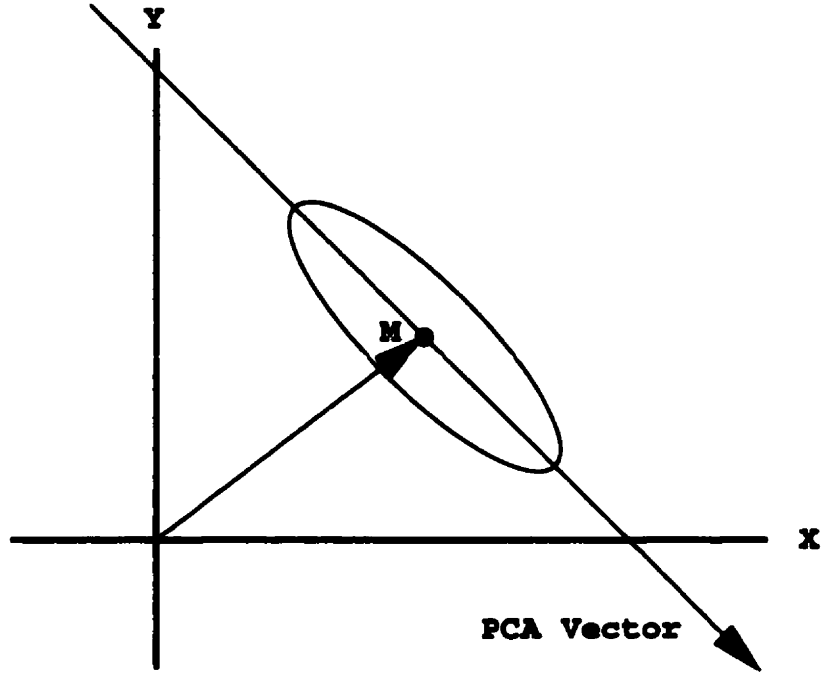


FIGURE 3.1. The first principal direction of a two-dimensional normal distribution. If a normal distribution has a mean $\mu = [\mu_1, \mu_2]^T$ and a covariance matrix C then the first principal direction is the first major axis of an ellipse representing the distribution.

methods as will be shown in later chapters. For more on principal components analysis see (Lebart, Morineau and Warwick 1984).

2. Computational Considerations

2.1. Reducing the Computational Complexity. If the dimension of the elements of χ is n then the covariance matrix $\mathbf{C}$ is $n \times n$. Since we are assuming that n is a very large number then the computation of $\mathbf{C}$ as well as the task of finding n eigenvectors becomes computationally expensive. Fortunately, we can get around this problem if the number of data points p is much smaller than the dimension of the space n ($p \ll n$). If this is the case, then the number of eigenvectors that have non-zero eigenvalues is $p - 1$ at most. Therefore, we do no need to calculate the n eigenvectors of $\mathbf{C}$. Instead we just need the first p eigenvectors. Furthermore, rather than finding the eigenvectors of the $n \times n$ matrix $\mathbf{A}\mathbf{A}^T$, we can first calculate the p eigenvectors of the $p \times p$ matrix $\mathbf{A}^T\mathbf{A}$, and from these eigenvectors find the eigenvectors of the original matrix. That way we reduce our calculations from dealing with n dimensions to dealing with only p dimensions, which is

a significant reduction in the computational complexity. We can see how this works by considering the following. Let $\mathbf{v}_1, \mathbf{v}_2, ..., \mathbf{v}_p$ be the eigenvectors of the matrix $\mathbf{A}^T\mathbf{A}$, that is,

$$\lambda \mathbf{v}_i = \mathbf{A}^T\mathbf{A}\mathbf{v}_i. \tag{3.7}$$

If we pre-multiply both sides by $\mathbf{A}$ we get,

$$\lambda \mathbf{A}\mathbf{v}_i = \mathbf{A}\mathbf{A}^T\mathbf{A}\mathbf{v}_i. \tag{3.8}$$

By comparing equations 3.8 and 3.4 we can clearly see that the vectors $\mathbf{A}\mathbf{v}_i$, where $i = 1, ..., p$, are the eigenvectors of the matrix $\mathbf{A}^T\mathbf{A}$, in other words,

$$\mathbf{u}_i = \mathbf{A}\mathbf{v}_i, \qquad i = 1, ..., n. \tag{3.9}$$

By following the previous steps we can only find the first p eigenvectors of $\mathbf{A}^T\mathbf{A}$ but for our purposes, we need at most $p - 1$ eigenvectors because the eigenvalues corresponding to the subsequent eigenvectors are all zero.

2.2. Flow of the Algorithm. Combining the information obtained in this chapter we obtain this simple, finite algorithm for finding the principal components of our data.

ALGORITHM 1.

(i) *Present the values of each object in a column vector* $\mathbf{x}_i$ $i = 1, ..., p$.

(ii) *Calculate the mean of all the objects* $\mu = \frac{1}{p}\sum_{i=1}^{p} \mathbf{x}_i$.

(iii) *Subtract each object* $\mathbf{x}_i$ *from the mean* μ *to obtain* $\mathbf{y}_i = \mathbf{x}_i - \mu$.

(iv) *Arrange the column vectors* $\mathbf{y}_i$ $i = 1, ..., p$ *into the* $n \times p$ *matrix* $\mathbf{A}$, *where* $\mathbf{A} = [\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_p]$.

(v) *Calculate the matrix* $\mathbf{A}^T\mathbf{A}$.

(vi) *Find the eigenvalues and eigenvectors of the matrix* $\mathbf{A}^T\mathbf{A}$. *The eigenvectors are named* $\mathbf{v}_1, \mathbf{v}_2, ..., \mathbf{v}_{p-1}$ *such that the eigenvalues associated with them are* $\lambda_1 > \lambda_2 > ... > \lambda_{p-1}$.

(vii) *Calculate* $\mathbf{u}_1, \mathbf{u}_2, ..., \mathbf{u}_{p-1}$, *the first p eigenvectors of* $\mathbf{A}\mathbf{A}^T$ *from* $\mathbf{u}_i = \mathbf{A}\mathbf{v}_i$.

(viii) *From* $\mathbf{u}_1, \mathbf{u}_2, ..., \mathbf{u}_p$ *take the first* m *eigenvectors (see Section 2.3 to see how to choose* m*), and consider these as the principal directions.*

(ix) *Project each vector* $\mathbf{y}_i$ *where* $i = 1, ..., p$ *onto the new basis* $\mathbf{u}_1, \mathbf{u}_2, ..., \mathbf{u}_m$ *to calculate* $\mathbf{z}_i$*, where* $\mathbf{z}_i = [z_{i1}, z_{i2}, ..., z_{im}] = [\mathbf{y}_i^T.\mathbf{u}_1, \mathbf{y}_i^T.\mathbf{u}_2, ..., \mathbf{y}_i^T.\mathbf{u}_m]$.

2.3. How to choose m. One question that we avoided to answer so far is: how many principal directions should we take, while still obtaining a good description of our data? Usually, the variations of the data around the mean are significant in the first few principal directions. The question is, where shall we stop? To be able to answer this question, we have to look at the relative importance of the eigenvectors. We can find this by looking at the eigenvalues. If we want to know how each principal direction contributes to describe the total variance of the system, then the following ratio is very useful,

$$r_k = \frac{\lambda_k}{\sum_{i=1}^{p} \lambda_i}. \tag{3.10}$$

Suppose then that want we our system to describe at least a percentage f% of the variance, then we take m such that $r_1 + r_2 + ... + r_m > \frac{f}{100}$.

3. Application to Image Indexing and Scene Classification

The purpose of this work is to be able to retrieve images from a database based on the content of the images themselves. Therefore, the objects we talked about in general terms in Section 1 are specifically images of real objects and real scenes. In general, images are represented by a large number of values. Therefore, principal components analysis can be applied to images to simplify their analysis.

Consider a large database of images where each image is 256 by 256 (typical size of an image). Each pixel in the image has associated with it a value ranging from 0 to 255 representing the gray scale level of the pixel, where 0 is black and 255 is white. In total, we have 256 by 256 = 65536 values for each image. if we treat the problem of image indexing based on a 65536-dimensional space, comparing each pixel with corresponding pixels from other images in the database, then our system will break down because the computational expense of processing the data will be huge, and the correct scenes we are

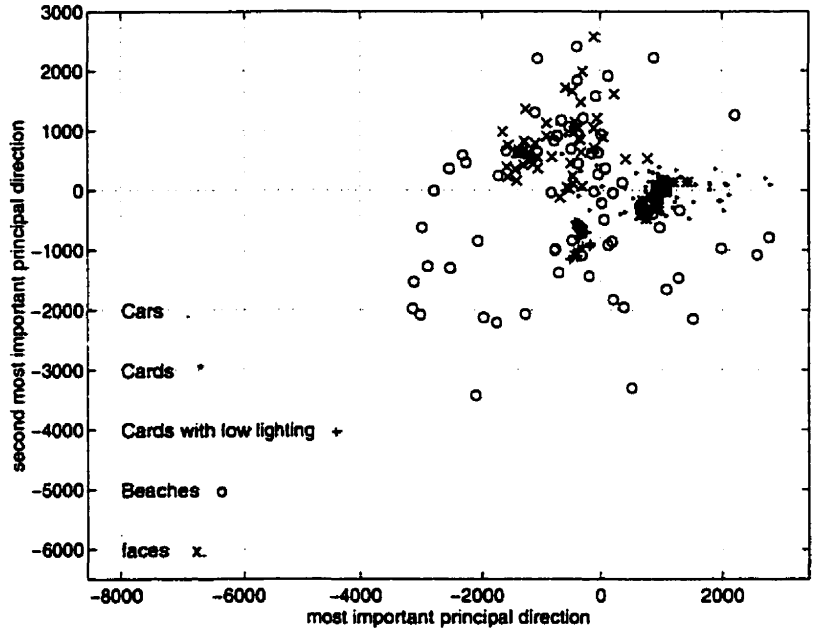


FIGURE 3.2. The first two principal components of a database of images containing four classes, human faces, cars, cards, and beaches.

seeking to extract will not be found in a reasonable amount of time. To solve this complexity problem, we seek to represent the images in terms of their projection intc the relatively low m-dimensional space which captures the important variations in the images to be analyzed. All this is performed by following the steps of Algorithm 1. Then each each image can be represented in terms of a projection on the new m axes and therefore will have m principal components. If we take m to be equal to, say, 20 principal directions, then we have reduced the dimensionality of each image from 65536 to 20, in the best possible linear projection. Of course, we are assuming that images are structurally similar and, therefore, will not be randomly distributed in the high dimensional space, but will vary only around a few directions. In Figure 3.2, the first two principal components of a collection of real images are plotted. The images are taken from a database containing cars, playing cards, human faces, and beach scenes.

4. The Most Discriminating Features

Principal components analysis has been shown to be the optimal linear subspace representation of a set of data. This means that the basis features generated have the most

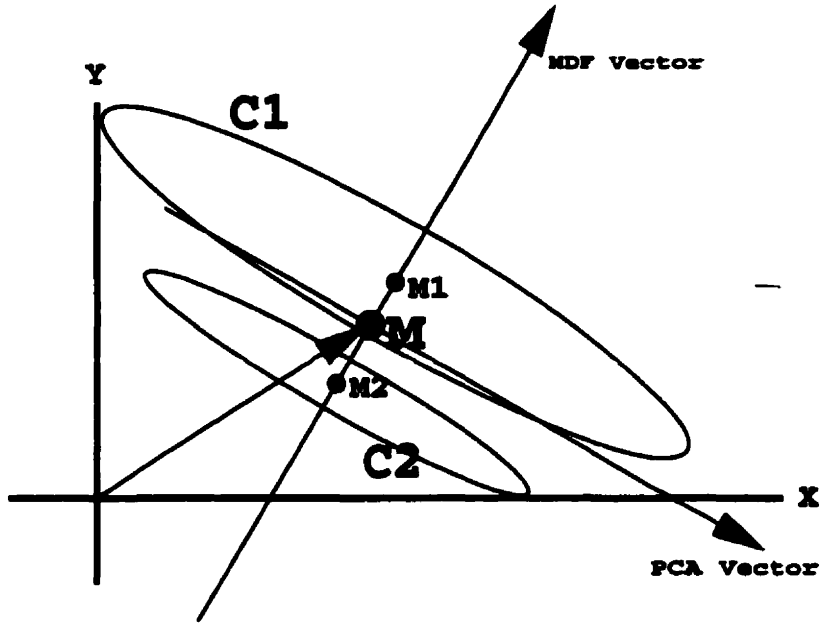


FIGURE 3.3. Representing two dimensional data in one dimension: we see that if we project both classes C1 and C2 along the first principal direction (labeled PCA vector) then the two classes will overlap and bad results for classification will result. However, if we project along the most discriminant feature (labeled MDF vector) then the two classes are clearly separable.

expressive power for this set of data. However, if the problem in hand is not object representation but object classification then the approach should take a slightly different point of view. The task now becomes finding the best basis for discriminating the various classes that form the database. With this in mind, Swets and Weng (1996) have shown that the basis generated in this case is different from the basis generated by principal components analysis, the reason being that the principal directions include variations in the data that may be irrelevant to how the classes are divided. They called the new vectors that form the basis the "Most Discriminating Features" (MDF) and have shown that the representation they produce gives better classification accuracy than the principal components analysis eigenvectors. As an intuitive argument supporting Swets and Weng's article, Figure 3.3 shows that the projections of the data points on the first principal direction will not result in the separation of the data into two distinct classes, although the variation of the data is expressed at its best. However, if we project the classes C1 and C2 on the first most discriminant feature in a linear projection then we can clearly separate the two classes from each other, and therefore obtain better classification results.

PROBLEM DEFINITION 2. *Let $\mathbb{R}^n$ be an n-dimensional Euclidean space, and let $\chi = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_p\}$ be a set of points belonging to $\mathbb{R}^n$. Furthermore, let these points form a set of classes $D_1, D_2, ..., D_l$. Find a subspace of $\mathbb{R}^n$ consisting of o vectors, where $o \ll n$, that will fit the p points belonging to χ in a manner that will maximize the separation between the different classes.*

Repeating equations 3.1 and 3.3, the mean of the whole set χ is,

$$\mu = \frac{1}{p}\sum_{i=1}^{p}\mathbf{x}_i, \tag{3.11}$$

and likewise the covariance matrix χ is

$$\mathbf{C} = \sum_{i=1}^{p}(\mathbf{x}_i - \mu)(\mathbf{x}_i - \mu)^T = \mathbf{A}\mathbf{A}^T. \tag{3.12}$$

However, each of the different classes $D_1, D_2, ..., D_l$ has its own mean, $\mu_1, \mu_2, ..., \mu_l$, and its own covariance matrix, $\mathbf{C}_1, \mathbf{C}_2, ..., \mathbf{C}_l$. These values are calculated in the same way the μ and $\mathbf{C}$ are calculated but by only including the members of the respective class.
The scattering within the classes is defined by the following matrix,

$$\mathbf{S}_w = \sum_{i=1}^{l}\mathbf{C}_i. \tag{3.13}$$

Whereas the scattering between the classes is defined as follows,

$$\mathbf{S}_b = \sum_{i=1}^{l}(\mu_i - \mu)(\mu_i - \mu)^T. \tag{3.14}$$

The matrix $\mathbf{S}_w$ is called the within class scatter matrix, whereas the matrix $\mathbf{S}_b$ is called the between class scatter matrix. To be able to separate the classes as much as possible we have to find a projection matrix that maximizes the between class scatter while minimizing the within class scatter. i.e., maximize the ratio $\frac{|\mathbf{C}|}{|\mathbf{S}|}$. It turns out that the eigenvectors of the matrix $\mathbf{S}_w^{-1}\mathbf{S}_b$ satisfy this requirement and therefore will form the basis of the most discriminating features. More can be found on discriminant analysis in (Wilks 1963).

There is one case, however, that can cause the algorithm to break down, and this is when the matrix $\mathbf{S}_w$ is singular. This happens when the matrix has zero eigenvalues, which means in our case that the number of data samples is smaller than the number of dimensions of each data point in the database (that is, $p < n$). Since we are assuming that our data is of very high dimensionality then this situation is bound to occur in all practical cases. We need therefore to represent the data in a much lower m-dimensional space where $p > m$, and then use the most discriminant analysis on this space where the degeneracy does not occur. As discussed earlier in this chapter, principal components analysis can perform the task of representing the data in a low dimensional space. Furthermore, we argued earlier that m (the dimensionality of the subspace) can be at most $p - 1$ since only the first $p - 1$ eigenvectors have non-zero eigenvalues, making the relation $p > m$ true at all time. This way, the matrix $\mathbf{S}_w$ can always be inverted and a solution can be found under all circumstances. Therefore, the overall discriminant analysis takes the form of two projections: the principal components analysis projection followed by the most discriminant analysis on the lower dimensional space. The flow of the algorithm is as follows.

ALGORITHM 2.

(i) *Perform principal components analysis on the objects belonging to the database* $\mathbf{x}_i = [x_{i1}, x_{i2}, ..., x_{in}]^T$ *where* $i = 1, ..., p$, *to obtain a representation in a much smaller subspace of* m *dimensions. The procedure follows Algorithm 1 and satisfies the relation* $p > m$. *Each representation of* $\mathbf{x}_i$ *in the lower dimensional space is labeled* $\mathbf{z}_i = [z_{i1}, z_{i2}, ..., z_{im}]^T$.

(ii) *Divide the objects in the database into classes* $D_1, D_2, ..., D_l$ *such that* $D_i \cap D_j = 0 \; \forall i \neq j$

(iii) *For each class* D_i, *where* $i = 1, ..., l$, *calculate the mean of the class* $\boldsymbol{\mu}_i$ *and its covariance matrix* $\mathbf{C}_i$.

(iv) *Calculate the within class scatter matrix* $\mathbf{S}_w$ *following equation 3.13.*

(v) *Calculate the between class scatter matrix* $\mathbf{S}_b$ *following equation 3.14.*

(vi) *Calculate the matrix* $\mathbf{S}_w^{-1}\mathbf{S}_b$.

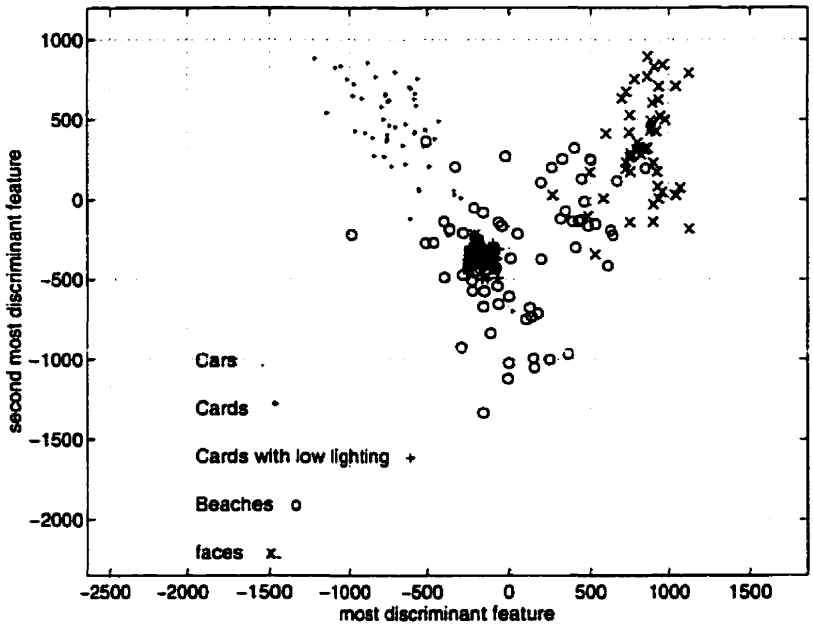


FIGURE 3.4. The components of the images based on the two most discriminating features instead of the two principal directions.

(vii) *Find the first* $o-1$ *eigenvalues and eigenvectors of* $\mathbf{S}_w^{-1}\mathbf{S}_b$. *The eigenvectors are labeled* $\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_{o-1}$ *such that the eigenvalues associated with them are* $\rho_1 > \rho_2 > ... > \rho_{o-1}$.

(viii) *Project each vector* $\mathbf{z}_i = [z_{i1}, z_{i2}, ..., z_{im}]^T$, *where* $i = 1, ..., p$, *onto the basis* $\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_{o-1}$ *to calculate* $\mathbf{q}_i = [q_{i1}, q_{i2}, ..., q_{io}]^T = [\mathbf{z}_i^T.\mathbf{w}_1, \mathbf{z}_i^T.\mathbf{w}_2, ..., \mathbf{z}_i^T.\mathbf{w}_m]^T$.

Figure 3.4 illustrates the difference between principal components analysis and the most discriminating features. Using the same images as in Figure 3.2, we can clearly see that in the new space objects of the same class are clustered more tightly than in the principal directions space.

5. Summary

The analytical framework with which the images are to be analyzed is now established. Through this chapter, it has been shown that it is possible to express all the images in a database in terms of a few meaningful coefficients. Two important issues have to be realized: (i) principal components analysis is the optimal description of a set of high dimensional data in terms of a few coefficients, and (ii) the most discriminant features are the best possible

set of vectors for class separation. I will show how the former method can be used to more accurately retrieve the closest images to a given example image, and how the latter can be better suited for Bayesian classification. In the next chapter, I will discuss the limitations of using both methods directly on the intensities of the images, and propose two ways to improve on traditional appearance-based image retrieval methods.

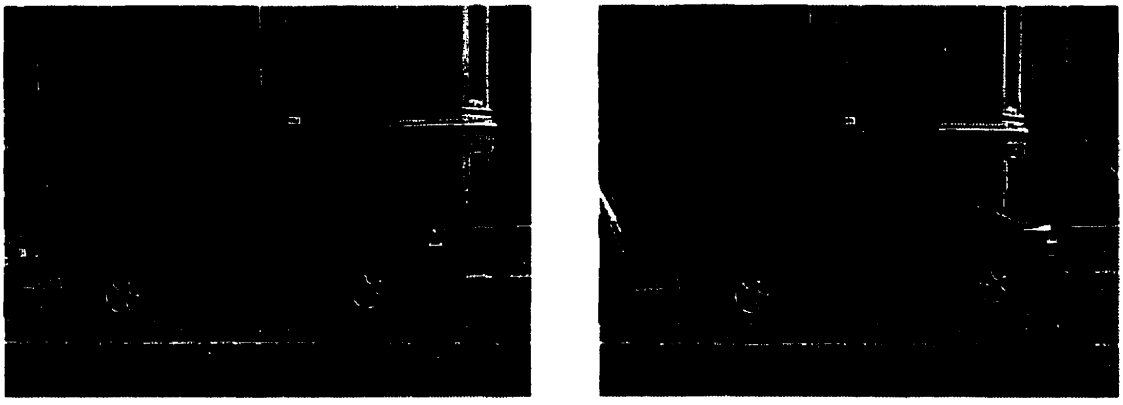
CHAPTER 4

The Indexing Phase

This chapter is concerned with the implementation of the indexing phase of the content-based image retrieval system. The indexing phase, as mentioned in Chapter 1, is the extraction of image features relevant to the retrieval process, where the features may or may not be directly related to the physical objects in the image. In Chapter 2 we discussed the general difficulties of having a "universal" feature that works for all systems under all circumstances. In this thesis, we consider principal components analysis to be a good tool for representing real images. It has several advantages that we exploit and disadvantages that we try to overcome. Section 1 discusses these issues illustrating the shortcomings of appearance-based representations. The next two sections propose two ways to cope with these shortcomings. Specifically, in Section 2 we propose a better solution for indexing well framed images, while in Section 3 we propose a better solution for indexing complex scenes. Section 4 includes a short summary of the ideas discussed in this chapter.

1. Improving on Appearance-Based Methods

The appearance of an image is a combination of all the factors that together, characterize the image at the specific moment it has been taken. As an example, the appearance of the image in Figure 4.1(a) is a function of the shapes of all the objects in the image (cars, buildings, tree, road, etc.), as well as on the relationship between all the objects, the position of the camera, the amount of lighting, the sources of light, the surface albedo of all the objects, and other factors that are generally hard to determine. The appearance of an image is a global property, which means it represents the image taken as a whole block.



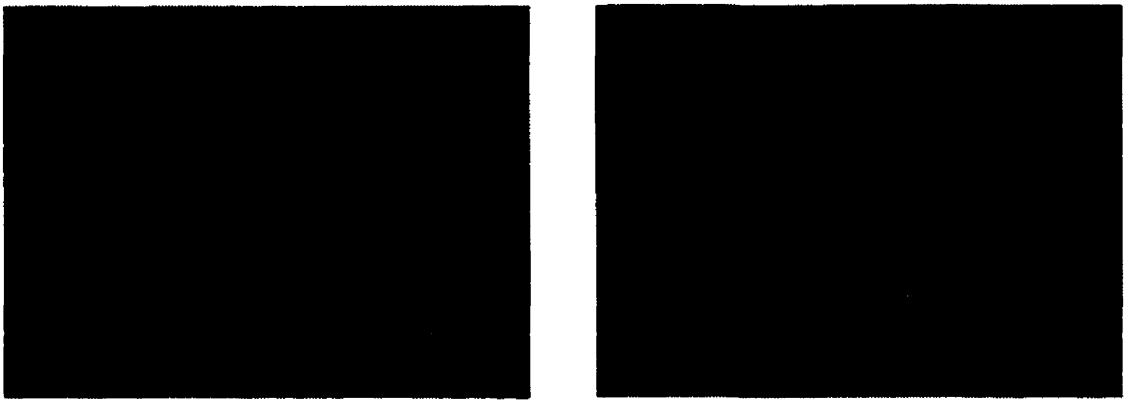
(a) (b)

FIGURE 4.1. A scene with a car taken from two different camera angles

Appearance-based methods have the advantage of being able to deal directly with real images in a quick and reliable manner. However, since the appearance of an image depends on so many factors other than the inner properties of the physical objects present in the image, then this makes it very sensitive to changes in any of these factors. For example, if the position of the camera angle with respect to the objects changes, as in the difference between the two images in Figure 4.1, then the appearance of the image might change considerably in some cases, although the objects present in the scene are practically the same. This is problematic for image retrieval systems that are concerned with retrieving images containing similar objects, and where the differences in the amounts of lighting and camera angle are of little importance to the user. In order to solve the problem of the sensitivity of appearance-based methods, we have to extract directly from the images, those features that are themselves insensitive to the factors mentioned above. This task is far from being simple for the following reasons,

(i) It is hard to determine all the factors that characterize the visual appearance of an image. Furthermore, it is hard to separate the different factors from each other.

(ii) It is impossible to find a universal feature that is invariant to all factors. This means that features in general would stay invariant when changing some factors, but would fluctuate when changing other factors.



(a) (b)

FIGURE 4.2. Two images of the same card but with different lighting conditions

However, in this work, we consider that the most important factors that change the appearance of objects in an image are (i) the scene illumination characteristics and, (ii) the camera position, with respect to the objects and with respect to ground. Having determined this, we extract two different features, one quasi-invariant to the changes in the lighting conditions and the other quasi-invariant to the changes in camera position. These two intermediate representations are the subject of interest of the next two sections. After having extracted the relevant features, we can then index the images based on the appearance of the extracted features, and not on the appearance of the intensity image.

2. Analysis of Well Framed Images

DEFINITION 1. *Well framed images are images where only a small variation in the size, position, and orientation of the objects in the images is allowed (Swets and Weng 1996).*

There are a number of computer vision applications whose images of concern are well framed. This constraint is very useful because it considers the problem of finding the objects in the images already solved, i.e., the figure/ground problem. Objects are well localized, and this makes the comparison between the images much easier than in the general case. Applying appearance based representations on well framed images allows for good results in some task-specific applications, like face tracking (Turk and Pentland 1991), or parametric representations of simple objects (Nayar, Murase and Nene 1996).

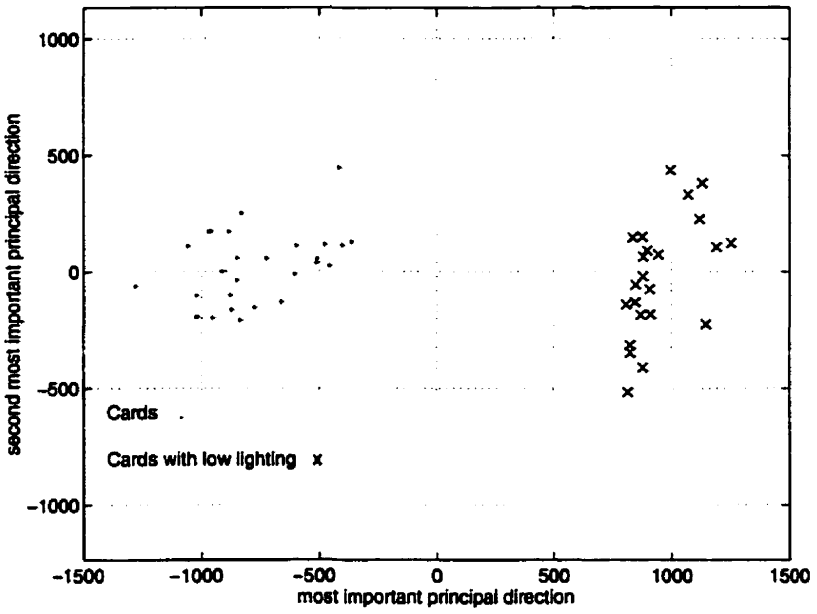


FIGURE 4.3. The first two principal directions of images of playing cards. Notice the two distinct clusters due to a change in the lighting conditions.

Unfortunately, appearance based methods still face the problem of being sensitive to the changes in the lighting conditions. For example, Figure 4.2 shows two images of the same playing card taken under similar conditions, i.e., there are very small variations in the camera position with respect to the card. The difference between the two images is that one source of light was turned off while taking the second image. If we apply principal components analysis to the absolute image intensities to index these images, then the two images will be far away from each other with respect to images of other cards in the new basis. This can be seen from the plot in Figure 4.3 which shows the first two principal components of a database of images containing playing cards. In this case, each card has been imaged twice, once with an extra light source and once without it. It can be clearly seen from the plot that there are two distinct clusters, one describing the cards with the extra light source, and the other describing the cards without the light source. If the image retrieval system is required to return the closest picture to, say, the card in Figure 4.2(a), then using classical appearance-based methods will cause the system to return at least all the cards in the database being imaged with the extra light source (i.e., the images represented by the dots in Figure 4.3) before returning the required picture, namely the one in Figure 4.2(b).

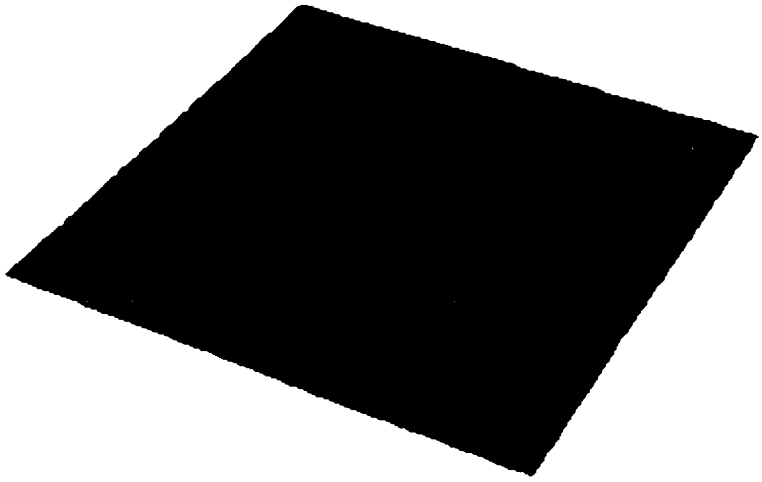


FIGURE 4.4. The Marr/Hildreth Laplacian of a Gaussian edge operator

In this thesis we deal with the problem of sensitivity to the lighting conditions by extracting from the map of absolute intensities (i.e., the original image), another map representing the image but which is insensitive to the absolute amount of lighting in the image. More precisely, we seek a map that identifies in the image, the positions that indicate interesting physical events, or edges. Such a representation is directly related to the shape of the objects in the image, and is therefore largely (quasi) invariant to external factors. Physical edges of objects have two important properties: (i) their structure in the world arises at different scales and, (ii) they are independent of the amount of light reflected from the object into the camera.

Considering the above properties, Marr and Hildreth (1980) observed that to locate edges, a first or second order differential operator is needed. In addition, the size of the operator should be easily changed so that it can be tuned to act at any scale to detect blurry edges as well as highly localized ones. Therefore they showed that a good edge detection operator can be defined mathematically as the Laplacian of a Gaussian, $\nabla^2 G$, where $\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$ and $G(x,y) = \frac{1}{2\pi\sigma^2}\exp(-\frac{x^2+y^2}{2\pi\sigma^2})$. An illustration of the operator, which looks like a Mexican hat, is shown in Figure 4.4. Considering psychophysical evidence,

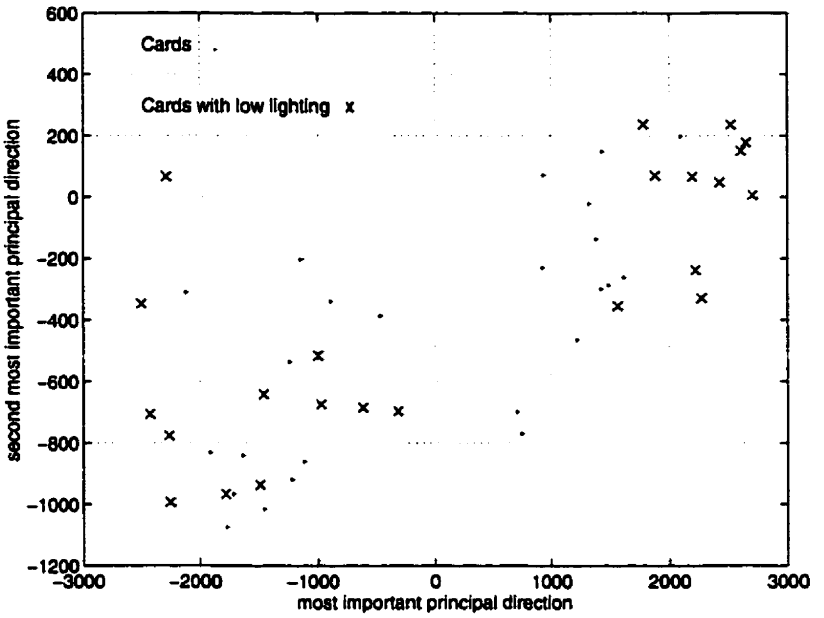


FIGURE 4.5. The first two principal directions of images of playing cards, being analyzed on the level of their edge map. As opposed to Figure 4.3, here the plot shows that the effect of the extra light source is not present.

they argued that this operator approximates the circular surround operators present in retinal ganglion cells (Rodieck 1965).

To be able to locate the edges in an image, the operator is applied as a two-dimensional mask on the original image, and passes through each pixel in the image (except for the image boundaries). The output is the result of the linear convolution of the mask and the image in the neighborhood of each pixel. Then, since the Laplacian is a second order derivative, then step changes in intensity can be detected whenever there is a zero crossing in the output of the operator. This means that, when the sign of the output of the operator changes from one pixel to a neighboring pixel, then an edge point has been localized. We call the edges located the *zero crossings* of the image.

In this work, we apply the operator on the original image at four different scales, from the finest scale to detect the small highly localized features to the coarsest to detect the general outline of the object. In all cases, the localized zero crossings are a property of the shape of the objects present in the image. Therefore, in addition to being insensitive to the absolute intensity values in an image, these maps contain more salient information about the shape of the object, separating all other factors that might influence the image analysis

process. This procedure results in four zero crossing maps for every image in the database. We combine the four output maps together in a very long array, or similarly, a point in a very high dimensional space. Next, we perform principal components analysis following Algorithm 1 in Chapter 3 on the new high dimensional point instead of the original image. To see the effect of analyzing the edge map instead of the intensity image, the plot in Figure 4.5 shows clearly that the two clusters representing the playing cards with different lighting conditions have merged together, discarding the effect of the extra light source. The difference in the images now, are only due to the shapes of the different objects, as well as their reflectance characteristics.

For the purposes of this work, the criteria for choosing a suitable edge detection operator are based on the following arguments. First, since it is very important for an image retrieval system to render near real-time responses to the queries, then the need emerges for a fast and practical operator. Second, the operator must be easily adaptable to different scales, where for each scale, it filters signals whose frequencies lie inside a certain range, acting like a band pass filter. Various linear edge detection operators satisfy the above conditions. The Marr/Hilderth operator was a convenient choice.

3. Analysis of Complex Scenes

We discussed in the previous section how the analysis of images is considerably less complicated when the images are well framed. If an image consists of a few objects with a very simple background, and if the objects' locations in the image are well known, then applying the Marr/Hildreth operator will considerably improve the analysis because it can compare the edge characteristics of the objects directly with other well framed images. However, the vast majority of images are very hard to analyze and compare because: (i) they consist of more than just a few objects, (ii) the background is complicated, (iii) it is hard to locate the objects in the image and, (iv) it is hard to separate all the objects from the background. While the human visual system has solved the above problems, it remains extremely hard for a computer to deal with them. The problems mentioned above are still open problems in the field of computer vision and have been the subject of research since the mid-sixties (Roberts 1965).

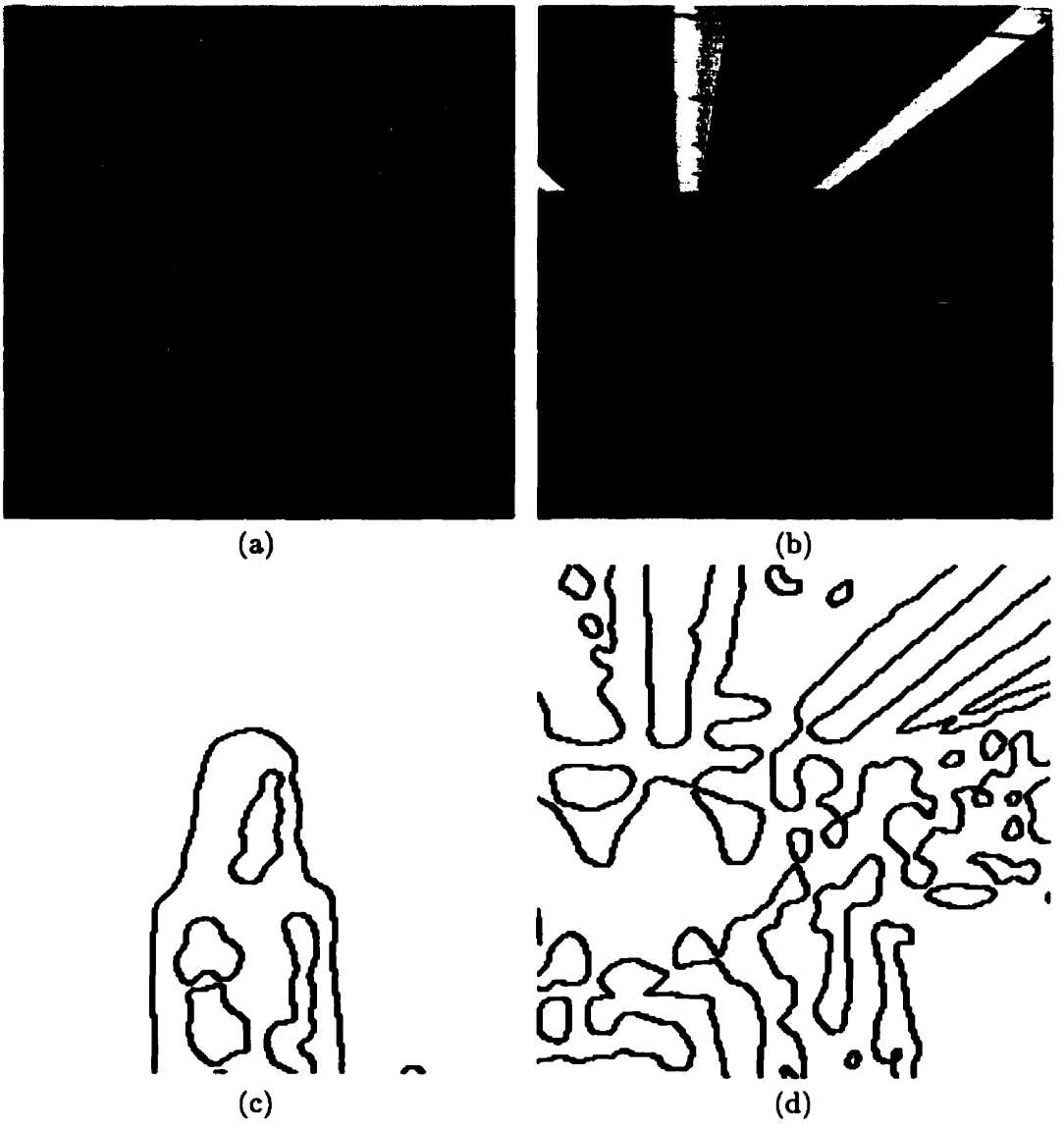


FIGURE 4.6. (a) and (b) two images, one with a simple background and one with a complex background. (c) and (d), the zero crossings of the images

In this work, the content-based image retrieval system is fully automatic (i.e., it does not require any human interference), and can index complex scenes. In the same fashion as in the well framed images analysis, we use principal components analysis to represent each image with just a few coefficients in an optimal way. However, unlike the well framed images case, the use of the Marr/Hildreth operator as an intermediate representation is not adequate. Figure 4.6 (a) and (b) shows two types of images, one with a simple background and one with a complex background. The zero crossings of these images are shown in Figure

4.6 (c) and (d). We can notice, that the zero crossings in (c) clearly designate a human face and body while the ones in (d) have little connection with the physical objects comprising the scene. If we apply a more complicated edge detection operator to the original image, then we might get more accurate results, but this will slow down the response time considerably. Another important reason that makes the zero crossings inappropriate for scene analysis is that the appearance of the image is very dependent on the position of the camera with respect to the objects. This fact is true whether we perform principal components analysis on the original image or on the zero crossings. In the case of well framed images we do not face this problem because we assume that there is very little change in the objects' positions. What we need here is a intermediate representation that is also invariant to the positioning of the camera. Finally, we still have not solved the problem of locating the objects of interest in the scene. The intermediate representation that is required needs to satisfy the following properties,

(i) It has to be computationally inexpensive.

(ii) It has to capture the important information in the scene, taking the above constraints into consideration.

(iii) It has to be invariant to the camera position with respect to the objects, as well as to translation and rotation.

In short, after defining the constraints imposed by the task at hand, we have to find an appropriate representation describing general scenes. In this case, although we do not impose constraints on the type of images comprising the database, we can profit from the constraint that the task consists of comparing images rather than describing the objects in them. In other words, since we consider it very hard to extract the physical objects in a complex scene; then a good representation would be one that can characterize the scene without having to deal with the individual objects, or their positions. In this thesis, we choose an intermediate representation that represents the image in the frequency domain rather than in the spatial domain. Specifically, we use the Fourier Transform to describe the frequencies that characterize each scene and segregate it from other scenes. A representation in the frequency domain is appropriate because it does not directly deal with the positions of

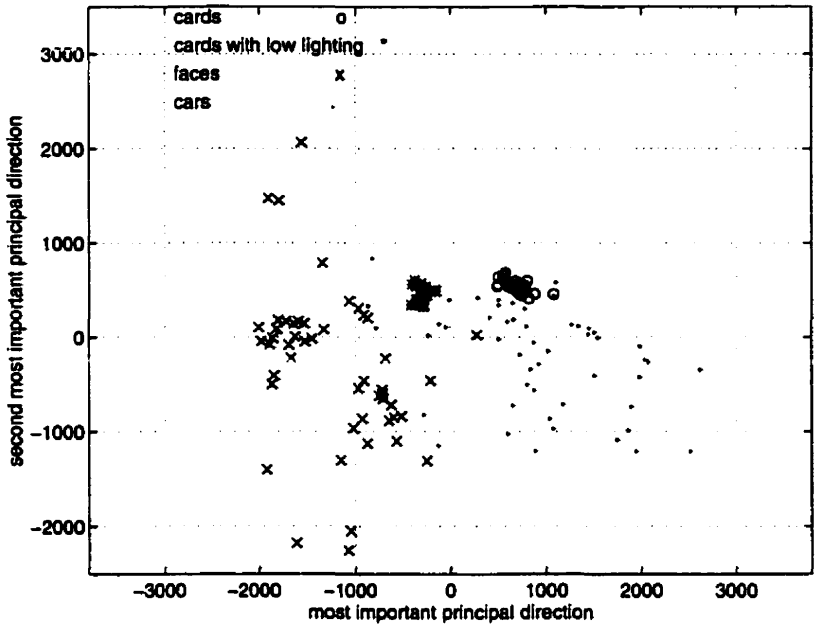


FIGURE 4.7. The first two principal directions of a database comprising images of playing cards, as well as cars, and human faces. Principal components analysis is performed on the original images.

the objects in the scene, but is rather a feature of the scene in general. The two-dimensional discrete Fourier Transform is defined as follows,

$$F(u,v) = \frac{1}{N}\sum_{i=0}^{N-1}\sum_{j=0}^{N-1} I(i,j)\ \exp\left[-\frac{2\pi i}{N}(iu+jv)\right]. \tag{4.1}$$

The transform is a mapping of the image $I(i,j)$ from the space domain to the frequency domain. The Fourier transform of the image is $F(u,v)$. We can consider the Fourier Transform as a two-dimensional histogram of the frequencies in the scene. In this thesis, we only use the magnitude of the Fourier Transform $\|F(u,v)\|$ because it is represented by real numbers only, and is furthermore invariant to translation and rotation in the two-dimensional signal. Furthermore, we expect that two images of the same scene taken under different camera positions to render closely resembling magnitudes of their respected Fourier Transforms.

Once the Fourier Transforms of the images have been calculated, principal components analysis is performed on them rather than directly on the absolute intensities. Figure 4.7

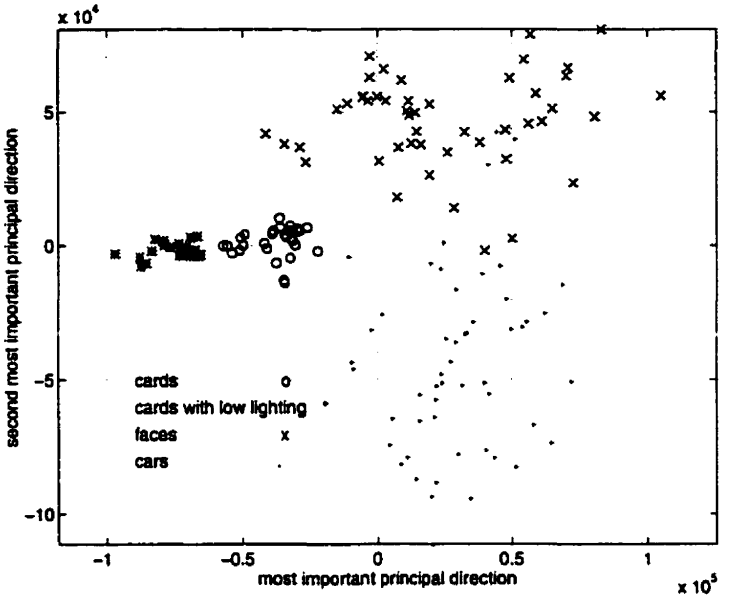


FIGURE 4.8. The first two principal directions of a database comprising images of playing cards, as well as cars, and human faces. Principal components analysis is performed on the Fourier Transform of the images.

shows a plot of the first two principal directions of a database comprising playing cards (the same as in Figures 4.3 and 4.5), car scenes, and scenes including humans in front of simple and complex backgrounds. Here principal components analysis is performed on the images themselves. We can notice again that the parameters of the playing cards (with and without the extra light source) are far apart from each other with respect to other images. Furthermore, the parameters of the human faces are scattered considerably. If we compare this plot to the plot in Figure 4.8, then we can see that the parameters of the cards are much closer relative to the other classes and more importantly can now be easily grouped in one class distinct from the others. The same thing happens to human faces where the parameters are more grouped than in the first case. This is the effect of analyzing the frequency content of the scene rather than having to worry about the spatial structure.

4. Summary

In this chapter, we discussed the inadequacies of classical appearance-based methods (i.e., performing image analysis on the absolute intensity values of the pixels). We argued

that it would be more useful to apply appearance-based methods to intermediate representations of the images. Unfortunately, there isn't one intermediate representation that is well suited to represent all possible cases. Rather the intermediate representations have to include features representative of the images and the objects in the images depending on the problem at hand. In the case of well framed images, where we know where the objects are located, we compare the images based on the shapes of the objects under different scales. We use the zero crossings calculated from the Marr/Hildreth operator as the shape characteristics. On the other hand, we use the magnitude of the Fourier Transform of complex scenes as an intermediate representation. The difference here is that the vision system does not know the location of the objects in the scene, but still needs a feature that describes the scene characteristics. The magnitude of the Fourier Transform does just that.

The next chapter will address the issues concerning the retrieval of the images from a large database.

CHAPTER 5

The Retrieval Phase

The previous chapter discussed the indexing phase, or what consists of the work that the system executes *off-line*. In this chapter, we address the issues concerning the on-line stage or more specifically, what happens during the image retrieval process. In Chapter 1 we defined the main task of a content-based image retrieval system. For more emphasis, we repeat the task here.

> *Given a query represented by an unknown example image, return from the database of known images, the images closest to the query. Furthermore, if the images in the database are separated into several classes, then classify the query as belonging to one of these classes.*

The next section discusses the process of returning the closest images to the query. Sections 2 and 3 are concerned with the possible classification of the query: Section 2 discusses the *k nearest neighbors* rule for classification, while Section 3 presents a Bayesian framework for probabilistic classification of the queries. We summarize the ideas discussed in the chapter in Section 4.

1. Returning the Closest Images

The main purpose of an image retrieval system is to return the closest images satisfying some type of query. In our system, the query takes the form of an example image, and the retrieval task consists of returning the images that the system "thinks" are the closest to

the example image. The following algorithm explains the necessary steps needed to perform this task.

ALGORITHM 3.

(i) *Pre-process the query by calculating the appropriate intermediate representation of the image. More precisely, if the database consists of well framed images, then locate the zero crossings of the query at four different scales. On the other hand, if the database consists of general scenes then calculate the magnitude of the Fourier Transform of the query.*

(ii) *Find the principal components representing the query by projecting the values of the intermediate representation on the principal directions* $\mathbf{u}_1, \mathbf{u}_2, ..., \mathbf{u}_m$ *computed in the off-line stage.*

(iii) *Calculate the* L_2 *distance (Euclidean Distance) between the principal components of the query and the principal components of each image in the database.*

(iv) *Sort the calculated* L_2 *norms from the smallest distance to the largest.*

(v) *Return the* N *images corresponding to the* N *lowest* L_2 *distances*

2. The k Nearest Neighbors

The image retrieval system developed in this work does not merely return the closest images to the query, but is also augmented with a "classifier". Most of the image databases can be logically divided into several groups of images, where each group represents different types of scenes like beaches, flowers, athletes, etc. If the query type can belong to one of the classes comprising the database, then it would be helpful for the system to return a decision on class membership for the unknown query. That way, if the search needs to be refined, the query would not be applied to all the images in the database but just to the images belonging to its class. Our system classifies the query in two different ways, the first one is the subject of this section.

Let the image be classified as belonging to one of l classes of images denoted D_i, where $i = 1, 2, ..., l$, and let $P(D_i)$ denote the a priori probability of occurrence of objects belonging to class D_i. Let $\mathbf{a} = (a_1, a_2, ..., a_m)$ denote a set of measurements made on the object to be classified, and let $P(\mathbf{a}|D_i)$ be the probability density function of D given that the pattern

on which **a** was observed belongs to class D_i. The rule that minimizes the probability of misclassification in making a decision on **a** is to choose D_i such that

$$P(\mathbf{a}|D_i)P(D_i) > P(\mathbf{a}|D_j)P(D_j) \text{ for all } j \neq i. \tag{5.1}$$

The resulting Bayes (optimal) probability of misclassification is (Toussaint 1974),

$$P_e^B = 1 - \int \max \{P(\mathbf{a}|D_i)P(D_i)\}\, d\mathbf{a}. \tag{5.2}$$

To be able to use the rule in equation 5.1, the a priori probabilities $P(D_i)$ and the class conditional probability density functions $P(\mathbf{a}|D_i)P(D_i)$ must be known for all classes. Since in most cases these probabilities are not known, one resorts to a non-parametric decision rule; that is, no a priori knowledge concerning the underlying distribution of the data is required.

One of the most attractive non-parametric decision rules is the k nearest neighbors (knn) rule. Let $(\mathbf{A}, \mathbf{\Lambda}) = \{(\mathbf{a}_1, \lambda_1), (\mathbf{a}_2, \lambda_2), ..., (\mathbf{a}_f, \lambda_f)\}$ be a data set of features comprising a database, where $\mathbf{a}_i$ denotes a set of measurements made on an object, and λ_i denotes the class label of the object. Let A be a new object to be classified and let $A^{(1)}, A^{(2)}, ..., A^{(k)} \in [\mathbf{a}_1, \mathbf{a}_2, ..., \mathbf{a}_f]$ be the k closest features to **a**. The knn decision rule classifies **a** as belonging to class λ^*, the one that has the highest score in the k nearest neighbors $A^{(1)}, A^{(2)}, ..., A^{(k)}$ among all other classes. Let $P_e^{knn} = P(\lambda \neq \lambda^*)$ be the probability of misclassification of the rule, where λ is the true class of **a**. If k is correctly chosen, then P_e^{knn} will approximate the optimal Bayes error P_e^B as much as desired. Unfortunately, it is hard to determine what the best value of k is. Still, there are two important criteria for choosing a good value for k and these are: (i) $k \to \infty$ as $f \to \infty$, and (ii) $\frac{k}{f} \to 0$ as $f \to \infty$.

In this work we choose k such that $k = \sqrt{f}$. The features $\mathbf{a}_1, ...\mathbf{a}_f$ represent the principal components of the database images, and the principal components of the query image are A. The big advantage of using knn is that it performs well even though no prior information about probability distribution is used. The problem is that it assigns the query image to one and only one class. Generally, one would like to have a probability measure representing the belief that the query belongs to a specific class. This is the subject of the next section.

3. Bayesian Classification

Instead of a single solution, we seek a method that generates a measure of confidence in various classes within the context of image classification. The problem we are addressing requires us to infer the class to which the query belongs. This problem can be expressed as finding the posterior probability that the measurement belongs to a certain class. The difference with the previous section lies in the fact that there is no rule to apply for choosing a class among other classes. All the system does is calculate probabilities, it does not decide on which class to choose. Instead it delays this decision as much as possible, and leaves it to the external agent to assess the quality of the result it obtains. This is a desirable feature of a system because in some cases you need to make a decision only when you have absolute certainty about the information concerning the measurement, while in other cases this certainty is not a major requirement.

The posterior probability that an unknown measurement $\mathbf{a} = (a_1, a_2, ..., a_m)$ belongs to a class D_i where $i = 1, ..., l$ is $P(D_i|\mathbf{a})$. Bayes rule states that this probability can be expressed as follows,

$$P(D_i|\mathbf{a}) = \frac{P(\mathbf{a}|D_i)P(D_i)}{P(\mathbf{a})}, \tag{5.3}$$

where $P(\mathbf{a})$ is the normalization factor, and can be calculated from the theorem of total probability,

$$P(\mathbf{a}) = \sum_{j=1}^{l} P(\mathbf{a}|D_j)P(D_j) = P(\mathbf{a}|D_1)P(D_1) + ... + P(\mathbf{a}|D_l)P(D_l). \tag{5.4}$$

As discussed in the previous section, we do not know the a priori probabilities $P(D_i)$, nor do we know the conditional probability density functions $P(\mathbf{a}|D_i)P(D_i)$. However, from observations of the problem at hand, we can make some fairly reasonable assumptions about them. First of all, since we do not have any prior information about the image classes, we have no reason to believe that one class is more probable than the other classes. Therefore we assume that all the prior probabilities in the classes are equal, that is $P(D_i) = \frac{1}{l}$ for

all $i = 1, ..., l$. Next, we assume that image classes can in general be approximated by a multivariate Gaussian (normal) distribution $G(\mathbf{X})$, where,

$$G_i(\mathbf{X}) = \frac{1}{\sqrt{(2\pi)^m |\mathbf{C}_i|}} \exp\left[-\frac{1}{2}(\mathbf{X} - \boldsymbol{\mu}_i)^T \mathbf{C}^{-1}(\mathbf{X} - \boldsymbol{\mu}_i)\right], \tag{5.5}$$

where $\boldsymbol{\mu}_i$ is the class mean and C_i is the class covariance. Of course, these values are not known in advance but have to be estimated from the data. That is why we call them the *sample* means and covariances, and are calculated as follows,

$$\boldsymbol{\mu}_i = \frac{1}{\nu_i} \sum_{j=1}^{\nu_i} \mathbf{a}_{ji}, \tag{5.6}$$

and,

$$\mathbf{C}_i = \frac{1}{\boldsymbol{\mu}_i} \sum_{j=1}^{\nu_i} (\mathbf{a}_{ji} - \boldsymbol{\mu}_i)(\mathbf{a}_{ji} - \boldsymbol{\mu}_i)^T, \tag{5.7}$$

where the $\mathbf{a}_{ji}$'s and are the elements belonging to class i and ν_i is the number of elements in class i. The problem of finding the posterior probabilities of all the classes given the unknown measurement reduces to,

$$P(D_i|\mathbf{a}) = \frac{G_i(\mathbf{a})}{\sum_{j=1}^{l} G_j(\mathbf{a})}. \tag{5.8}$$

The natural logarithm of the above posterior probability can be written as follows,

$$\log P(D_i|\mathbf{a}) = -\frac{1}{2}(\mathbf{a} - \boldsymbol{\mu}_i)^T \mathbf{C}^{-1}(\mathbf{a} - \boldsymbol{\mu}_i) + K, \tag{5.9}$$

where K is a constant and can be discarded. The rest of the term on the right side is called the "Mahalanobis distance", and can be used as a measure of confidence in the class D_i with covariance C_i. The Mahalanobis distance can be used without a prior assumption about the nature of the density functions of the classes. However, it is equivalent to the log of the posterior probability if the density function of all the classes are multivariate normal

distributions. Generally, all we need to know to calculate the Mahalanobis distance is the mean and covariance of the class.

4. Summary

In this chapter, we discussed how our image retrieval system returns an answer to the user's query. Taking a query in the form of an example image, the system pre-processes the image, calculates the corresponding principal components, calculates the difference between the query's principal components and the principal components of all the images in the database, then returns the images that correspond to the closest Euclidean distances. Furthermore, the system classifies the image in two different ways. The first one is the k nearest neighbors (knn), where no assumption about the distribution of the classes is required and where, if k is well chosen, the probability of misclassification is as close to optimal as desired. This method assigns the query image to one and only one of the classes. The second classification method on the other hand, does not attribute the image to one class but finds the posterior probability that the unknown image belongs to all the classes comprising the database. It uses a probabilistic framework where assertions are represented by conditional probability density functions. Although this method includes assumptions about the nature of the class distributions that may not hold in some cases, it permits an external agent to assess the quality of the information obtained, and make informed decisions as to what action to take next. In our system, when we use knn, we represent the images using the principal directions. On the other hand, when we use Bayesian classification, we represent our images using the most discriminating features. The reason for making this distinction is that when we consider the data to be clustered in several classes as in the probabilistic case, we would like to choose the features that will maximize the distances between these classes. On the other hand, the task of extracting the nearest neighbors of an item requires optimal descriptive power rather than optimal discrimination.

Assuming that the components of all the images in the database have been computed beforehand (both in the principal directions space and the discriminating features space), we summarize the work done by the classification module by the following algorithm,

ALGORITHM 4.

(i) *Find the k nearest neighbors to the query image from Algorithm 3.*

(ii) *Calculate which class D_i is represented the most among the k nearest neighbors.*

(iii) *Assign the query image as belonging to class D_i.*

(iv) *Compute the most discriminating components of the query by projecting the values of the intermediate representation on the most discriminating features $\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_{o-1}$ computed in the off-line stage.*

(v) *Compute the mean and the covariance of each class from equations 5.6 and 5.7.*

(vi) *Assuming that all the classes are multivariate normal distributions, find the posterior probability of each class given the query $P(D_i|A)$ from applying equation 5.8 to all the classes.*

CHAPTER 6

Experimental Results

1. The System

In the previous chapters, we discussed the theory supporting all the phases of the content-based image retrieval system we built. Here, we put the system into practice, and show how we integrate all the parts to achieve a completely functional and practical image retrieval tool. The following algorithm summarizes the operation.

ALGORITHM 5.

(i) *Calculate the appropriate intermediate representations for all the images in the database, depending on whether the images are well framed or not.*

(ii) *Find the principal directions* $\mathbf{u}_1, \mathbf{u}_2, ..., \mathbf{u}_m$ *by applying principal components analysis (Algorithm 1) on a test set of intermediate representations of images. The test set has to be appropriately chosen so that it is representative of the full database.*

(iii) *Compute the* m *principal components of all the images in the database by projecting their intermediate representations on the eigenvectors* $\mathbf{u}_1, \mathbf{u}_2, ..., \mathbf{u}_m$. *Each image* I_i *is now represented by a much smaller vector* $[z_{i1}, z_{i2}, ..., z_{im}]^T$. *Empirical analysis of the data concluded that 20 eigenvectors renders a good description of the data variation. Therefore, in all the experiments included in this chapter, we set* m *to be equal to 20.*

(iv) *Find the most discriminating vectors* $\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_{o-1}$ *by applying Algorithm 2, on the principal components of the data.*

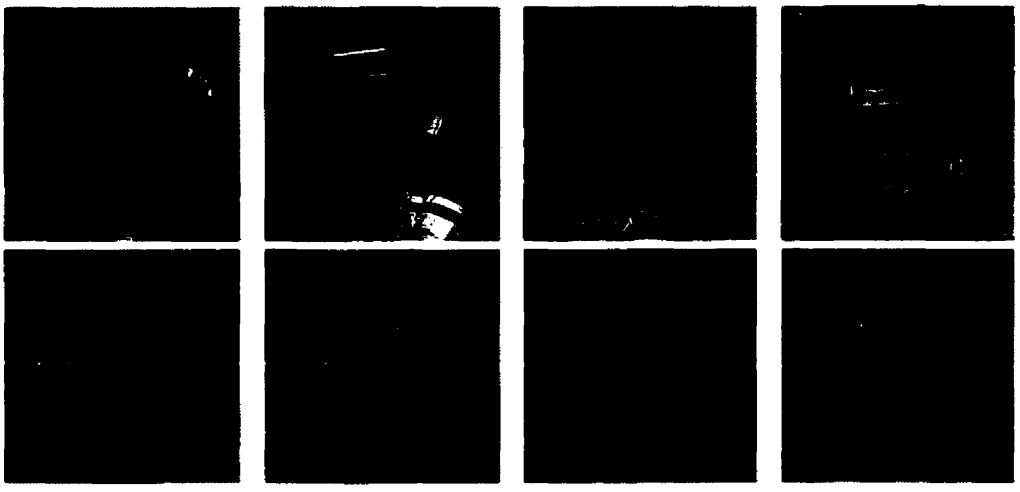


FIGURE 6.1. A database of faces

(v) *When the user provides a query in the form of an example image, follow the steps in Algorithm 3 to obtain the N closest images to the query.*

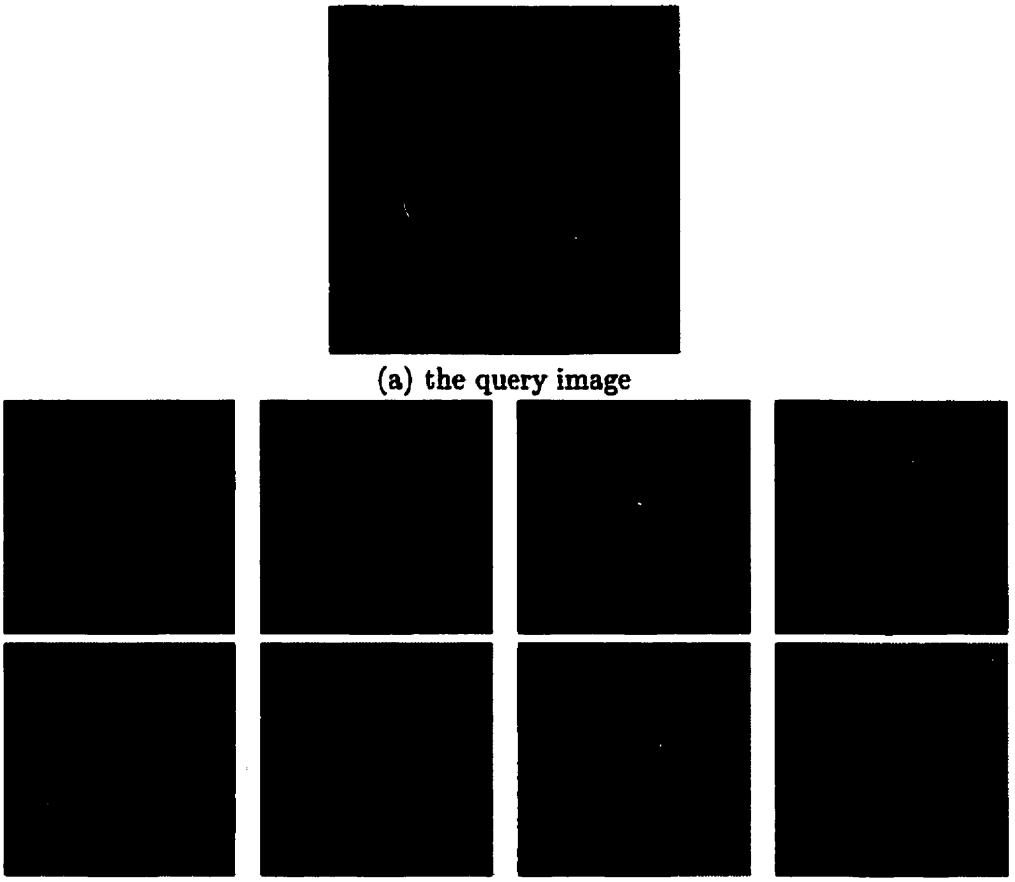
(vi) *If the database can be contextually divided into several classes, then follow the steps in Algorithm 4 to obtain a classification of the query.*

The first four steps of Algorithm 5 are done off-line. The reason principal components analysis is applied on a test set rather than on the complete database is that in general, a database of images is very large, which makes applying principal components analysis on the full database computationally expensive. Therefore, whenever the database is too large for practical purposes, we compute the principal directions of the data based on a small test set of images representative of the database. It is a hard task to find a good test set because it relies on the subjective judgment of whoever is choosing the test images. In general, every class has to have a few representative views in the test set, this becomes harder to determine if the classes are complicated.

In the next section, we compare the respective performances of the three different indexing approaches explained in Chapter 4, by applying our image retrieval system on a database of human faces.

2. Retrieving Pictures of Humans

Our first set of experiments consists of testing the ability of the system to cope with retrieving images of humans. We ran the tool on a database of human pictures taken under



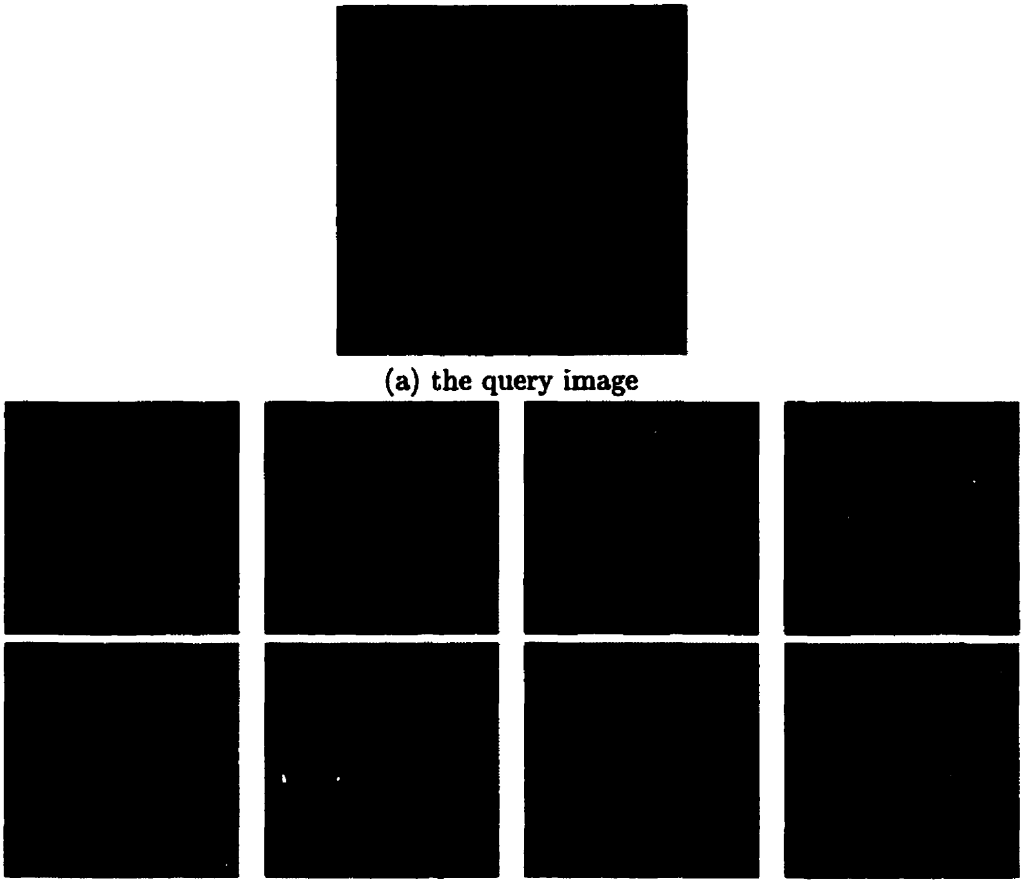
(a) the query image

(b) the system's response to the query

FIGURE 6.2. Image analysis on the absolute intensity values

different types of backgrounds. Each individual was imaged under different camera angles and different lighting conditions. Some examples of the images present in the database can be seen in Figure 6.1. The main purpose of the experiments is to test which method of indexing works better for image retrieval if the objects are embedded in simple backgrounds and complicated backgrounds. It is expected that very dissimilar images are used, the system should not have a difficulty separating the right responses from the wrong ones. On the other hand, when images of different humans are quite similar (which means they are taken under similar lighting conditions, camera positions, and background), then the system should be expected to have more trouble depicting the correct images.

In this set of experiments, the database of humans used is not very large (300 images). Therefore, we apply principal components analysis on the whole database to get the best



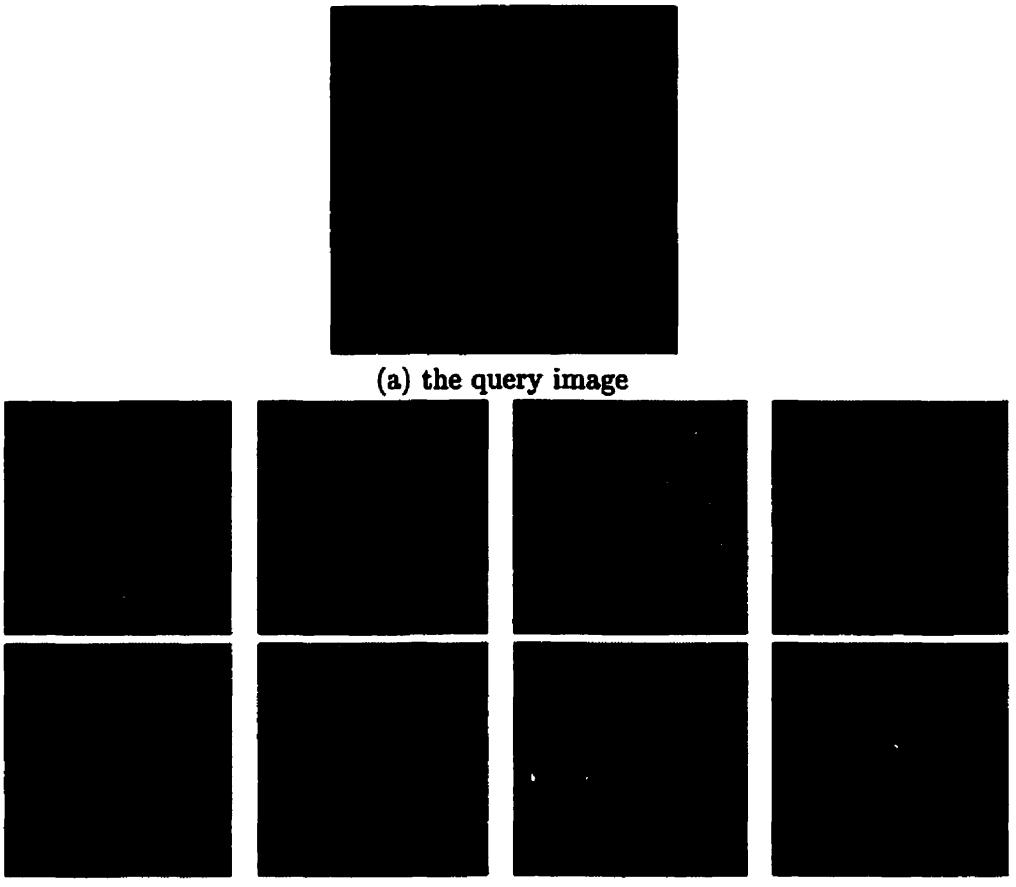
(a) the query image

(b) the system's response to the query

FIGURE 6.3. Image analysis by locating the zero crossings as an intermediate representation

possible representation of the data. We consider here that there is no need to separate the data into different classes. Consequently, no classification information is returned by the system. We perform experiments to test the image classifier on another (more elaborate) database later on.

2.1. Images with simple backgrounds. To demonstrate the superiority of using the zero crossings as an intermediate representation when we want to retrieve well framed images (in this case, well framed images are equivalent to images with simple backgrounds), we designed a set of experiments to compare the performance of the system when the representation of the data is based: (i) on the absolute image intensities, (ii) on the zero crossings and, (iii) on the magnitude of the Fourier Transform. A typical example of how the system functions can be seen in Figure 6.2. Here the query image is the large image in



(a) the query image

(b) the system's response to the query

FIGURE 6.4. Image analysis by calculating the magnitude of the Fourier Transform as an intermediate representation

Figure 6.2(a), and the system's response is in 6.2(b), where the images are ranked in the order of closeness to the query from left to right, top to bottom. Figure 6.2 is the system's response when the image analysis is performed on the level of the intensity image. We can notice that the closest images returned by the system include some correct choices. If we compare, however, to the images returned by the system when using the zero crossings as an intermediate representation (Figure 6.3), then we notice that more correct choices have been returned in the second case. In fact, in this case, all the relevant images have been returned as the closest images. The system discarded the differences in the lighting conditions and compared the images based solely on the shapes of the subjects. Figure 6.4 shows the system's response to the query when using the magnitude of the Fourier transform as an intermediate representation. Again, the response is not as good as in the

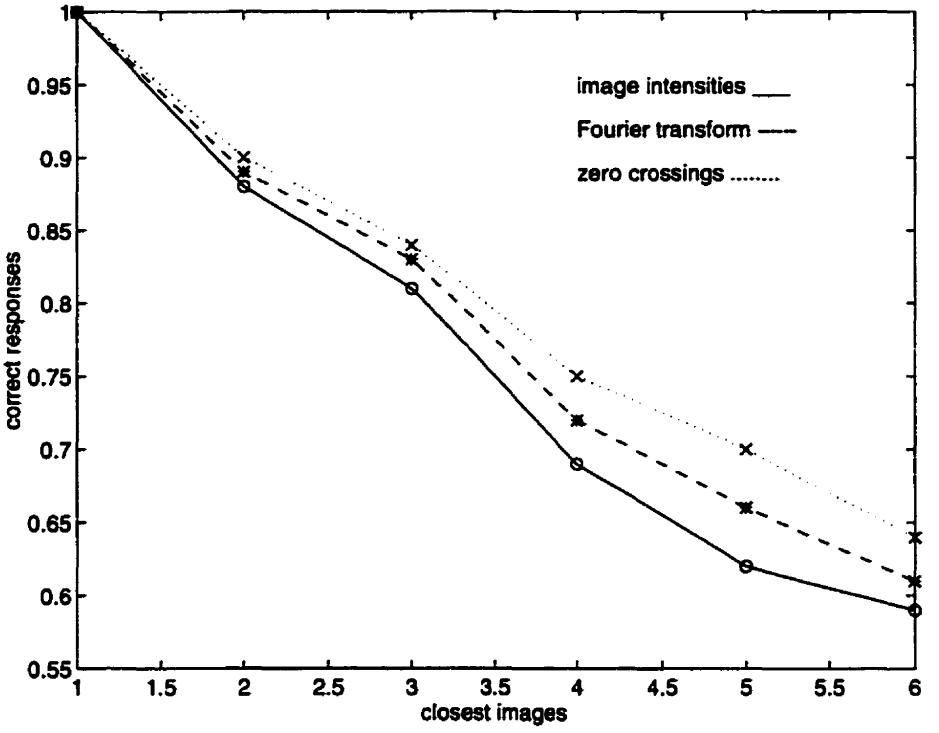
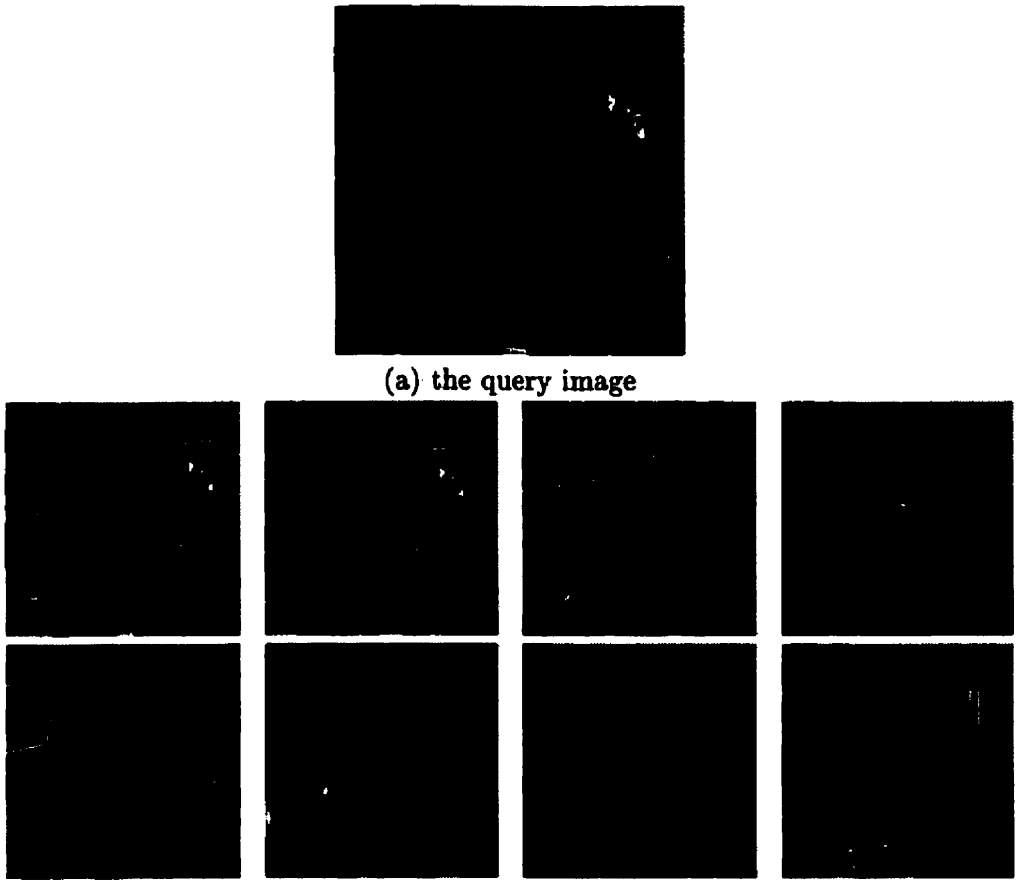


FIGURE 6.5. A comparative plot to illustrate the relative performances of the intermediate representations when retrieving images with simple backgrounds.

case of the zero crossings (Figure 6.3). We will see however, that for retrieving complex images, the Fourier transform proves to be a better feature for the purposes of returning the closest images.

To be able to see the advantage of using shape features to help retrieve well framed images, the plot in Figure 6.5 shows the correct return rate of correct responses of the system using all three representations. The x axis represents the closest images retrieved by the system, and the y axis represents the percentage of correct images retrieved. Since all the queries considered in this experiment are part of the test set, then the closest image to the query is always the image itself. This is why the three representations give 100% correct answers for the closest image. However, we can see that from the second closest images and on, the system performs better when the images are compared based on their shape characteristics.

2.2. Images with complex backgrounds. In Section 2.1, we studied the performance of the image retrieval system when the query image consists of a human in front of



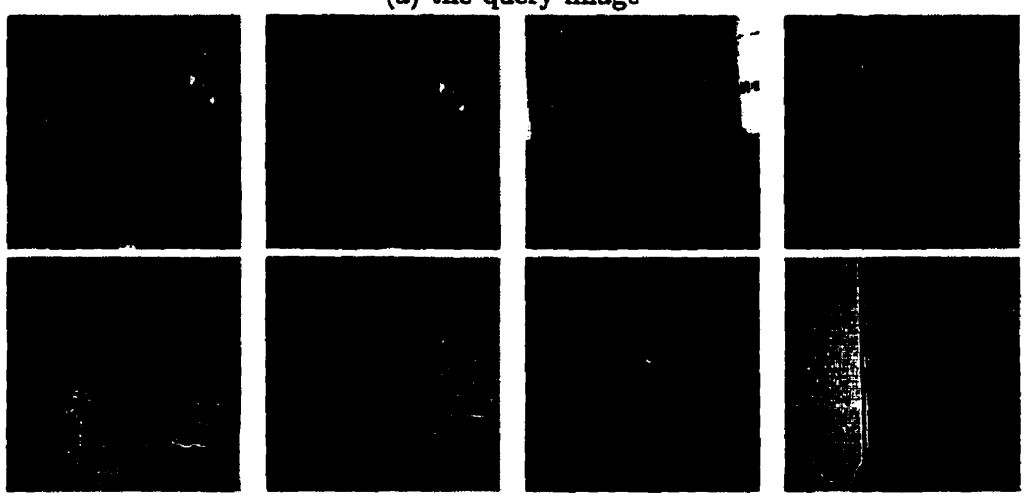
(a) the query image

(b) the system's response to the query

FIGURE 6.6. Image analysis on the absolute intensity values

a simple background. In this section, we inspect the system's performance when querying a scene including a human and a complicated background that can be composed of several objects of different sizes and shapes. We discussed in Chapter 4 that we consider it a very hard task to separate all the objects in a complex image, and therefore we analyze the image as a whole. In Figure 6.6(a), a query image is presented to the system, and the system's response to the query is shown in Figure 6.6(b). In this case, principal components analysis has been used directly on the absolute image intensities. We can notice that although the top three responses are correct (including the image itself), some of the returned images are very different from the query. Furthermore, some of the returned images have only simple backgrounds, which should not be acceptable. If we compare the images based on their zero crossings then the results are even less accurate (Figure 6.7). This is expected since the

(a) the query image

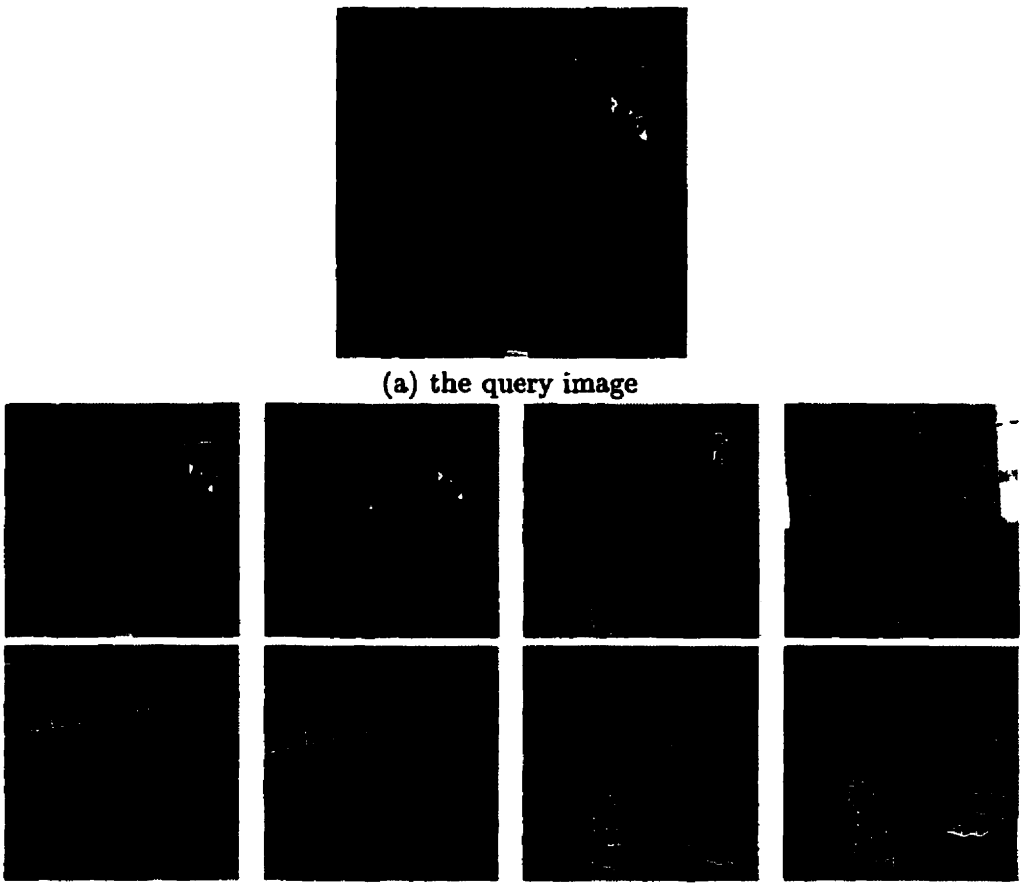


(b) the system's response to the query

FIGURE 6.7. Image analysis by calculating the zero crossings as an intermediate representation

zero crossings of complex objects become very noisy, and moreover, they are not related to the physical objects and are therefore a poor representation of such images.

Another problem arises when the images are complex. If the camera is allowed to alter its position in the scene, then some of the objects in the image are shifted and rotated by some amount. Furthermore, some objects disappear from the scene, and new objects appear. If the system compares images based on the pixel to pixel differences between them then the corresponding objects in both images do not match in the comparison. What is needed is a feature that is independent of the spatial position of the individual objects in the scene. This is a major advantage in favor of using the magnitude of the Fourier transform rather than the two other representations. It represents the scene in the frequency domain and not the spatial domain, and hence the importance of the frequencies of the objects in the



(a) the query image

(b) the system's response to the query

FIGURE 6.8. Image analysis by calculating the magnitude of the Fourier Transform as an intermediate representation

scene rather than their positions. The shapes of the objects are only represented implicitly. Figure 6.8 illustrates the fact that the retrieval of complex images improves considerably if the analysis is performed on the level of the magnitude of the Fourier transform of the images. All the relevant pictures have been returned successfully as the closest images, even though the pictures are taken under different camera positions.

It can be seen from the plot of Figure 6.9 that the comparison of the images based on the magnitude of their Fourier transform returns considerably higher rates of correct responses. If we compare the images in the frequency domain, then the view variation is allowed to be augmented while still having correct images retrieved. Another advantage is that this allows us to work with databases that include different types of images that are unconstrained as to their camera position.

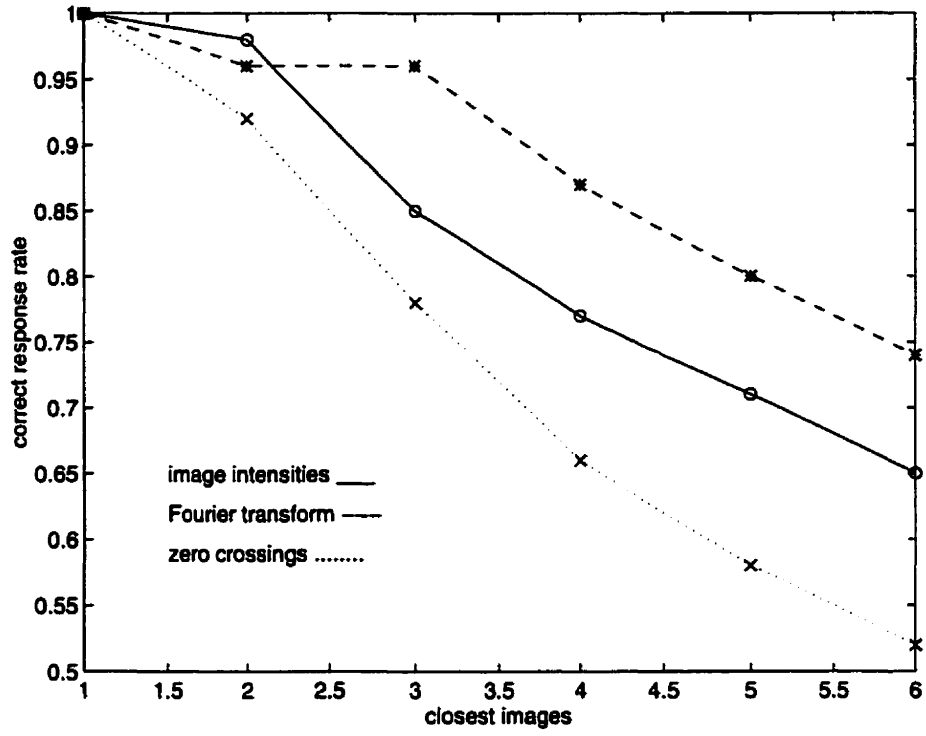
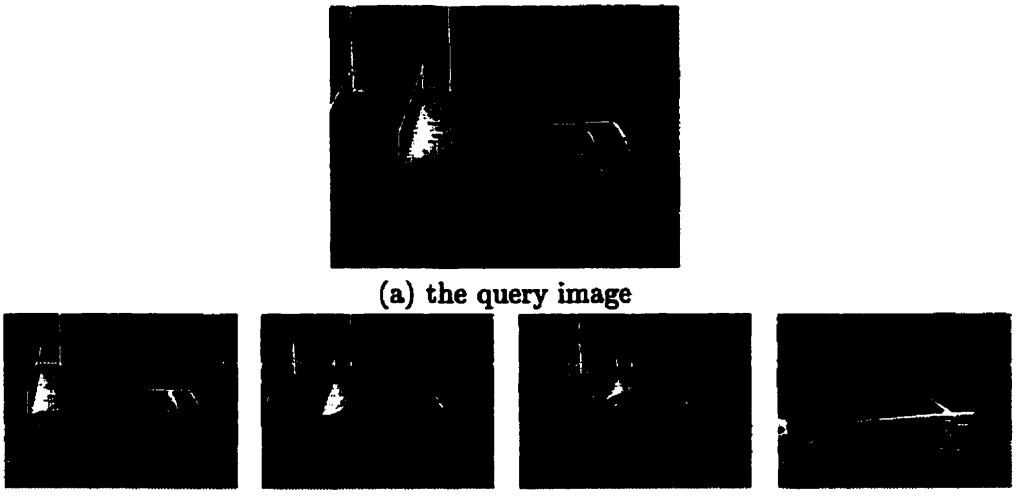


FIGURE 6.9. A comparative plot to illustrate the relative performances of the intermediate representations when retrieving images with complex backgrounds.

In the next section, we see how the system works in a more general environment when the database is large and when the images in it can be grouped in different classes.

3. Image Retrieval From a Large Database

The experiments performed in the previous section demonstrated that the performance of classical appearance-based methods can be considerably improved if the correct intermediate representations are used as relevant features to index the data. Since the database was not very large, the principal directions were computed using all the images. Furthermore, the example images were part of the database and therefore well represented in the test set. In this section, we investigate how the system performs if the database is large, and if only a small part of it is used as a test set. We ran two sets of experiments, the first one is to observe the system's response when the query image is not part of the test set, but at least one image of the same scene (but different camera angles) is represented in the test set. The other experiment involves trying to retrieve close images to a query image not belonging to the database at all. We ran the experiments on a database including 1500 images separated



(a) the query image

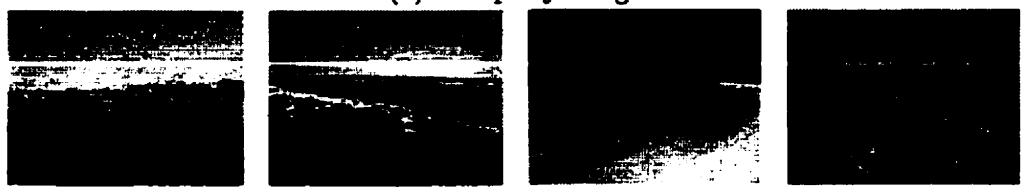
(b) the system's response to the query

FIGURE 6.10. Image retrieval of a car, comparison is base on the magnitude of the Fourier transform of the images

into four types of classes: human faces (the same faces as in Section 2), cards (the same as in Chapter 4), car scenes, and beach scenes. We applied principal components analysis on a test set of 300 representative images. For scenes that are imaged more than once, we made sure that at least one image representing the scene is present in the test set. Since in this case we can contextually form four classes of images (cars, cards, faces, beaches), then the system can also classify the query as belonging to one of the four classes. It should be noted that in this set of experiments, since we are dealing with complex scenes, the magnitude of the Fourier transform is always used as an intermediate representation of the images. A plot of the first two principal components of the test set was already shown if Figure 4.8. Although the first two components of beach images seem to be overlapping on all the other classes, there are 20 principal directions to take into account. Unfortunately not all of them can be shown in a plot. Notice that some classes are very concentrated in a small area (cards) while some others are very diffuse (beaches). This indicates that when the class is concentrated, only small changes in the imaging conditions happen. However, the more general the class is the more the conditions are expected to vary hence more scattering occurs in the parameters.

Figure 6.10(b) shows the system's response to the query in Figure 6.10(a). The query is not part of the test set. However, we notice that the first three responses are correct

(a) the query image



(b) the system's response to the query

FIGURE 6.11. Image retrieval of a beach scene

ones, and the fourth one also belongs to the same class. The classification module assigned the query image to be belonging to the cars class when using the k nearest neighbors rule. In these experiments we considered the 20 nearest neighbors. The results scored by each class were,

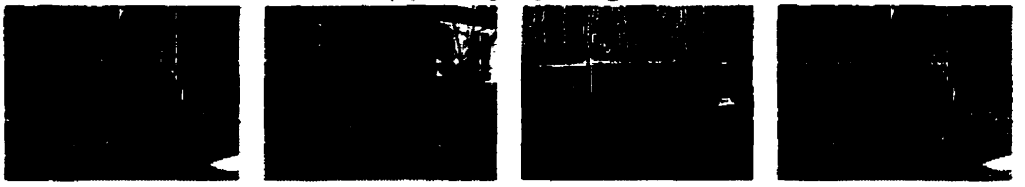
```
cars = 15
faces = 5
beaches = 0
cards = 0
```

On the other hand, the Bayesian classifier returned the following results,

```
belief in model:  beaches is 8.1874e-08
belief in model:  cars is 1
belief in model:  cards is 0
belief in model:  faces is 6.0601e-61
```

The next two examples test the system's response when a new unknown image is presented to the database. In Figure 6.11, the query image is a beach scene where the image has been selected from the Internet. As we can see from the system's response, the closest images returned are all beach scenes as well. The fact that frequency

(a) the query image



(b) the system's response to the query

FIGURE 6.12. Image retrieval of an unknown car scene

characteristics are used as relevant features to retrieve images improves the system's *generalization* performance considerably. The results of classification using knn were as follows,

```
cars = 0
faces = 0
beaches = 15
cards = 5
```

While the Bayesian classifier returned,

```
belief in model:  beaches is 1
belief in model:  cars is 6.52124e-110
belief in model:  cards is 0
belief in model:  faces is 0
```

The next example shown is that of a car scene not belonging to the database. The purpose of this experiment is to test if the system is capable of returning correct responses despite the fact that the distribution of beaches is much more diffuse than the distribution of cars (and all the other classes for that matter). Intuitively, it is expected that the system misclassifies some queries and assigns them to the beaches class because it covers a large area of the total distribution of classes (Figure 4.8). The results of the system to the query are shown in figure 6.12(b) where we can see that the closest image returned are all cars. Applying the knn rule to the query resulted in a correct classification: The results of the

		Query from database	Query from outside database
Rate of correct retrieval: closest images	1	94.1%	x
	2	79.4%	x
	3	74.3%	x
Rate of retrieved images from correct class	1	100%	94.7%
	5	96%	83.1%
	10	92%	80%
Rate of correct knn classification		92%	78.9%
Rate of correct Bayesian classification: Highest probability		84%	63.1%

TABLE 6.1. Performance Rates of the system

knn classification are,

```
cars = 18
faces = 1
beaches = 1
cards = 0
```

While the Bayesian classifier was not as confident in the classification. The result were as follows,

```
belief in model:  beaches is 0.477497
belief in model:  cars is 0.522503
belief in model:  cards is 0
belief in model:  faces is 7.24014e-90
```

From the large set of experiments run to test the system, we observed that generally the system does not perform as well when the image is completely new as opposed to an image being represented in the test set, which is to be expected. In Table 6.1, we summarize all the results obtained in this set of experiments. The first column (Query from database) shows the response of the system when the query belongs already to the database, but is not part of the training set of principal components analysis. The second column shows the response of the system to completely unknown queries. The first set of results (Rate of correct retrieval: closest images) investigates whether the system returns the correct images as the closest ones (in the table we consider the 1st, 2nd, and 3rd closest images). Here, our criteria for correct retrieval means that the system should return an image containing the exact same object as the query, i.e., same person, same car, etc. The second set of results

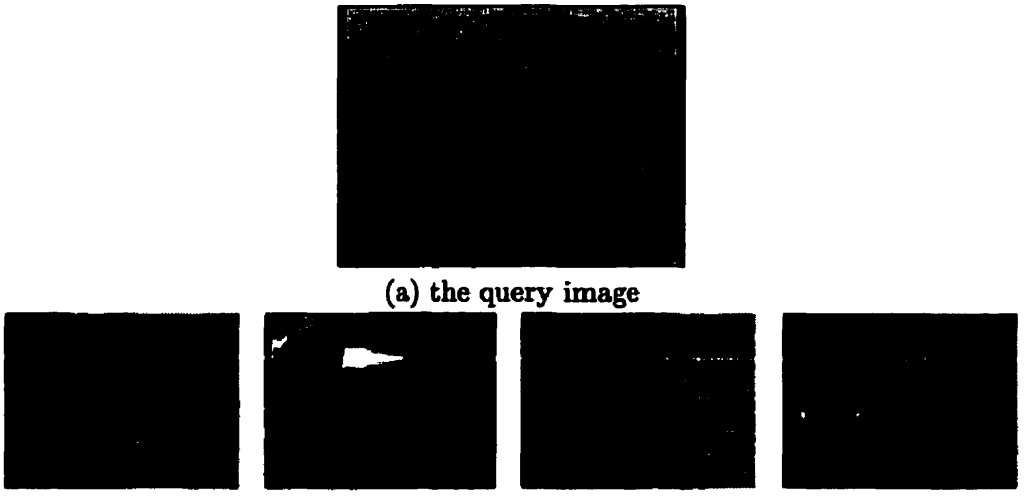
investigates whether the closest images returned belong to the correct class (in the table we consider the closest, the five closest, and the ten closest images). For example, we see that, if the query belongs to the database, then the five closest images belong the correct class 96% of the time. This figure becomes 83.1% if the query does not belong to the database. We see that when the query image belongs to the database (i.e., an image with the same scene has been represented in the test set), the rates of retrieval are higher than if the image is completely unknown. The system's performance is very promising however. The results in Table 6.1 show that among the closest images to the unknown query returned by the system, almost 95% belong to the correct class. Furthermore, among the 10th closest images, 8 on average belong to the correct class of the query. If the image belongs to the database then the results are even better. Notice that 94% of the time, the system returns the correct scene as the closest answer. As for the classification module, then knn classifies the query correctly 92% of the time if it belongs to the database, and 78% of the time if it is completely unknown.

Finally, we show an example where the system seems to fail. A completely successful system is expected to attribute a given query to the right class most of the time. We demonstrated above that our system has such capability. However, another important characteristic of a successful system is to be able to signal the presence of an outlier. In other words, if the query does not belong to any of the classes, then the system should detect that, and eliminate false positives. Figure 6.13 shows such a false positive and the system's response to it. Applying the knn classification rule returned the following results,

```
cars = 5
faces = 10
beaches = 4
cards = 1
```

And applying the Bayesian classification rule returned the following results,

```
belief in model:  beaches is 1
belief in model:  cars is 0
belief in model:  cards is 0
belief in model:  faces is 0
```



(a) the query image

(b) the system's response to the query

FIGURE 6.13. Image retrieval of a completely unrelated (outlier) image

We can see that in this case, the Bayesian classifier assigns a probability of 100% to the beaches class. Therefore, the system is failing in this case. If the system were to detect such false positives we would expect a more uniform distribution in all the classes, i.e., an uninformative classification. The knn methods returns more reliable results. We ran a set of experiments to verify the systems performance in the presence of outliers, and most of the responses were similar to the above example. Therefore, this is a case where it is dangerous to use the Bayesian classifier. Arbel and Ferrie (1996) Specifically deal with the issue of Bayesian analysis in detecting informative and uninformative viewpoints in 3-dimensional environments.

4. Summary

In this chapter, the algorithm necessary to implement the content-based image retrieval system was described. It consists of putting together all the theory and algorithms explained in earlier chapters. The experiments to test the performance of the system were illustrated in Sections 2 and 3. In Section 2 we have shown that the performance improves considerably when relevant features are used as intermediate representations. We chose the relevant features to be the zero crossings to index well framed images and the magnitude of the Fourier transform to index complex scenes. In Section 3, we tested if the system works well on a large database with general images of a complex nature. We have shown that the

system is able to generalize well to images not present in the test set, and also to completely new images (not present in the database). Furthermore, we have shown that the system's classifier returns correct classification in most cases (see the details in table 6.1). The system was implemented using the C programming language, and returns the closest images as well as the classification results in no more than 5 seconds for a 320 by 240 image.

CHAPTER 7

Conclusion

1. Review of the Thesis Objectives

The primary intent of this work is to build a fast, robust, and practical content-based image retrieval system that can handle the retrieval of real images. Image analysis is a very difficult problem, mainly because the problem of recognizing objects embedded in complicated backgrounds remains largely unsolved. For this reason, nearly all content-based image retrieval systems use constrained environments, or even images of very simple objects like toys. We feel it is necessary to build a system that can relax the constraints on the nature of the images as much as possible.

In this work appearance based methods have been used to retrieve images. Specifically, principal components analysis was applied as a tool to be able to represent the images by storing no more than a few coefficients. Images are then compared based on the small number of these coefficients. Traditionally principal components analysis is performed directly on the absolute image intensities. In Chapter 4, we explained in detail the reasons why this approach will fail to work well in a lot of environments. We summarize the reasons here again,

(i) The absolute intensities of the images are directly related to the lighting conditions at the moment the picture is taken. Therefore, any changes in these conditions will allow undesirable variations in the principal components of the data.

(ii) The process is very sensitive to translation and rotation of the objects with respect to the camera.

For these reasons, we need representations of the images that are invariant as to the external factors that might alter the appearance of images. We need to represent the images based on the inner properties of the objects. Therefore, we used intermediate representations relevant to the retrieval process to index the images and performed principal components analysis on these intermediate representations. If the database consists of well framed images then we use the zero crossings across four different scales to represent the images. On the other hand, if the images are complex, we use the magnitude of the Fourier Transform as an intermediate representation. The results in Chapter 6 demonstrated that the performance of the system improves and, more importantly, the system becomes well suited to deal with complex scenes. The final outcome was an image retrieval system largely insensitive to the lighting conditions, as well as translation and rotation.

In addition, the system we built does not merely stop at returning the closest images to a query. It further classifies this query using two different methods, the k nearest neighbors approach and the Bayesian classification approach. The results obtained are very promising as they allow this system to integrate the classification results as well as the retrieval results to further refine the query.

2. Limitations of the Approach

Our content-based image retrieval system suffers from the following limitations,

(i) The Marr/Hildreth operator used to extract the zero crossings is fast and can be adapted easily to represent multiple scales. However, it suffers from shortcomings that makes it inefficient for localizing edges in complex scenes containing several objects (recall figure 4.6). To be able to represent the objects' salient features in an accurate fashion, other, more complex operators are needed. Unfortunately, computational complexity limits what operators can be employed while still meeting real-time constraints.

(ii) The Bayesian classification approach generalizes well, i.e., it returns accurate results if a query not belonging to the database belongs to one of the classes. On the other hand, it performs poorly in the presence of outliers. More research has to be done in order to validate the assumptions made in the work about the nature of the different classes.

(iii) The problem of locating the individual objects in an image is not solved in this thesis. Appearance based methods treat the image as a full block, disregarding the relations between the objects in the image. Sometimes, however, a user is interested in retrieving complex images that include a specific object that occupies only a small part of the image. One way to address this issue is to build an interactive system that let the user "box" the object of interest and then build a query based on the boxed object. This raises another set of problems that slows down the process of retrieval considerably. Our system in its current form does not address this issue.

3. Future Directions

The system built in this work added one degree of "generality" to the type of images that can be retrieved in a system. The system can be further improved by including a feedback module which allows the user to refine the query as much as possible until the desired image is returned. Furthermore, better and more relevant features for retrieval would definitely accelerate the pace of the retrieval. It is clear that it is never enough to build a computer vision system that only analyzes the images intensities, but a successful system would consider the important information implicitly present in the image. The ultimate goal of this research remains to recognize the objects in the image and retrieve images with similar objects despite all the external factors, and despite the complexity of the scene it is exposed to. This is somewhat ambitious, however, and requires major advances in physiology and computer science simultaneously. For the moment, content-based image retrieval systems are not well advanced to include semantic contexts.

4. Summary

We repeat here the thesis goals stated in Chapter 1,

The primary goal of this research is to design a content-based image retrieval system that improves on the performance of classical appearance methods. The improvements in question are concerned with (i) Generality: the system has to be able to perform well in generalized environments like real scenes. (ii) Robustness: the system has to be less sensitive to factors like the absolute illumination level, camera angle, translation, and rotation.

We believe in the light of the theory described in Chapters 3, 4, and 5; and the experimental results obtained in Chapter 6, that the above objectives have been fulfilled. It was shown that using intermediate representations to represent images is a definite improvement on classical appearance based methods. Furthermore, the system becomes more "complete" when using the classifier that assigns the query to a class. Other definite advantages of this system are its extensibility and adaptability to integrate other indexing methods, depending on the context. Indeed this system is part of a digital library project that is still under construction.

REFERENCES

Arbel, T. and Ferrie, F. P., 1996, Informative views and sequential recognition, *in* B. Buxton and R. Cippola (eds), *Computer Vision, ECCV 96*, Springer - Verlag, Cambridge, UK, pp. 469–481.

Ardizzone, E. and LaCascia, M., 1997, "Automatic video database indexing and retrieval, " *MultToolApp*, Vol. 4, No. 1, pp. 29–56.

Ardizzone, E., LaCascia, M. and Molinelli, D., 1996, Motion and color based video indexing and retrieval, *Proc. of International Conference on Pattern Recognition, ICPR*, Wien, Austria.

Belhumeur, P., Hespanha, J. and Kriegman, D., 1996, Eigenfaces vs. fisherfaces: Recognition using class-specific linear projection, *ECCV96*, pp. 45–58.

Blum, H., 1973, "Biological shape and visual science, " *Theoretical Biology*, Vol. 38, pp. 205–287.

Brunelli, R. and Poggio, T., 1993, "Features vs. templates, " *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 15, No. 10.

Chandrasekaran, S., Manjunath, B. S., Wang, Y. F., Winkeler, J. and Zhang, H., 1996, An eigenspace update algorithm for image analysis, *Technical report*, Department of Electrical and Computer Engineering, University of California, Santa Barbara CA, 93106.

Chang, S. F., 1995, Compressed-domain techniques for image/video indexing and manipulation, *IEEE International Conference on Image Processing*, Washington D.C.

Chang, S. F. and Smith, J., 1995, Single color extraction and image query, *IEEE International Conference on Image Processing*, Washington D.C.

Chang, S. K., Shi, Q. Y. and Yan, C. W., 1987, "Iconic indexing by 2d strings, " *IEEE Transactions on Pattern Analysis and Machine Intelligence,* Vol. 3, pp. 413–427.

Chang, S. and Yang, C., 1983, "Picture information measures for similarity retrieval, " *CVGIP,* Vol. 23, pp. 366–375.

Costagliola, G., Tucci, M. and Chang, S. K., 1992, Representing and retrieving symbolic pictures by spatial relations, *Visual Database Systems II,* pp. 49–59.

Del-Bimbo, A. and Pala, P., 1997, "Visual image retrieval by elastic matching of user sketches, " *IEEE Transactions on Pattern Analysis and Machine Intelligence,* Vol. 19, No. 2, pp. 121–132.

Dubuc, B. and Zucker, S., 1995, Indexing visual representations through the complexity map, *ICCV95,* pp. 142–149.

Faloutsos, C., 1994, "Efficient and effective querying by image content, " *Journal of Intelligent Information Systems,* Vol. 3, No. 231.

Fleck, M., Forsyth, D. and Bregler, C., 1996, Finding naked people, *ECCV96,* pp. II:593–602.

Francos, J. M., Meiri, A. Z. and Porat, B., 1993, "A unified texture model based on a 2-d wold like decomposition, " *IEEE Transactions on Signal Processing,* pp. 2665–2678.

Fukunaga, K., 1972, *Introduction to Statistical Pattern Recognition,* New York, Academic.

Grosky, W. I. and Jiang, Z., 1994, "Hierarchical approach to feature indexing, " *Image and Vision Computing,* Vol. 5, pp. 275–283.

Grosky, W. I. and Lu, Y., 1986, "Iconic indexing using generalized pattern matching techniques, " *Computer Vision Graphics and Image Processing,* Vol. 35, pp. 383–403.

Grosky, W. I. and Mehrotra, R., 1990, "Index-based object recognition in pictorial data management, " *Computer Vision Graphics and Image Processing,* Vol. 52, pp. 416–436.

Hirata, K. and Kato, T.: 1992, Query by visual example, Lecture Notes in Advances in Database Technology.

Huttenlocher, D., Lilien, R. and Olson, C., 1996, Object recognition using subspace methods, *ECCV96,* pp. I:536–545.

Jacobs, C. E., Finkenstein, A. and Salesin, D. H., 1995, Fast multiresolution image querying, *Technical report*, Department of Computer Science and Engineering University of Washington.

Jungert, E., 1993, Qualitative spatial reasoning for determination of object relations using symbolic interval projections, *IEEE Symposium on Visual Languages*, pp. 83–87.

Kato, T., Turita, T., Otsu, N. and Hirata, K., 1992, A sketch retrieval method for full color image database, *International Conference on Pattern Recognition*, pp. 530–533.

Kelly, P. M. and Cannon, T. M., 1994, Candid: Comparison algorithm for navigating digital image databases, *Technical report*, Los Alamos National Laboratory.

Kirby, M. and Sirovich, L., 1990, "Application of the karhunen-loeve procedure for the characterization of human faces, " *IEEE Transactions on Pattern Analysis and Machine Intelligence,* Vol. 12, No. 1.

Kliot, M. and Rivlin, E., 1997, Shape retrieval in pictorial databases via geometric features, *Technical report*, Technion, Computer Science Department, IIT Haifa 32000, Israel.

LaCascia, M. and Ardizzone, E., 1996, Jacob: Just a content-based query system for video databases, *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, Atlanta.

Laws, K., 1980, Rapid texture identification, *Proc. SPIE, Image Processing for Missile Guidance*, Vol. 238, pp. 376–380.

Lebart, L., Morineau, A. and Warwick, K., 1984, *Multivariate Descriptive Statistical Analysis*, Wiley.

Lee, S. Y. and Hsu, F. J., 1991, "Picture algebra for spatial reasoning of iconic images represented in 2d strings, " *Pattern Recognition Letters,* Vol. 12, pp. 425–435.

Liu, F. and Picard, R. W., 1996, "Periodicity, directionality, and randomness: Wold features for image modeling and retrieval, " *IEEE Transactions on Pattern Analysis and Machine Intelligence,* Vol. 18, No. 7, pp. 722–733.

Manjunath, B. and Ma, W., 1996, "Texture features for browsing and retrieval of image data, " *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 18, No. 8, pp. 837–842.

Marr, D. and Hildreth, E., 1980, Theory of edge detection, *Proc. R. Soc. London*, pp. 187–217.

Mehrotra, R., Kung, F. K. and Grosky, W. I., 1990, "Industrial part recognition using a component-index, " *Image and Vision Computing,* Vol. 3, pp. 225–231.

Moghaddam, B., Pentland, A. and Nastar, C., 1996, Bayesian face recognition using deformable intensity surfaces, *CVPR96*, pp. 638–645.

Moghaddam, B., Wahid, W. and Pentland, A., 1998, Beyond eigenfaces: Probabilistic matching for face recognition, *The 3rd IEEE International Conference on Automatic Face & Gesture Recognition*, Nara, Japan.

Mundy, J. and Zisserman, A., 1993, *Geometric Invariance in Computer Vision*, The MIT Press, Cambridge Massachusetts 02142.

Nan, L., Dettmer, S. and Shah, M., 1997, Visually recognizing speech using eigen sequences, *MBR97*, p. Chapter 15.

Nayar, S. K., Murase, H. and Nene, S. A.: 1996, *Parametric Appearance Representation in Early Visual Learning*, Oxford University Press, chapter 6.

Niblack, W., Barber, R., Equitz, W., Flickner, M., Glasman, D., Petkovic, D. and Yanker, P., 1993, "The qbic project: Querying image by content using color, texture, and shape, " *SPIE,* Vol. 1908, pp. 173–187.

Ohba, K., Sato, Y. and Ikeuchi, K., 1998, Appearance based visual learning and object recognition with illumination invariance, *ACCV 98*, pp. 424–431.

Pearson, K., 1901, "On lines and planes of closest fit to systems of points in space., " *Phil. Mag. 6th Series,* .

Pentland, A., Picard, R. W. and Sclaroff, S., 1996, "Photobook: Content-based manipulation of image databases, " *International Journal of Computer Vision,* Vol. 18, No. 3, pp. 233–254.

Pernus, F., Leonardis, A. and Kovacic, S., 1994, "Two-dimensional object recognition using multiresolution non-information preserving shape features, " *Pattern Recognition Letters,* Vol. 11, pp. 1071–1079.

Picard, R. W. and Liu, F., 1994, A new wold ordering for image similarity, *Proc. IEEE Conf. on Acoustics, Speech and Signal Processing,* Vol. 5, pp. 129–132.

Pun, T. and Squire, D., 1996, "Statistical structuring of pictorial databases for content-based image retrieval systems, " *Pattern Recognition Letters,* Vol. 17, pp. 1299–1310.

Rao, A. R. and Lohse, G. L., 1993, Towards a texture naming system: Indentifying relevant dimensions of texture, *Proceedings of IEEE Conference Visualization*, San Jose, California, pp. 220–227.

Ravela, S. and Manmatha, R., 1998, Retrieving images by appearance, *ICCV98*, pp. 608–613.

Roadhouse, Z. F. and Kimia, B. B., 1997, A morphogenetic approach to generating shape queries, *Technical report*, Brown University, Providence RI 02912.

Roberts, L.: 1965, *Optical and Electro-Optical Information Processing*, MIT Press, Cambridge, Mass., chapter Machine Percepction of 3-Dimensional Solids.

Rodieck, R., 1965, "Quantitative analysis of cat retinal ganglion cell response to visual stimuli, " *Vis. Res.*, Vol. 5, pp. 583–601.

Sirovich, L. and Kirby, M., 1987, "Low-dimensional procedure for the characterization of human faces, " *Journal of the Optical Society of America*, Vol. 4, No. 3, pp. 519–524.

Smith, J. and Chang, S. F., 1995, Automated image retrieval using color and texture, *Technical report*, Columbia University.

Smith, J. R. and Chang, S. F., 1996, Tools and techniques for color image retrieval, *Proc. of IS&T SPIE, Storage and Retrieval Image and Video Database IV*, San Jose, CA.

Stollnitz, E. J., DeRose, T. D. and Salesin, D. H., 1995, "Wavelets for computer graphics: A primer, " *IEEE Computer Graphics and Applications*, Vol. 15, No. 3, pp. 76–84.

Swain, M. and Ballard, D., 1995, "Color indexing, " *International Journal of Computer Vision*, Vol. 7, No. 11.

Swets, D. L. and Weng, J., 1996, "Using discriminant eigenfeatures for image retrieval, " *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 18, No. 8, pp. 831–836.

Tamura, H., Mori, S. and Yamawaki, T., 1978, "Textural features corresponding to visual perception, " *IEEE Transactions on Systems, Man and Cybernectics*, Vol. 8, No. 6.

Toussaint, G. T., 1974, "Bibliography on estimation of misclassification, " *IEEE Transactions on Information Theory*, Vol. IT-20, No. 4, pp. 472–478.

Turk, M. and Pentland, A., 1991, "Eigenfaces for recognition, " *Journal of Cognitive Neuroscience*, Vol. 3, No. 1, pp. 71–86.

Watanabe, S., 1965, Karhunen-loève expansion and factor analysis. theoretical remarks and applications, *Proc. 4th Prague Conf. Inform. Theory.*

Whaite, P. and Ferrie, F., 1993, Model building and autonomous exploration, *Intelligent Robots and Computer Vision XII: Active Vision and 3d Methods*, SPIE, SPIE, Boston, Massachussets, pp. 73–85.

Wilks, S. S., 1963, *Mathematical Statistics*, Wiley, New York.

Wu, T. C. and Chang, C. C., 1994, "Application of geometric hashing to iconic database retrieval, " *Pattern Recognition Letters*, Vol. 15, pp. 871–876.

Zhang, H. J., Low, C. Y., Smoliar, S. W. and Wu, J. H., 1995, Video parsing retrieval and browsing: and integrated and content based solution, *Proc. ACM Multimedia'95*, pp. 15–24.

Document Log:

Manuscript Version 0

Typeset by $\mathcal{A}\mathcal{M}\mathcal{S}$-LaTeX — 1 February 1999

FADI BEYROUTI

CENTER FOR INTELLIGENT MACHINES, MCGILL UNIVERSITY, 3480 UNIVERSITY ST., MONTRÉAL (QUÉBEC) H3A 2A7, CANADA, *Tel.* : (514) 398-2185

E-mail address: polaris@cim.mcgill.ca

Typeset by $\mathcal{A}\mathcal{M}\mathcal{S}$-LaTeX