NEAR EAST UNIVERSITY

Faculty of Engineering

Department of Computer Engineering

FACE RECOGNITION TECHNIQUES

Graduation Project
COM-400

Student: Usman Sultan (981306)

Supervisor: Assoc. Prof. Dr Adnan Khashman

Nicosia - 2002

TABLE OF CONTENTS

ACKNOWLEGMENT

All my thanks goes to Near East University and to my supervisor Pro.Dr Adnan Khashman, for his effort less support, guidance and assistance through out the project.

A gigabyte of thanks go from me to all of my friends for encouragement, motivating and helping me whenever I need them.

Words cannot express my thanks to my parents who had provided me an ocean of support, and backing me up, keep rounding put my perfect life that they began so long ago and I'm most fortunate son on Earth.

ABSTRACT

In recent years considerable progress has been made in the area of face recognition. Through the development of techniques like Eigenfaces and Local feature Analysis computers can now outperform humans in many face recognition tasks, particularly those in which large databases of faces must be searched. Given a digital image of a person's face, face recognition software matches it against a database of other images. If any of the stored images matches closely enough, the system reports the sighting to its owner, and so the efficient way to perform this is to use an Artificial Intelligence system.

The main aim of this project is to discuss the development of the face recognition system. For this purpose the state of art of the face recognition is given. However, many approaches to face recognition involving many applications and there eignfaces to solve the face recognition system problems is given too. For example, the project contain a description of a face recognition system by dynamic link matching which shows a good capability to solve the invariant object recognition problem.

A better approach is to recognize the face in unsupervised manner using neural network architecture. We collect typical faces from each individual, project them onto the eigenspace or local feature analysis and neural network learns how to classify them with the new face descriptor as input.

INTRODUCTION

The project has presented an approach to the detection and identification of human faces and describe a working, near real time face recognition system which tracks a subject's head and then recognizes the person by comparing characteristics of the face to those of known individuals.

Face recognition in general and the recognition of moving people in natural scenes in particular, require a set of visual tasks to be performed robustly.

These includes:

- *Acquisition:* the detection and tracking of face-like image patches in a dynamic scene.
- *Normalization:* the segmentation, alignment and normalization of the face images.
- *Recognition:* the representation and modeling of face images as identities, and the association of novel face images with known models.

The project describes the ways that perform these tasks, and it also gives some results and researches for Face Recognition by several methods. The project consists of introduction, 4 chapters and conclusion.

Chapter one presents the history of face recognition and why it is important, with some technologies that are used now a days.

Chapter two describes the Techniques and calculations used in face recognition with Eigenfaces and Local Feature Analysis.

Chapter three present a Neural Network approach to face recognition with some experimental results.

Chapter Four describes the face recognition application.

Finally conclusion presents the obtained important results and contributions in the project.

The objectives of this project are:

1. Describe the important of face recognition and show where we can use it.
2. Maintain the techniques and calculations for detection and recognition by given results from eigenfaces and local feature analysis.
3. Maintain a face recognition by neural network approach and see if it is has the capability to extract the images from convolutional network.
4. Show the approaches to face recognition and discuss its applications.

CHAPTER ONE
INTRODUCTION TO FACE RECOGNITION

1.1 Overview

In recent years considerable progress has been made in the areas of face recognition. Through the work of people like Alex Pentland computers can now perform outperform humans in many face recognition tasks, particularly those in which large databases of faces must be searched. A system with the ability to detect and recognize faces in a crowd has many potential applications including crowd and airport surveillance, private security and improved human computer interaction.

1.2 History of Face Recognition

The subject of face recognition is as old as computer vision, both because of the practical importance of the topic and theoretical interest from cognitive scientists. Despite the fact that other methods of identification (such as fingerprints, or iris scans) can be more accurate, face recognition has always remains a major focus of research because of its non-invasive nature and because it is people's primary method of person identification.

Perhaps the most famous early example of a face recognition system is due to Kohonen [1], who demonstrated that a simple neural net could perform face recognition for aligned and normalized face images. The type of network he employed computed a face description by approximating the eigenvectors of the face image's autocorrelation matrix; these eigenvectors are now known as `eigenfaces.' Destiny is not a matter of chance; it's a matter of choice.

This method functions by projecting a face onto a multi-dimensional feature space that spans the gamut of human faces. A set of basis images is extracted from the database presented to the system by Eigenvalue-Eigenvector decomposition. Any face in the feature space is then characterized by a weight vector obtained by projecting it onto the set of basis images. When a new face is presented to the system, its weight vector is calculated and compared with those of the faces in the database. The nearest neighbor to this weight vector, computed using the Euclidean norm, is determined. If this distance is below a certain threshold (found by experimentation) the input face is

adjudged as that face corresponding to the closest weight vector. Otherwise, the input pattern is adjudged as not belonging to the database.

Kohonen's system was not a practical success, however, because of the need for precise alignment and normalization. In following years many researchers tried face recognition schemes based on edges, inter-feature distances, and other neural net approaches. While several were successful on small databases of aligned images, none successfully addressed the more realistic problem of large databases where the location and scale of the face is unknown.

Kirby and Sirovich (1989) [6] later introduced an algebraic manipulation which made it easy to directly calculate the eigenfaces, and showed that fewer than 100 were required to accurately code carefully aligned and normalized face images. Turk and Pentland (1991) [1] then demonstrated that the residual error when coding using the eigenfaces could be used both to detect faces in cluttered natural imagery, and to determine the precise location and scale of faces in an image. They then demonstrated that by coupling this method for detecting and localizing faces with the eigenface recognition method, one could achieve reliable, real-time recognition of faces in a minimally constrained environment. This demonstration that simple, real-time pattern recognition techniques could be combined to create a useful system sparked an explosion of interest in the topic of face recognition.

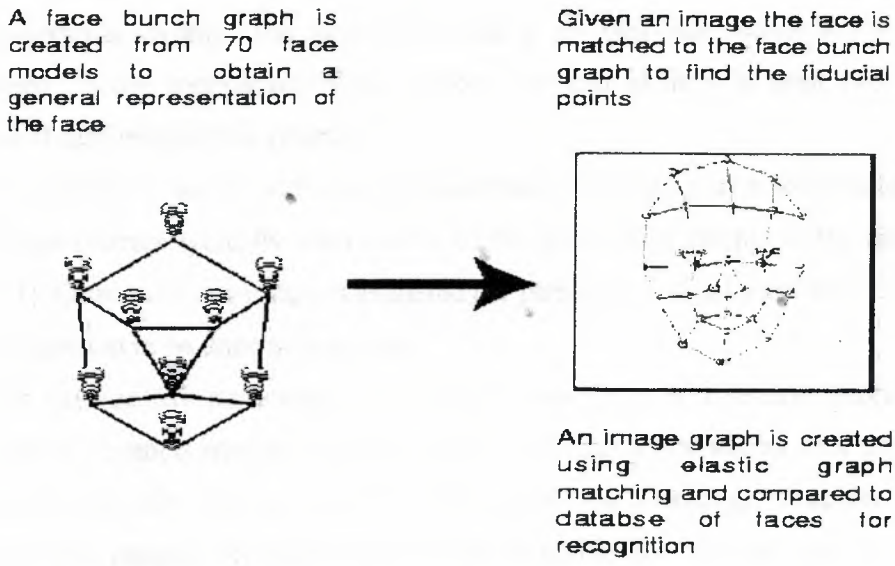


Figure1.1 Face recognition using elastic graph matching.

1.3 Face Recognition

Smart environments, wearable computers, and ubiquitous computing in general are thought to be the coming `fourth generation' of computing and information technology . Because these devices will be everywhere -- clothes, home, car, and office, their economic impact and cultural significance are expected to dwarf previous generations of computing. At a minimum, they are among the most exciting and economically important research areas in information technology and computer science.

However, before this new generation of computing can be widely deployed we must invent new methods of interaction that don't require a keyboard or mouse -- there will be too many small computers to instruct them all individually. To win wide consumer acceptance such interactions must be friendly and personalized (no one likes being treated like just another cog in a machine!), which implies that next-generation interfaces will be aware of the people in their immediate environment and at a minimum know who they are.

The requirement for reliable personal identification in computerized access control has resulted in an increased interest in biometrics. Biometrics being investigated includes fingerprints , speech, signature dynamics, and face recognition. Sales of identity verification products exceed $100 million.

Face recognition has the benefit of being a passive, non-intrusive system for verifying personal identity. The techniques used in the best face recognition systems may depend on the application of the system. We can identify at least two broad categories of face recognition systems:

1. We can find a person within a large database of faces (e.g. in a police database). These systems typically return a list of the most likely people in the database [41]. Often only one image is available per person. It is usually not necessary for recognition to be done in real-time.
2. We can identify particular people in real-time (e.g. in a security monitoring system, location tracking system, etc.), or we can allow access to a group of people and deny access to all others (e.g. access to a building, computer, etc.) . Multiple images per person are often available for training and real-time recognition is required.

1.4 Why Face Recognition

Given the requirement for determining people's identity, the obvious question is what technology is best suited to supply this information? There are many different identification technologies available, many of which have been in widespread commercial use for years. The most common person verification and identification methods today are Password/PIN (Personal Identification Number) systems, and Token systems (such as your driver's license). Because such systems have trouble with forgery, theft, and lapses in users' memory, there has developed considerable interest in biometric identification systems, which use pattern recognition techniques to identify people using their physiological characteristics. Fingerprints are a classic example of a biometric; newer technologies include retina and iris recognition.

While appropriate for bank transactions and entry into secure areas, such technologies have the disadvantage that they are intrusive both physically and socially. They require the user to position their body relative to the sensor, and then pause for seconds to `declare' themselves. This `pause and declare' interaction is unlikely to change because of the fine-grain spatial sensing required. Moreover, there is an `oracle-like' aspect to the interaction: since people can't recognize other people using this sort of data, these types of identification do not have a place in normal human interactions and social structures.

While the `pause and present' interaction and the oracle-like perception are useful in high-security applications (they make the systems look more accurate), they are exactly the opposite of what is required when building a store that recognizes its best customers, or an information kiosk that remembers you, or a house that knows the people who live there. Face recognition from video and voice recognition have a natural place in these next-generation smart environments -- they are unobtrusive (able to recognize at a distance without requiring a `pause and present' interaction), are usually passive (do not require generating special electro-magnetic illumination), do not restrict user movement, and are now both low-power and inexpensive. Perhaps most important, however, is that humans identify other people by their face and voice, therefore are likely to be comfortable with systems that use face and voice recognition.

1.5 Mathematical Framework

Twenty years ago the problem of face recognition was considered among the hardest in Artificial Intelligence (AI) and computer vision. Surprisingly, however, over the last decade there have been a series of successes that have made the general person identification enterprise appear not only technically feasible but also economically practical.

The apparent tractability of face recognition problem combined with the dream of smart environments has produced a huge surge of interest from both funding agencies and from researchers themselves. It has also spawned several thriving commercial enterprises. There are now several companies that sell commercial face recognition software that is capable of high-accuracy recognition with databases of over 1,000 people.

These early successes came from the combination of well-established pattern recognition techniques with a fairly sophisticated understanding of the image generation process. In addition, researchers realized that they could capitalize on regularities that are peculiar to people, for instance, that human skin colors lie on a one-dimensional manifold (with color variation primarily due to melanin concentration), and that human facial geometry is limited and essentially 2-D when people are looking toward the camera. Today, researchers are working on relaxing some of the constraints of existing face recognition algorithms to achieve robustness under changes in lighting, aging, rotation-in-depth, expression and appearance (beard, glasses, makeup) -- problems that have partial solution at the moment.

1.5.1 The Typical Representational Framework

The dominant representational approach that has evolved is descriptive rather than generative. Training images are used to characterize the range of 2-D appearances of objects to be recognized. Although initially very simple modeling methods were used, the dominant method of characterizing appearance has fairly quickly become estimation of the probability density function (PDF) of the image data for the target class.

For instance, given several examples of a target class Ω in a low-dimensional representation of the image data, it is straightforward to model the probability distribution function $P(x|\Omega)$ of its image-level features **x** as a simple parametric

function (e.g., a mixture of Gaussians), thus obtaining a low-dimensional, computationally efficient appearance model for the target class.

Once the PDF of the target class has been learned, we can use Bayes' rule to perform maximum a posteriori (MAP) detection and recognition. The result is typically a very simple, neural-net-like representation of the target class's appearance, which can be used to detect occurrences of the class, to compactly describe its appearance, and to efficiently compare different examples from the same class. Indeed, this representational framework is so efficient that some of the current face recognition methods can process video data at 30 frames per second, and several can compare an incoming face to a database of thousands of people in under one second -- and all on a standard PC!

1.6 Dealing with the Curse of Dimensionality

To obtain an `appearance-based' representation, one must first transform the image into a low-dimensional coordinate system that preserves the general perceptual quality of the target object's image. This transformation is necessary in order to address the `curse of dimensionality'. The raw image data has so many degrees of freedom that it would require millions of examples to learn the range of appearances directly.

Typical methods of dimensionality reduction include Karhunen-Loève transform (KLT) (also called Principal Components Analysis (PCA)) or the Ritz approximation (also called `example-based representation'). Other dimensionality reduction methods are sometimes also employed, including sparse filter representations (e.g., Gabor Jets, Wavelet transforms), feature histograms, independent components analysis, and so forth.

These methods have in common the property that they allow efficient characterization of a low-dimensional subspace with the overall space of raw image measurements. Once a low-dimensional representation of the target class (face, eye, hand, etc.) has been obtained, standard statistical parameter estimation methods can be used to learn the range of appearance that the target exhibits in the new, low-dimensional coordinate system. Because of the lower dimensionality, relatively few examples are required to obtain a useful estimate of either the PDF or the inter-class discriminant function.

An important variation on this methodology is discriminative models, which attempt to model the differences between classes rather than the classes themselves.

Such models can often be learned more efficiently and accurately than when directly modeling the PDF. A simple linear example of such a difference feature is the Fisher discriminant. One can also employ discriminant classifiers such as Support Vector Machines (SVM), which attempt to maximize the margin between classes.

1.7 Current State of the Art

By 1993 there were several algorithms claiming to have accurate performance in minimally constrained environments. To better understand the potential of these algorithms, DARPA and the Army Research Laboratory established the FERET program with the goals of both evaluating their performance and encouraging advances in the technology [42].

At the time of this writing, there are three algorithms that have demonstrated the highest level of recognition accuracy on large databases (1196 people or more) under double-blind testing conditions. These are the algorithms from University of Southern California (USC) [43], University of Maryland (UMD) [44], and the MIT Media Lab [45]. All of these are participants in the FERET program. Only two of these algorithms, from USC and MIT, are capable of both minimally constrained detection and recognition; the others require approximate eye locations to operate. A fourth algorithm that was an early contender, developed at Rockefeller University [46], dropped from testing to form a commercial enterprise. The MIT and USC algorithms have also become the basis for commercial systems.

The MIT, Rockefeller, and UMD algorithms all use a version of the eigenface transforms followed by discriminative modeling. The UMD algorithm uses a linear discriminant, while the MIT system, seen in Figure 1.2, employs a quadratic discriminant. The Rockefeller system, seen in Figure 1.3, uses a sparse version of the eigenface transform, followed by a discriminative neural network. The USC system, seen in Figure 1, in contrast, uses a very different approach. It begins by computing Gabor `jets' from the image, and then does a `flexible template' comparison between image descriptions using a graph-matching algorithm.

The FERET database testing employs faces with variable position, scale, and lighting in a manner consistent with mugs hot or driver's license photography. On databases of fewer than 200 people and images taken under similar conditions, all four algorithms produce nearly perfect performance. Interestingly, even simple correlation

matching can sometimes achieve similar accuracy for databases of only 200 people [42]. This is strong evidence that any new algorithm should be tested with at databases of at least 200 individuals, and should achieve performance over 95% on mugshot-like images before it can be considered potentially competitive.

In the larger FERET testing (with 1166 or more images), the performance of the four algorithms is similar enough that it is difficult or impossible to make meaningful distinctions between them (especially if adjustments for date of testing, etc., are made). On frontal images taken the same day, typical first-choice recognition performance is 95% accuracy. For images taken with a different camera and lighting, typical performance drops to 80% accuracy. And for images taken one year later, the typical accuracy is approximately 50%. Note that even 50% accuracy is 600 times chance performance.

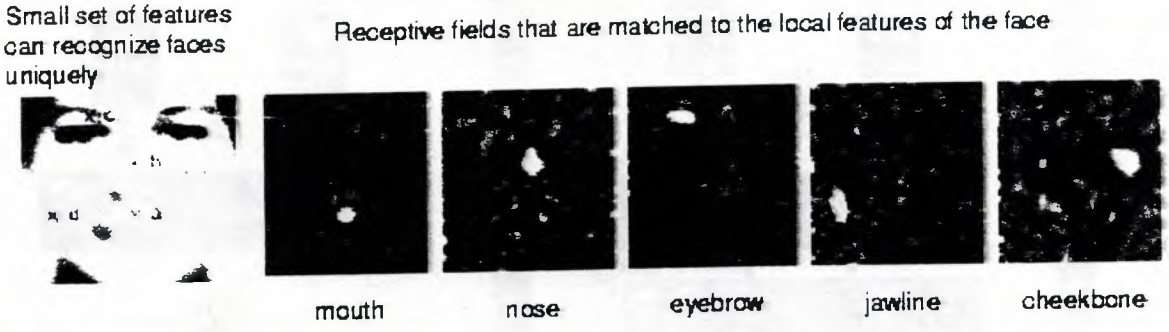


Figure1.2 Face recognition using Local Feature Analysis

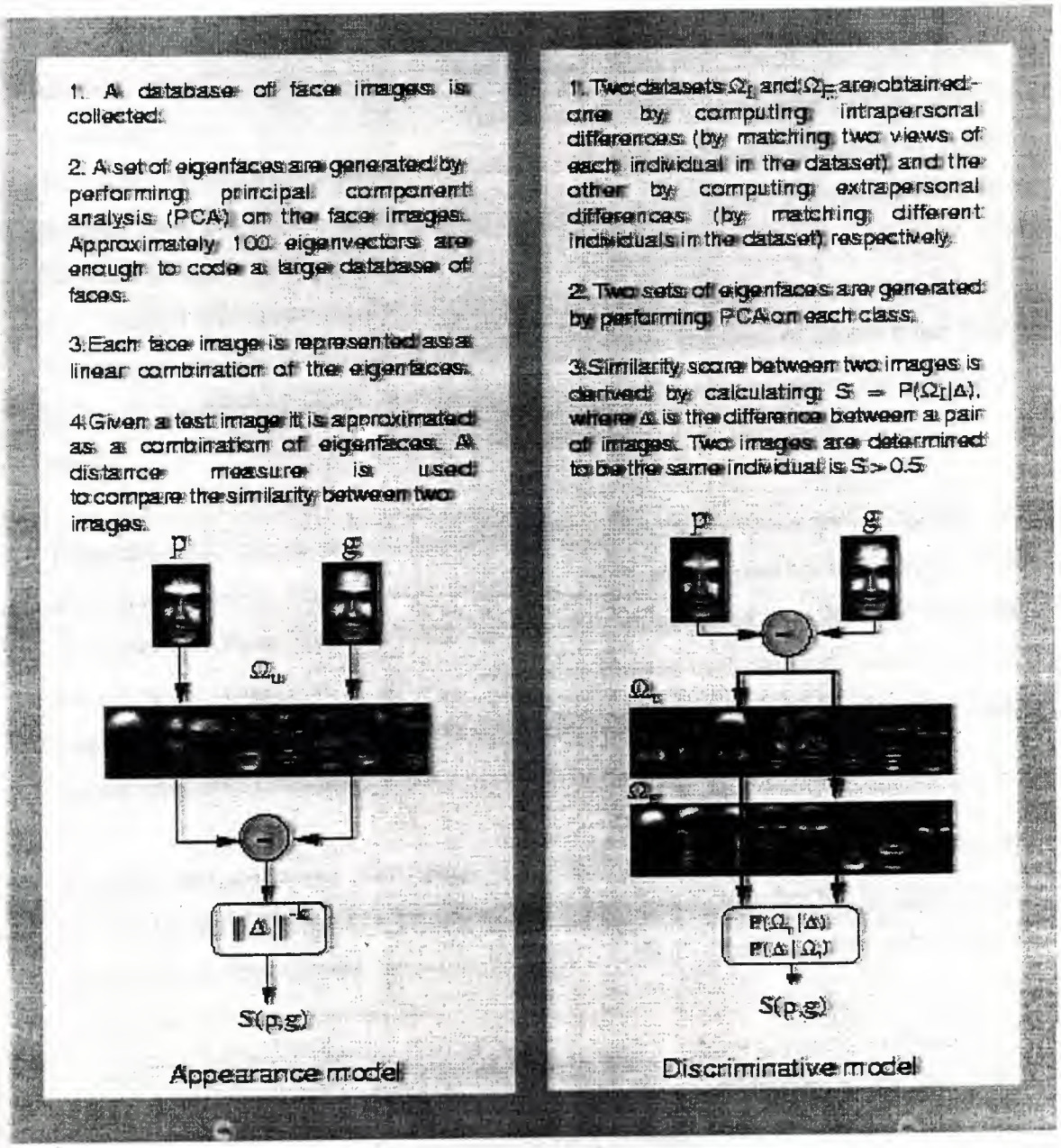


Figure1.3 Face recognition using Eigenfaces

1.8 Commercial Systems and Applications

Currently, several face-recognition products are commercially available. Algorithms developed by the top contenders of the FERET competition are the basis of some of the available systems; others were developed outside of the FERET testing

framework. While it is extremely difficult to judge, three systems -- Visionics, Viisage, and Miros -- seem to be the current market leaders in face recognition.

Visionics FaceIt face recognition software is based on the Local Feature Analysis algorithm developed at Rockefeller University. FaceIt is now being incorporated into a Close Circuit Television (CCTV) anti-crime system called 'Mandrake' in United Kingdom. This system searches for known criminals in video acquired from 144 CCTV camera locations. When a match occurs a security officer in the control room is notified.

FaceIt will automatically detect human presence, locate and track faces, extract face images, perform identification by matching against a database of people it has seen before or pre-enrolled users. The technology is typically used in one of the following ways:

Identification (one-to-many searching): To determine someone's identity in identification mode, FaceIt quickly computes the degree of overlap between the live face print and those associated with known individuals stored in a database of facial images. It can return a list of possible individuals ordered in diminishing score (yielding resembling images), or it can simply return the identity of the subject (the top match) and an associated confidence level

Verification (one-to-one matching): In verification mode, the face print can be stored on a smart card or in a computerized record. FaceIt simply matches the live print to the stored one--if the confidence score exceeds a certain threshold, then the match is successful and identity is verified.

Monitoring: Using face detection and face recognition capabilities, FaceIt can follow the presence and position of a person in the field of view.

Surveillance: FaceIt can find human faces anywhere in the field of view and at any distance, and it can continuously track them and crop them out of the scene, matching the face against a watch list. Totally hands off, continuously and in real-time.

Limited size storage devices: FaceIt can compress a face print into 84 bytes for use in smart cards, bar codes and other limited size storage devices.

Visage, another leading face-recognition company, and uses the eigenface-based recognition algorithm developed at the MIT Media Laboratory. Their system is used in conjunction with identification cards (e.g., driver's licenses and similar government ID cards) in many US states and several developing nations.

Miros uses neural network technology for their TrueFace face recognition software. TrueFace is for checking cash system, and has been deployed at casinos and similar sites in many US states.

1.9 Novel Applications of Face Recognition Systems

Face recognition systems are no longer limited to identity verification and surveillance tasks. Growing numbers of applications are starting to use face-recognition as the initial step towards interpreting human actions, intention, and behavior, as a central part of next-generation smart environments. Many of the actions and behaviors humans' display can only be interpreted if you also know the person's identity, and the identity of the people around them. Examples are a valued repeat customer entering a store, or behavior monitoring in an eldercare or childcare facility, and command-and-control interfaces in a military or industrial setting. In each of these applications identity information is crucial in order to provide machines with the background knowledge needed to interpret measurements and observations of human actions.

1.10 Face Recognition for Smart Environments

Researchers today are actively building smart environments (i.e. visual, audio, and hap tic interfaces to environments such as rooms, cars, and office desks) . In these applications a key goal is usually to give machines perceptual abilities that allow them to function naturally with people -- to recognize the people and remember their preferences and peculiarities, to know what they are looking at, and to interpret their words, gestures, and unconscious cues such as vocal prosody and body language. Researchers are using these perceptually aware devices to explore applications in health care, entertainment, and collaborative work.

Recognition of facial expression is an important example of how face recognition interacts with other smart environment capabilities. It is important that a smart system knows whether the user looks impatient because information is being presented too slowly, or confused because it is going too fast -- facial expressions

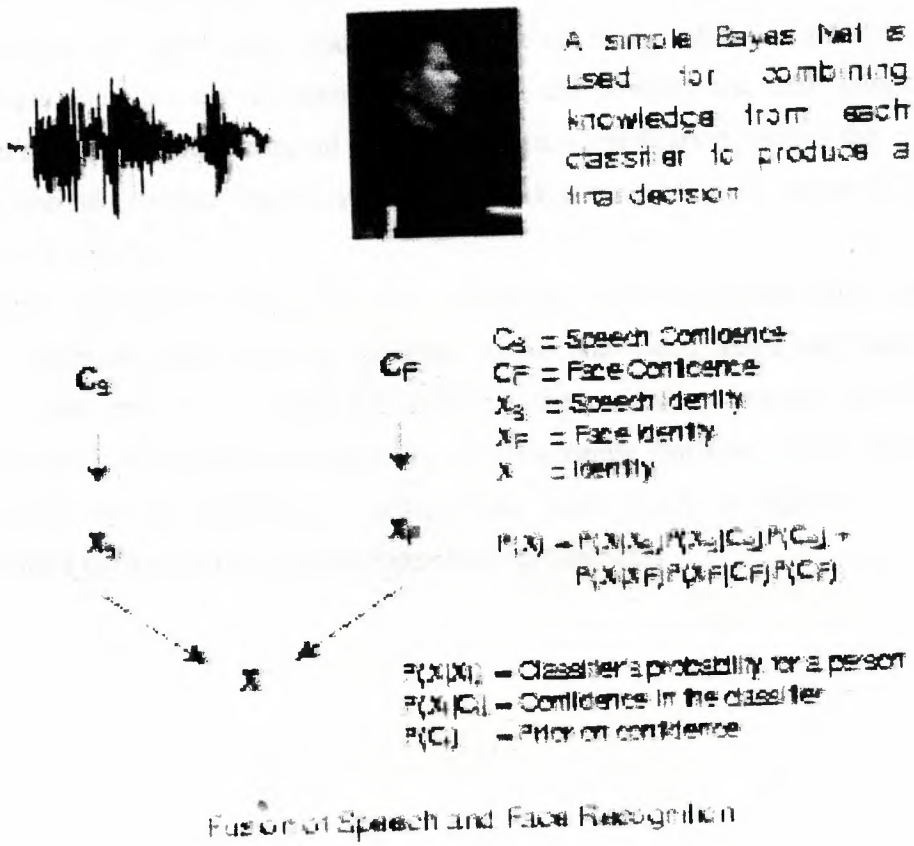
capability that is critical for a variety of human-machine interfaces, with the hope of creating a person-independent expression recognition capability. While there are indeed similarities in expressions across cultures and across people, for anything but the grossest facial expressions analysis must be done relative to the person's normal facial rest state -- something that definitely isn't the same across people. Consequently, facial expression research has so far been limited to recognition of a few discrete expressions rather than addressing the entire spectrum of expression along with its subtle variations. Before one can achieve a really useful expression analysis capability one must be able to first recognize the person, and tune the parameters of the system to that specific person.

1.11 Wearable Recognition Systems

When we build computers, cameras, microphones and other sensors into a person's clothes, the computer's view moves from a passive third-person to an active first-person vantage point (Figure1.4) . These wearable devices are able to adapt to a specific user and to be more intimately and actively involved in the user's activities. The field of wearable computing is rapidly expanding, and just recently became a full-fledged Technical Committee within the IEEE Computer Society. Consequently, we can expect to see rapidly growing interest in the largely unexplored area of first-person image interpretation.

Figure1.4 Wearable face recognition system.

Face recognition is an integral part of wearable systems like memory aides, remembrance agents, and context-aware systems. Thus there is a need for many future recognition systems to be integrated with the user's clothing and accessories. For instance, if you build a camera into your eyeglasses, then face recognition software can help you remember the name of the person you are looking at by whispering their name in your ear. Such devices are beginning to be tested by the US Army for use by border guards in Bosnia, and by researchers at the University of Rochester's Center for Future Health for use by Alzheimer's patients.



Fusion of Speech and Face Recognition

Figure1.5 Multi-modal person recognition system

1.12 Summary

Face recognition systems used today work very well under constrained conditions, although all systems work much better with frontal mug-shot images and constant lighting. All current face recognition algorithms fail under the vastly varying conditions under which humans need to and are able to identify other people. Next generation person recognition systems will need to recognize people in real-time and in much less constrained situations.

We believe that identification systems that are robust in natural environments, in the presence of noise and illumination changes, cannot rely on a single modality, so that fusion with other modalities is essential (Figure1.4). Technology used in smart environments has to be unobtrusive and allow users to act freely. Wearable systems in particular require their sensing technology to be small, low powered and easily integral with the user's clothing. Considering all the requirements, identification systems that use face recognition and speaker identification seem to us to have the most potential for wide-spread application.

Cameras and microphones today are very small, light-weight and have been successfully integrated with wearable systems. Audio and video based recognition systems have the critical advantage that they use the modalities humans use for recognition. Finally, researchers are beginning to demonstrate that unobtrusive audio-and-video based person identification systems can achieve high recognition rates without requiring the user to be in highly controlled environments.

CHAPTER TWO

TECHNIQUES USED IN FACE RECOGNITION

2.1 Overview

This chapter describes a face detection approach via learning eigenfaces and local features analysis. The first part of the chapter describes about eigenfaces. Eigenfaces are an excellent basis for face recognition system, providing high recognition accuracy and moderate insensitivity to lighting variations. The second part of the chapter details about local feature analysis. The key idea is that local features, being manifested by a collection of pixels in a local region, are learnt from the training set instead of arbitrarily defined.

2.2 Introduction

Face recognition is a well-studied problem in computer vision. Its current applications include security (ATM's, computer logins, and secure building entrances), criminal photo "mug-shot" databases, and human-computer interfaces.)

One of the more successful techniques of face recognition is Local feature analysis, and specifically eigenfaces [1, 2, 3]. Infrared images (or thermo grams) represent the heat patterns emitted from an object. Since the vein and tissue structure of a face is unique (like a fingerprint), the infrared image should also be unique (given enough resolution, you can actually see the surface veins of the face). At the resolutions used in this study (160 by 120), we only see the averaged result of the vein patterns and tissue structure. However, even at this low resolution, infrared images give good results for face

Recognition the only known usage of infrared images for face recognition is by company Technology Recognition Systems [4]. Their system does not use principle component analysis, but rather simple histogram and template techniques. They do claim to have a very accurate system (which is even capable of telling identical twins apart), but they unfortunately have no published results, which we could use for comparison.

To determine someone's identity

- The computer takes an image of that person and

- Determines the pattern of points that make that individual differ most from other people. Then the system starts creating patterns
- Either randomly or based on the average eigenface.
- The computer constructs a face image and compares it with the target face to be identified.
- New patterns are created until a facial image that matches with the target can be constructed. When a match is found, the computer looks in its database for a matching pattern of a real person.

2.3 Eigen Faces

Developing a computational model of face recognition is quite difficult, because faces are complex, multidimensional, and meaningful visual stimuli. They are a natural class of objects, and stand in stark contrast to sine wave gratings, the "blocks world", and other artificial stimuli used in human and computer vision research [5]. Thus unlike most early Visual functions, for which we may construct detailed models or retinal or striate activity, face recognition is a very high level task for which computational approaches can currently only suggest broad constraints on the corresponding neural activity.

This chapter is focusing towards developing a sort of early, protective pattern recognition capability that does not depend on having full three-dimensional models or detailed geometry. The aim is to develop a computational model of face recognition, which is fast, reasonably simple, and accurate in constrained environments such as an office or household.

Although face recognition is a high level visual problem, there is a quite a bit of structure imposed on the task. We take advantages of some of this structure by proposing a scheme for recognition which is based on an information theory approach, seeking to encode the most relevant information in a group of faces which will best distinguish them form one another. The approach transform face images into a small set of characteristic feature images, called "eigenfaces", which are the principal components of the initial training set of face images. Recognition is performed by projecting a new image into the subspace spanned by the eigen face ("face space") and then classifying the face by comparing its position in face space with the positions of known individuals.

Automatically learning and later recognizing new faces is practical with this framework. Recognition under reasonably varying conditions is achieved by training on a limited number of characteristic views (e.g. a "straight on" view, a 45° view, and a profile view). The approach has advantage over other face recognition schemes in its speed and simplicity, learning capacity, and relative insensitivity to small or gradual changes in the face.

2.3.1 Eigen Faces for Recognition

Much of the previous work on automated face recognition has ignored the issue of just what aspects of the face stimulus are important for identification, assuming that predefined measurements were relevant and sufficient. This suggested to us that an information theory approach of coding and decoding face images may give insight into the information content of face images, emphasizing the significant local ad global "features". Such features may or may not be directly related to our intuitive notion of face features such as the eyes, nose, lips and hair.

In the language of information theory, to extract the relevant information in a face Image, encode it as efficiently as possible, and compare one face encoding with a database of models encoded similarly. A similar approach to extract the information contained in an image of a face is to somehow capture the variation in a collection in an image of a face is images, independent of any judgment of features, and use this information to encode and compare individual face images.

In mathematical terms, to find the principal components of the distributions of faces, or the eigenvectors of the covariance matrix of the set of face images. These eigenvectors can be thought of as a set of features, which together characterize for variation between face images. Each image location contributes more or less to each eigenvector, so that we can display the eigenvector as sort of ghostly face, which we call an eigenface. Some of these faces are shown in figure (2.1).

Each face image in the training set can be represented exactly in terms of a linear combination of the eigenfaces. The number of possible eigenfaces is equal to the number of face images in the training set. However the faces can also be approximated using only the "best" eigenfaces- those that have the largest eigenvalues, and which therefore account for the most variance within the set of face images. The primary reason for using fewer eigenfaces is computational efficiency. The best M' eigenfaces

span a M'-dimensional subspace-"face space"-of all possible images. As sinusoids of varying frequency and phase are the basis functions of a fourier decomposition (and are in fact eigenfunctions of linear systems), the eigenfaces are the basis vectors of the eigenface decomposition.

The idea of using eigenfaces was motivated by a technique developed by Sirovich and Kirby [6] for efficiently representing pictures of faces using principal components analysis. They argued that a collection of face images can be approximately reconstructed by storing a small collection of weights for each face and a small set of standard pictures.

It occurred that if a multitude of face images can be reconstructed by weighted sums of a small collection of characteristic images, then an efficient way to learn and recognize faces might be to build the characteristic features from known face images and to recognize particular faces by comparing the feature weights needed to (approximately) reconstruct them with the weights associated with the known individuals.

The following steps summarize the recognition process.

1. Initialization: Acquire the training set of face images and calculate the eigenfaces, which define the face space.
2. When a new face image is encountered, calculate a set of weights based on the input image and the M eigenfaces by projecting the input image onto each of the eigenfaces.
3. Determine if the image is a face at all (whether known or unknown) by checking to see if the image is sufficiently close to "face space".
4. If it is a face, classify the weight pattern as either a known person or as unknown.
5. (Optional) If the same unknown face is seen several times, calculate its characteristic weight pattern and incorporate into the known faces (i.e. Learn to recognize it).

A general idea for face recognition is to extract the relevant information in a face image, encode it as efficiently as possible, and compare one face encoding with a database of similarly encoded images. In the eigenfaces technique, we have training and test set of images, and we compute the eigenvectors of the covariance matrix of the training set of Images. These eigenvectors can be thought of as a set of features that together characterize the variation between face images. When the eigenvectors are

displayed, they look like a ghostly face, and are termed eigenfaces. The eigenfaces can be linearly combined to reconstruct any image in the training set exactly. In addition, if we use a subset of the eigenfaces, which have the highest corresponding eigenvalue (which accounts for the most variance in the set of training images), we can reconstruct (approximately) any training image with a great deal of accuracy. This idea leads not only to computational efficiency (by reducing the number of eigenfaces we have to work with), but it also makes the recognition more general and robust.

Storage: The face recognition system that we worked on builds a set of orthonormal basis vectors based on the Karhunen-Loève procedure for generation of orthonormal vectors. Using the best (highest eigenvalue, or most face-like) of these basis vectors, which we call eigenfaces, we map images to "face-space". Using this representation, we can store each image as only a vector of N numbers where N is the number of eigenfaces. This results in huge storage savings as both the MIT group and it was concluded that 50 eigenfaces forms a fairly comprehensive set of eigenvectors for characterizing faces. Thus, 80K images are stored as 50 numbers.

Matching: Using this stored representation of the images, when presented with a new image we can map it to face-space as well and quickly see which vector it most corresponds to or whether it corresponds to any of the vectors at all. By seeing if it corresponds to any of the stored vectors better than a certain threshold we can determine who the person is. If the image does not correspond to any of the stored vectors we conclude that we do not know (or fail to recognize) the person. Also, by taking the image to face-space and then back to image space we can see how good the reconstruction is and by this determine whether the image is in fact a face or not.

Reconstruction: The ability to reconstruct the images from our stored vectors gives us both the ability for face-checking, the determination of whether the image is a face, and also image compression since the 50 values and corresponding set of eigenfaces are enough to reconstruct most any face.

Applications:The face recognition system has a number of uses, which cause apprehension.

- System (key) access based on face/voice recognition

- Tracking people either spatially with a large network of cameras or temporally by monitoring the same camera over time. (London is currently attempting to do both)
- Locating of people in large images

The face-key and tracking system both are based on matching faces to other faces stored in a database, while the people locating system is based on 'face-ness'. For the location task, an image is scanned and each region is converted to face-space and back to check to see if it is a face. This scanning task can be used to find everything from license plates (using eigen-license-plates) to Waldo (using eigen-Waldos).

2.4 Constructing Eigenfaces

This procedure is a form of principle component analysis. First, the conceptually simple version:

- Collect a bunch (call this number N) of images and crop them so that the eyes and chin are included, but not much else.
- Convert each image (which is x by y pixels) into a vector of length xy.
- Pack these vectors as columns of a large matrix.
- Add xy - N zero vectors so that the matrix will be square (xy by xy).
- Compute the eigenvectors of this matrix and sort them according to the corresponding eigenvalues. These vectors are your eigenfaces. Keep the M eigenfaces with the largest associated eigenvalues.

Unfortunately, this procedure relies on computing eigenvectors of an extremely large matrix. Our images are 250x300, so the matrix would be 75000 by 75000 (5.6 billion entries!). On the bright side, there's another way (the Karhunen-Loève expansion):

Collect the N images, crop them, and convert them to vectors. Compute the N by N outer product matrix (call it L) of these images. The entry L_{ij} of this matrix is the inner product of image vectors number i and j. As a result, L will be symmetric and non-negative. Compute the eigenvectors of L. This will produce N - 1 vectors of length N. Use the eigenvectors of L to construct the eigenfaces as follows: for each eigenvector v, multiply each element with the corresponding image and add those up. The result is an eigenface, one of the basis elements for face space. Use the same sorting and selecting process described above to cut it down to M eigenfaces.

Transforming an Image to Face Space

This procedure is exactly what had expected for the usual Hilbert space change of basis.Take inner products between the image and each of the eigenfaces and pack these into a vector of length M.

The Inverse Face Space Transforms

- Multiply each of the elements of the face space vector with the corresponding eigenfaces, and add up the result.
- Transform it to face space.
- Record the resulting vector (which will be much smaller than the image).

Recognizing a known Face

- Transform the image presented for recognition to face space.
- Take inner products with each of the learned face space vectors (think Cauchy-Schwartz).
- If one of these inner products is above the threshold, take the largest one and return that its owner also owns the new face.
- Otherwise, it's an unknown face. Optionally add it to the collection of known faces as "Unknown Person #1".

Evaluating "Face-ness" of an Image

If unsure whether an image is a face or not, transform it to face space, then do the inverse transform to get a new image back. Use mean-squared-error to compare these two images. If the error is too high, it isn't a face at all. Note that this process does not rely on knowing any faces, just having a set of eigenfaces.

The Face recognition is an important task for computer vision systems, and it remains an open and active area of research. To implement and experiment with a promising approach to this problem: eigenfaces.

Think of an image of a face (grayscale) as an N by N matrix - this can be rearranged to form a vector of length N^2, which is just a point in R^{N2}. That's a very high dimensional space, but picture of faces only occupy a relatively small part of it. By doing some straightforward principal component analysis (discussion of this part to be

added later), a smaller set of M "eigenfaces" can be chosen (M is a design parameter), and the faces to be remembered can be expressed as a linear combination of these M eigenfaces. In other words the faces have been transformed from the image domain (where they take up lots of storage space: ~N^2) to the face domain (where they require much less: ~M). This will necessarily be an approximation, but it turns out to be a pretty good one in practice. To recognize a new image of a face, simply transform it to the face domain and take an inner product with each of the known faces to see if we have a match. Faces presented for recognition will be scaled, rotated, and shifted the same as they were first seen. However, changes in lighting, facial expression, etc are fair game. No hats or heavy make-up or anything silly like that.

The general implementation plan is:

1. Take some pictures with a handy digital camera (got one).
2. Scale, rotate, crop, etc the images by hand using image-editing software.
3. Construct the eigenfaces.
4. Compute and store face domain versions of each person's face.
5. Grab and fix up some more images - some of known people and some of unknown people.
6. Test the recognizer!

Most likely the actual implementation stuff will be done in some combination of Python and Matlab, unless we get crazy and decide to try this in real-time (it should be feasible - these are efficient algorithms), in which case, some C will be necessary. Procedure could also serve for searching for faces in a larger image.

2.5 Computing Eigen Faces

Consider a black and white image of size N×N $I(x,y)$.$I(x,y)$ is simply a matrix of 8-bit values with each element representing the intensity at that particular pixel. These images can be thought of as a vector of dimension N^2, or a point in N^2 dimensional space. A set of images therefore corresponds to a set of points in this high dimensional space. Since facial images are similar in structure, these points will not be randomly distributed, and therefore can be described by a lower dimensional subspace. Principal component analysis gives the basis vectors for this subspace (which is called the "facespace"). Each basis vector is of length N^2, and is the eigenvector of the covariance matrix corresponding to the original face images.

So 128*128 pixel image can be represented as a point in a 16,384 dimensional space facial images in general will occupy only a small sub-region of this high dimensional "image space" and thus are not optimally represented in this coordinate system.

The eigenfaces technique works on the assumption that facial images from a simply connected sub-region of this image space. Thus it is possible, through principal components analysis (PCA) to work out an optimal co-ordinate system for facial images. Here an optimal coordinate system refers to one along which the variance of the facial images is maximized.

This becomes obvious when we consider the underlying ideas of PCA. PCA aims to catch the total variation in a set of facial images, and to explain this variation by as few variables as possible. This not only decreases the computational complexity of face recognition, but also scales each variable according to its relative importance in explaining the observation.

Let $T_1, T_2, ..., T_M$ be the training set of face images. The average face is defined by

$$\Phi = \frac{1}{M}\sum_{i=1}^{M} T_i. \tag{2.1}$$

Each face differs from the average face by the vector $\phi = T_i - \Psi$. The covariance matrix

$$C = \frac{1}{M}\sum_{i=1}^{M} \Phi_i \Phi_i^T \tag{2.2}$$

has a dimension of N^2 by N^2. Determining the eigenvectors of C for typical sizes of N is intractable. We are determining the eigenvectors by solving a M by M matrix instead.

2.5.1 Classification

The eigenfaces span an M' dimensional subspace of the original N^2 image space. The M' significant eigenvectors are chosen as those with the largest corresponding eigenvalues. A test face image I' is projected into face space by the following operation $w_i = u_i^T(T - \Psi)$, for $i = 1, ..., M'$, where u_i are the eigenvectors for C. The weights w_i form a vector $\Omega^T = [w_1, w_2, ..., w_{M'}]$, which describes the contribution of each eigenface in representing the input face image. This vector can then be used to fit the test image to a predefined face class. A simple technique is to use the

Euclidian distance $\varepsilon_i = \|\Omega - \Omega_i\|$, where Ω_i describes the ith face class. A test image is in class i when $\varepsilon_i < \theta_i$, where θ_i is a user specified threshold.

Given a vector C the eigenvectors u and eigen values λ of C satisfy

$$Cu = \lambda u \tag{2.3}$$

The eigenvectors are orthogonal and normalized hence

$$u_i^T u_j = \begin{cases} 1 & i = j \\ 0 & i \neq j \end{cases} \tag{2.4}$$

Let T_k represent the column vector of face k obtained through lexographical ordering of $I_k(x, y)$. Now let us define ϕ_k as the mean normalized column vector for face k. this means that

$$\phi_k = \Gamma_k - \psi \tag{2.5}$$

Where

$$\psi = \frac{1}{M}\sum_{k=1}^{M}\Gamma_k \tag{2.6}$$

Now let C be the covariance matrix of the mean normalized faces.

$$C = \frac{1}{M}\sum_{k=1}^{M}\phi_k\phi_k^T \tag{2.7}$$

M is the number of facial images in our representation set. These facial images help to characterize the sub-space formed by faces within image space. This sub-space will henceforth be referred to as 'face-space'.

$$\begin{aligned} Cu_i &= \lambda_i u_i \\ u_i^T C u_i &= u_i^T \lambda_i u_i \\ &= \lambda_i u_i^T u_i \end{aligned} \tag{2.8}$$

Now since $u_l^T u_j = 1$

$$
\begin{aligned}
u_i^T C u_i &= \lambda_i \\
\lambda_i &= \frac{1}{M} u_i^T \sum_{k=1}^{M} \phi_k \phi_k^T u_i \\
&= \frac{1}{M} \sum_{k=1}^{M} u_i^T \phi_k \phi_k^T u_i \\
&= \frac{1}{M} \sum_{k=1}^{M} (u_i \phi_k^T)^T (u_i \phi_k^T) \\
&= \frac{1}{M} \sum_{k=1}^{M} (u_i \phi_k^T)^2 \\
&= \frac{1}{M} \sum_{k=1}^{M} (u_i \Gamma_k^T - mean(u_i \Gamma_k^T))^2 \\
&= \frac{1}{M} \sum_{k=1}^{M} var(u_i \Gamma_k^T)
\end{aligned}
\tag{2.9}
$$

Thus eigenvalue i represent the variance of the representation facial image set along the axis describes by eigenvector i.

So by selecting the eigenvector with the largest eigenvalues as our basis, we are selecting the dimensions, which can express the greatest variance in facial images or the dominant modes of face-space. Using this coordinate system a face can be accurately reconstructed with as few a 6 coordinates. This means that a face, which previously took 16,384 bytes to represent in image space, now requires only 6 bytes. Once again, this reduction in dimensionality makes the problem of face recognition much simpler since we concern only with the attributes of the face.

2.5.2 Equipment and Procedures

The infrared camera used is a Cincinnati Electronics IRC-160. This camera has a resolution of 160 by 120 pixels, 12 bit planes, and is sensitive over the 2.5 to 5.5 nm infrared range. The IRC has a digital interface, which was connected to a Sparc 20 with an EDT SDV board.

The subjects were at a fixed distance from the camera (6.5'); a 50 mm lens was used on the IRC. Three views points were used in this study (frontal, 45°, profile). In addition, for each view the subject made two expressions (normal and smile). For each expression, two images were captured 4 seconds apart. Thus a total of 12 images were captured for each subject, giving a grand total of 288 images in the database.

The faces were aligned (by hand) to improve the performance of the eigenface technique. Specifically, frontal images were aligned using the midpoint of the subject's eyes; 45° 45_ images were aligned on the subject's right eye; and profile images were aligned using the tip of the subject's nose. The images were not scaled in any way. The subjects did not have glasses on during the imaging, as most glasses appear completely opaque in infrared. While this may be reasonable for security applications, it isn't for most others.

2.5.3 Results

For each of the three views, 24 normal-expression images were used as the training set, and 24 smiling-expression images were used as the test set. For the frontal and 45° views only one person was incorrectly classified; the profile view classified all 24 people correctly. A separate face space was used for each test. Figure (2.1) for an example of the training images, and Figure (2.2) for an example of the eigenfaces generated from this training set.

2.6 Face Recognition using Eigen Faces

Once the optimal coordinate system has been calculated any facial image can be projected into face-space by calculating its projecting onto each axis.

Thus for some test images T_a we can find its projection onto axis i w_k by

$$\omega_k = u_k^T(\Gamma_a - \psi) \tag{2.10}$$

Now let us define the vector Ω_a, which contains the projections of T_a onto each of the dominant eigenvectors.

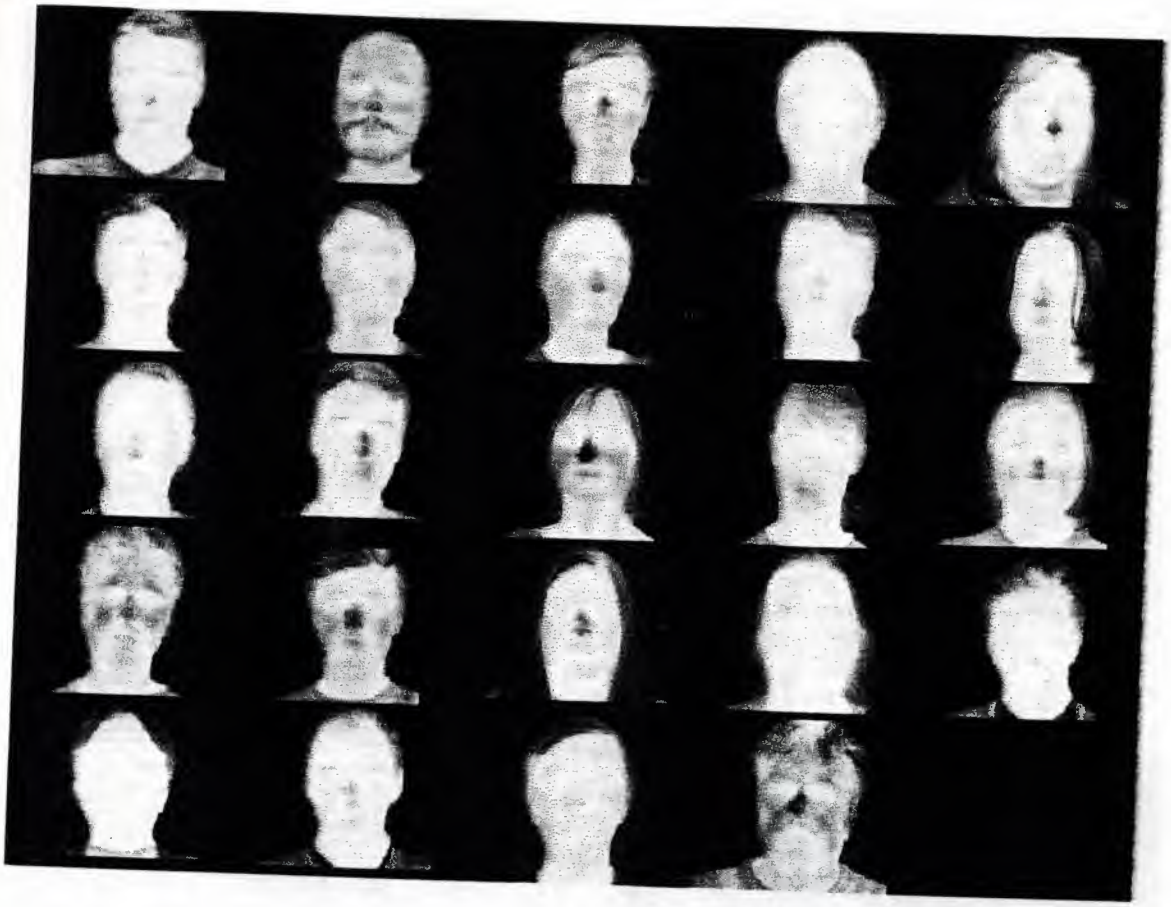


Figure (2.1) Training set example

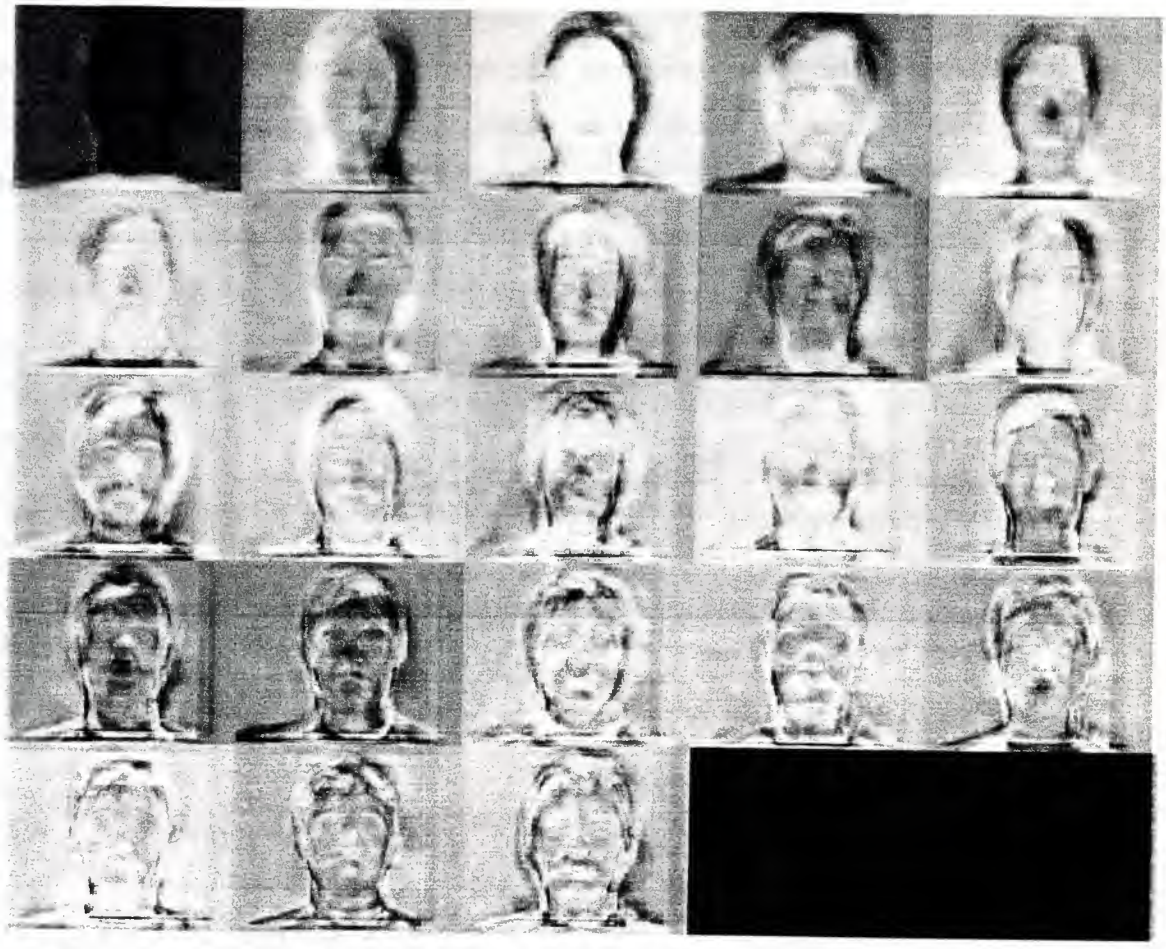


Figure (2.2) Eigen faces created form the training set.

$$\Omega_a = [\omega_1, \omega_2, \ldots \omega_{M'}] \tag{2.11}$$

Where M' is the number of dominant eigenvectors $M' << 16384$.

Given a set of photographs of 30 people $T_1 - T_{30}$ we can then determine the identity of an unknown face T_a by finding which photograph it is most closely positioned to in face-space. A simplistic way to achieve this would be to determine the Euclidean distance.

$$\epsilon_n = \|\Omega_a - \Omega_n\| \tag{2.12}$$

Where Ω_n is the projection of T_n into face-space.

Statistically, the Euclidean distance can be used to model the probability that T_a and T_n are the same person through the use of a high dimensional Gaussian distribution.

This distribution will have uniform variance in each of the eigen-dimensions since we give equal weighting to each of the projection errors when we calculate the Euclidean distance. Here the projection error is simply the difference between Ω_a and Ω_n for eigen-dimension i.

The purpose of this model is to convert the distance measure into a probability. Assuming that the data follows Gaussian distributions the relationship is as follows.

$$\begin{aligned} P((\Gamma_a \mid \Omega_n) \mid (\Gamma_a \mid \Omega)) &= e^{-\sum_{i=1}^{N} \frac{\Delta w_i^2}{2\lambda_i}} \\ &= e^{-\sum_{i=1}^{N} \frac{\Delta w_i^2}{2\lambda_i}} \\ &= e^{-\sum_{i=1}^{M} \frac{\Delta w_i^2}{2\lambda_i}} \cdot e^{-\sum_{i=M+1}^{N} \frac{\Delta w_i^2}{2\lambda_i}} \end{aligned}$$

(2.13)

Now if we consider the minor principal components to be insignificant.

$$\begin{aligned} P((\Gamma_a \mid \Omega_n) \mid (\Gamma_a \mid \Omega)) &\simeq e^{-\sum_{i=1}^{M} \frac{\Delta w_i^2}{2\rho}} \\ &\simeq e^{-\frac{d^2}{2\rho}} \end{aligned}$$

$$d = \sum_{i=1}^{N} \frac{\Delta w_i^2}{\lambda_i}$$

$$P((\Gamma_a \mid \Omega_n) \mid (\Gamma_a \mid \Omega)) = e^{-\frac{d}{2}}$$

(2.14)

Furthermore, it can be shown that the optimal value for p is simply the average of the eigenvalues for the first M principal components. Whilst this method has been shown to work, it ignores the fact that each of the eigen-dimensions exhibit a different variance. A measure, which takes this into account by normalizing each of the eigen-dimensions for unity variance, is the Mahalanobis distance. The Mahalanobis distance is defined as:

$$d = \sum_{i=1}^{N} \frac{\Delta w_i^2}{\lambda_i}$$

(2.15)

This distance measure will result in high-dimensional Gaussian distributions with different variances in each of the eigen-dimensions. This is illustrated below for the simplistic case of M = 2. The circular cross-section of the Euclidean distance probability model and ellipsoidal cross-section of the Mahalanobis distance probability model and the ellipsoidal cross-section of the Mahalanobis distance probability model. Here the height of the graph represents the probability that T_a and T_n are the same

person, whilst the two horizontal dimensions correspond the projection error in the first and second principal components.

By a method similar to that above, the relationship between probability and the Mahalanobis distance can be found to be:

$$P((\Gamma_a \mid \Omega_n) \mid (\Gamma_a \mid \Omega)) = e^{-\frac{1}{2}} \tag{2.16}$$

2.7 Local Feature Analysis

Local feature analysis is derived from the eigenface method but overcomes some of its problems by not being sensitive to deformations in the face and changes in poses and lighting. Local feature analysis considers individual features instead of relying on only a global representation of the face. The system selects a series of blocks that best define an individual face. These features are the building blocks from which all facial images can be constructed.

The procedure starts by collecting a database of photographs and extracting eigenfaces from them. " Applying local feature analysis, the system selects the subset of building blocks, or features, in each face that differ most from other faces. Any given face can be identified with as few as 32 to 50 of those blocks. The most characteristic points as shown to the right are the nose, eyebrows, mouth and the areas where the curvature of the bones changes.

The patterns have to be elastic to describe possible movements or changes of expression. The computer knows that those points, like the branches of a tree on a windy day, can move slightly across the face in combination with the others without losing the basic structure that defines that face.

2.7.1 Learning Representative Features for Face Recognition

There is psychological [7] and physiological [8,9] evidence for parts based representations in the brain. Some face detection algorithms also rely on such representations. However, the spatial shape of their local features is often subjectively defined instead of being learnt from the training data set.

Yang *et al.* [10] describe a method for frontal face detection on 20x20 regions. They assign a weight to every possible pixel value at every possible location within the region.

The weights are determined by an iterative training procedure using the Winnow update rule. Once they have determined the weights they can classify any region by looking up and summing the weights corresponding to each pixel value. Thus each of their local features relies on only one pixel.

Colmenarez and Huang [11] used first order Markov Chain model over 11x11 input region to model face and non-face class conditional probabilities. To build the model, they calculate 1st order conditional probabilities for all pixels pairs, indicating that each of their local feature involve two pixels. The training procedure finds the mapping from the region into a 1 dimensional array with maximum sum of the corresponding 1st order conditional probabilities according to the training set. Any region can then be classified as face or non-face by looking up and summing the probabilities corresponding to the intensity values of each selected pixel pair.

Schneiderman and Kanade [12] argued that local features, which are too small – one pixel at the extreme – would not be powerful enough to describe anything distinctive about the object. They use multiple appearance-based detectors that span a range of the object's orientation. Each detector uses a statistical model to represent object's appearances over a small range of views, to capture variation that cannot be modeled explicitly. They use rectangular sub regions at multi-scales as local features in the statistical model. Size of those rectangles is pre-defined.

Burl and Perona [13] detected 5 types of features on the face: the left eye, right eye, nose/lip junction, left nostril, and right nostril. They assume that the feature detectors for each feature are fallible. Since they assume only one face is present in each image, at most one feature response is correct for each type of detector. Such handpicked local features can also be found in Pentland's method [13].

Rowley *et al.* [14] used a multiplayer perceptron neural network system for classification. A 20x20 input region is divided into blocks of 5x5, 10x10, or 20x5. Each hidden unit has one block as its receptive field. In their experiments with modular systems, they separately trained two or three of the above networks and then applied various methods for merging their results. Since the hidden units have only local support, we can infer that this particular network topology emphasizes local features over global one.

Viola and Jones [15] argued that the most common reason for using features rather than the pixels directly is that features can act to encode ad-hoc domain knowledge that is difficult to learn using a finite quantity of training data. Given a

24x24 region, they use an exhaustive set of three kinds of Harr like rectangular features. A following AdaBoost procedure is applied to learn important features from the over complete feature set. In contrast to their method, Papageorgiou et al. [16] use a over complete set of Quadruple density 2D Harr basis at scales 4× 4 and 2× 2 pixels since they think the dimensions correspond to typical facial features for their 19× 19 face images. They average the normalized coefficients over the entire set of example to identify the important Harr basis.

From the methods above it had been conclude that there are two main steps for learning local features. The first step determines various characteristics of the local feature, including size, shape, location and calculations over the corresponding pixels, etc. Generally an over complete feature set is required for further selection of the features. The second step aims to find out the important features among the over complete set with the knowledge contained in the training data. Most previous face detection algorithms put learning procedure in the second step while little or no attention was put in the much, if not more, important first step. Instead, they define the spatial shape and other properties of their local features manually and intuitively.

Several existing algorithms can be applied to learn parts based representation from examples. Local feature analysis (LFA) [17] is a method for extracting local topographic representation in terms of local features. The extraction is from the global PCA basis, also based on second order statistics. The LFA representation enables use of specific local features for identification instead of a global representation.

Independent component analysis [18,19] is a linear nonorthogonal transform, which makes unknown linear mixtures of multi-dimensional random variables as statistically independent as possible. It not only decor relates the second order statistics but also reduces higher-order statistical dependencies. It extracts independent components even if their magnitudes are small whereas PCA extracts components having largest magnitudes. It is found that independent component of natural scenes are localized edge like filters [20].

The projection coefficients for the linear combinations in the above methods can be either positive or negative, and such linear combinations generally involve complex cancellations between positive and negative numbers. Therefore, these representations lack the intuitive explanation from the relationship between parts and the whole.
Non-negative matrix factorization (NMF) [21] imposes the non-negativity constraints in learning basis images. The pixel values of resulting basis images, as well as coefficients

for reconstruction, are all non-negative. By this way, only non-subtractive (or additive) combinations are allowed. This ensures that the components are combined to form a whole in an accumulative means. For this reason, NMF is considered as a procedure for learning a parts based representation [21]. However, Li et al. [22] found that the non-negative basis components learned by NMF are not necessarily as localized as describe in the original NMF paper, at least for the ORL face database; moreover, the original NMF representation yields low recognition accuracy – lower than can be obtained by using the standard PCA method. Motivated by these observations, they proposed a local non-negative matrix factorization (LNMF) algorithm, which optimizes the objective to learn truly localized, parts-based components. Their experimental results demonstrate that LNMF basis leads to much more stable recognition results when there are occlusions, better than the standard NMF and PCA methods.

LNMF is employed to learn parts-based components. It has been applied on the input region (I) and (1-I) to get both bright local components and dark local components, suppose the input region (I) has the pixel value in the range of [0, 1]. Each local feature is calculated from a bright component and a dark one. We can then construct a face detector by selecting a small number of important features using AdaBoost from the over complete local feature set.

2.7.2 NMF and Constrained NMF

Given a set of NT training images represented as an $n \times NT$ matrix $X = [x_{ij}]$, each column of which contains n nonnegative pixel values. Denote a set of $m \leq n$ basis images by an $n \times m$ matrix W. Each image can be represented as a linear combination of the basis images (eigenvectors of unit length), and hence the (approximate) factorization

$$X \approx WH$$

Where H is the matrix of $m \times NT$ coefficients or weights. Dimension reduction is achieved when $m < n$.

The PCA factorization requires that the basis images (columns of W be orthonormal and the rows of H be mutually orthogonal. It imposes no other constraints than the orthogonality, and hence allows the entries of W and H to be of arbitrary sign. The NMF and LNMF, however, allow only positive coefficients and thus additive combinations of basis components.

2.7.3 NMF

NMF imposes the non-negativity constraints instead of the orthogonality. As the result, the entries of w and h are all non-negative. This way, only additive combinations are allowed, and no subtractions can occur. This is believed to be compatible to the intuitive notion of combining parts to form a whole, and is how NMF learns a parts-based representation [8]. It is also consistent with the physiological fact that the firing rate is non-negative. NMF uses the divergence of X from Y =WH, defined as

$$D(\mathbf{X}\|\mathbf{Y})=\sum_{i,j}\left(x_{ij}\log\frac{x_{ij}}{y_{ij}}-x_{ij}+y_{ij}\right) \tag{2.17}$$

As the measure of cost for factorizing X into WH. An NMF factorization is defined as a solution to the following constrained optimization problem

$$\min_{\mathbf{W},\mathbf{H}} D(\mathbf{X}\|\mathbf{WH})$$
$$\text{s.t}\quad \mathbf{W},\mathbf{H}\geq 0,\ \sum_i w_{ij}=1\ \ \forall j \tag{2.18}$$

Where W, H≥ 0 means that all entries of W and H are nonnegative.

2.7.4 Constrained NMF

The NMF model defined by (3) does not impose any constraints on the spatial locality. Therefore, minimizing the objective function can hardly yield a factorization, which reveals local features in the data X. LNMF is aimed to improve the locality of the learned features by imposing additional constraints. Let $(W^TW)=U=[u_{ij}],(HH^T)=V=[v_{ij}]$. The following three additional constraints are imposed on the NMF basis:

1. The number of basis components, which is required to represent X, should be minimized. This requires that a basis component should not be further decomposed into more components. Let w_j be a basis vector. Given the existing constraints $\sum w_{ij}=1$ for all j, the value $\sum_i w_{ij}^2$ 2 should be as small as possible so that w_j contains as many non-zero elements as possible. This constraint can be formulated as minimizing $\sum i\ u_{ii}$.
2. To minimize redundancy between different bases, different bases should be as orthogonal as possible. This can be imposed by minimizing $\sum_{i\neq j} u_{ij}$

3. Only basis containing most important information need to be retained. Given that every image in X is normalized into a range such as in [0, 1], the total "activity" on each component, i.e. the total squared projection coefficients summed over all training images, should be maximized. This is imposed by $\sum_i$ vii = max.

Incorporating the above constraints into the original NMF formulation, the new objective function for LNMF is:

$$D(\mathbf{X} \| \mathbf{WH}) = \sum_{i,j} \left[x_{ij} \log \frac{x_{ij}}{y_{ij}} - x_{ij} + y_{ij} \right] + \alpha \sum_{i,j} u_{ij} - \beta \sum_{i} v_{ii} \qquad (2.19)$$

Where $\alpha, \beta > 0$ are some constants.

A comparison shown in Figure (2.4) gives the different factorization results (image basis) of NMF and LNMF in our face database. LNMF basis are obviously more localized than NMF basis. One should note that because of the orthogonality constraint , the coefficient matrix H is no longer sparse in LNMF as it is in NMF. But this takes no effect since only image basis has been used.

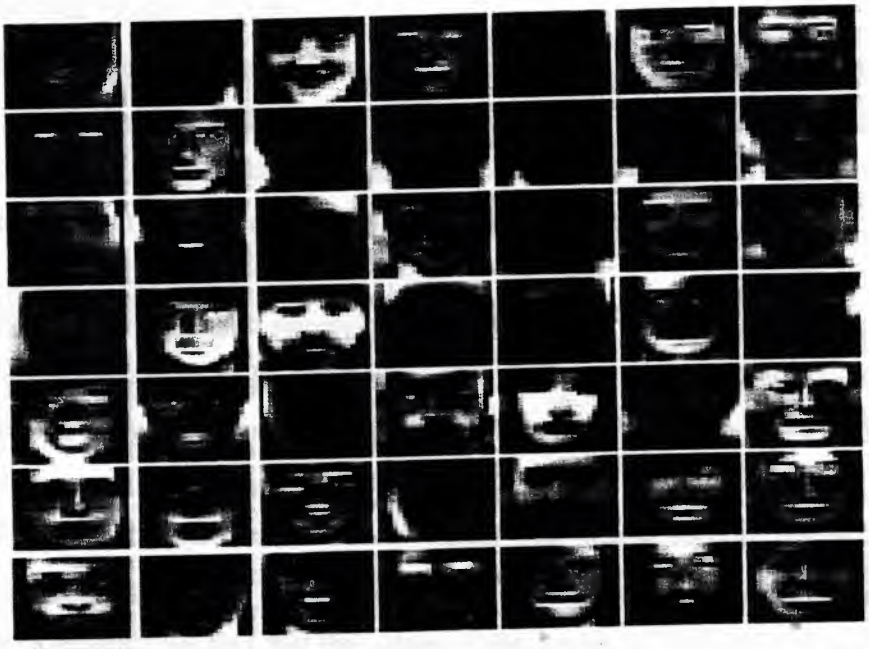


Figure 2.3 Factorization result of 49 bases on face database

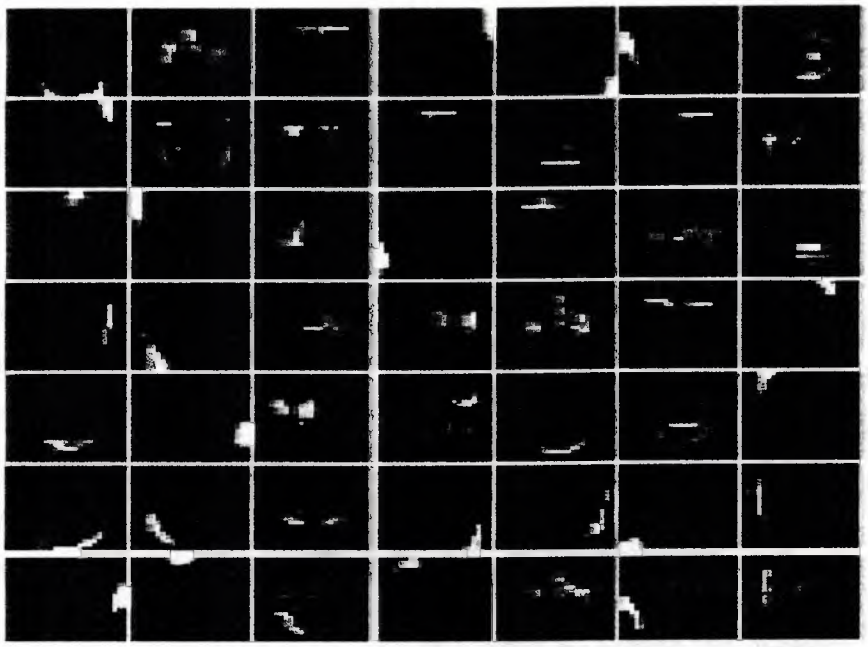


Figure 2.4 Constrained NMF Obviously LNMF has more localized basis.

2.7.5 Getting Local Features

Investigating the Harr-like features used in Viola's [15] and Papageorgiou's [16] systems, we notice that differential operator is robust to varying lighting. Inspired by this, we desire to get local components that contain both bright and dark parts of the faces, and then put differential operator on bright and dark components to get the final value.

To achieve this, each sample (I) in X is mapped into X' as (1-I), suppose (I) have its pixel value in range [0, 1]. Then apply LNMF on both the sample set X and X' to get two sets of basis, W and W', which could be used as bright and dark components, respectively.

This can be explained as below. Recall that in last section, the matrix V = (HH^T) indicates the energy relationship between the basis (include each basis itself). From the experiment we find that the values of the entries of V matrix are much closed to each other, implying that each basis contribute roughly the same to the whole data set. Thus we cannot say individually which component is more "bright" than others. That is why LNMF is performed on the other sample set X'.

Given the two basis sets W and W', for each input region we can get two coefficient vector h and h'. The local feature set corresponding to the basis sets could be

$\{hi - h_j^l\}, \forall i, j$. In practice, several local feature sets, correspond to different basis sets, are combined together to form an over complete feature set. In next section, AdaBoost is applied on the set to select important features and construct the classifier at the same time.

2.7.6 ADABOOST for Feature Selection

After the process described in previous section, an over-complete set of local features has been obtained. Using the entire feature set is obviously infeasible in practice. Oppositely, we seek for an approach to select those most discriminating features. Viola [15] uses a variant of AdaBoost to select features from an overcomeplete Harr-like feature set and train the classifier. The similar method is being used in this project.

The AdaBoost algorithm was first introduced in 1995 by Freund and Schapire [23]. In its original forms, the goal of AdaBoost is to improve the performance of any given classification algorithms via combining a collection of classification functions to form a stronger classifier. These classification functions, in the language of boosting, are usually called weak learners. The major idea of AdaBoost is to enforce the weak learners to focus on the examples misclassified by previous classifiers. It does this by adjusting the weight of each training sample. In the initial state, all weights are set equally but on each round of training, the weights of misclassified samples will be increased in the proportion of previous classification errors.

Viola et al. adapted this greedy boosting procedure to feature selection. The weak learner is restricted to a set of classification functions while each of which depends on only one single feature. For each feature, the weak learner determines an optimal threshold classification function, such that the number of misclassified examples is minimized.

The procedure of applying AdaBoost to feature selection [15] can be formulated as follows. Given a set of training examples $(x_1, y_1), (x_2, y_2), \ldots (x_m, y_m)$ where x_1 represents 20×20. image patterns and $y_1 = 0,1$ for faces and nonfaces examples respectively, assign a weight value w_i to each. Example (x_i, y_i). Before the training all w_i are equal and the sums of all eights are normalized to unit. For each feature, we train a simple Bayesian classifier which is restricted to this single feature. The classification

error is evaluated with respect to $w_i, \varepsilon_j = \sum w_i |h_j(x_i) - y_i|$. The classifier, h_i with the lowest error ε_j chosen as one component of the final strong classifier and its importance in final classification function is determined by classification rate. Subsequently all weights are updated in terms of the training error before next round of training.

Besides Viola's successful experience, the formal guarantees provided by the AdaBoost learning procedure are quite strong. Freund and Schapire [23] proved that the training error of the strong classifier approaches zero exponentially in the number of rounds. More importantly a number of results were later proved about generalization performance . The key insight is that generalization performance is related to the margin of the examples, and that AdaBoost achieves large margins rapidly.

Using the same learning framework, we can now compare our learnt local features with Viola's Harr like features. This comparison is done on the training set which contains 5,000 face samples and 10,000 non-face samples. LNMF representations of dimensions 25, 36, 45, 47, 49, 51, 53, 55, 64, 81, 100 are computed from the training set to form a feature set with 37648 local features. From each features set, we select 200 features. The error ε_i of first 20 features is shown in Figure 2. Figure 3 shows the ROC curves of the two classifiers on our testing set which contains 2000 face samples and 5000 non-face samples.

2.7.7 Experimental Results

This section describes the final face detection system, including training data preparation, training procedure, and the performance comparison with state-of-the-art face detection system.

2.7.8 Training Data Set

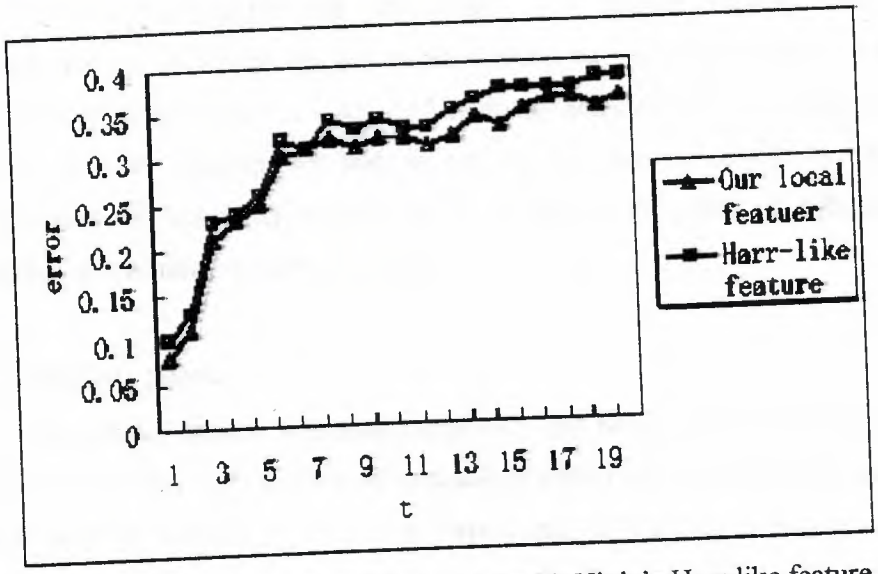


Figure 2.5 Comparison of local feature set with Viola's Harr-like feature set using the first 20 features selected by AdaBoost.

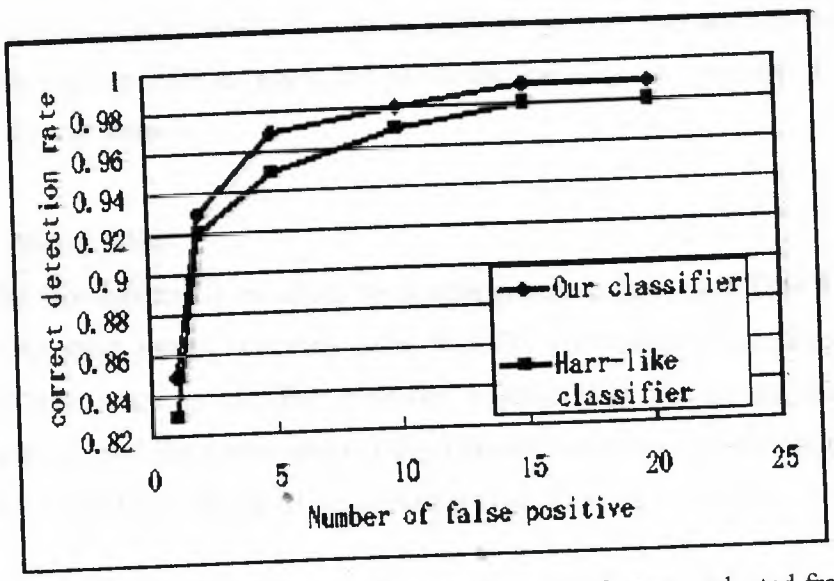


Figure 2.6 ROC curves of the two-classifier using 200 features selected from local feature set and Harr-like feature set.

The frontal face images are collected from databases of CMU, Rockefeller, Umist, Corel and our own database. There are more than 7,000 faces in total. 5,000 of them are selected as positive training samples and 2,000 as testing samples. Each face

image is resized into 20x20 and aligned by the center point of the two eyes and the horizontal distance between the two eyes.

For non-face training set, an initial 10,000 non-face samples were selected randomly from 15,000 large images, which contain no face. The 5,000 testing non-face samples mentioned in section 4 are also randomly selected from the large images. All samples, both in training set and in testing set, are processed by illumination compensation and histogram equalization to minimize the effect of different lighting conditions, as was done in Rowley's method.

2.7.9 Training phase

The similar feature selection framework has been used with Viola 's method [15]. The final detector is a 29-layer cascade of classifier. 2 features had been used in the first layer, 5 features in the second layer, and 20 features in three layers. In the fifteenth layers 200 features are used for training the classifier.

The initial 10,000 non-face samples are used to train the first three layers. In subsequent layers, scanning the partial cascade across large non-face images and collecting false positive samples obtain non-face samples. Different sets of nonface sub-windows are used in training the different classifiers to ensure that they are somewhat independent and use different features.

2.7.10 Testing phase

The face detector is tested on the images collected from the MIT+CMU test set [24]. For an input image, scanning each 20 x 20 sub-window exhaustively in both spatial and scale space, as was done is Rowley's system [14]. The starting scale is 1, the scale step is 1.25 and the spatial step is 1 pixel at each scale level. Results from different scale levels or spatial locations are merged to get the final result.

Figure 2.7 An example image of Output by face detector.

2.8 Face Modeling for Recognition

Current trend in face recognition is to use 3D face model explicitly. As an object-centered representation of human faces, 3D face models are used to overcome the large amount of variations present in human face images. These variations, which include extra-subject variations (individual appearance) and intra-subject variations (3D head pose movement, facial expression, lighting, and aging) are still the primary challenges in face recognition. However, the three major recognition algorithms [25] merely use viewer-centered representations of human faces: (i) a PCA-based algorithm; (ii) An LFA-based (local feature analysis) algorithm; and (iii) a dynamic-link-architecture-based paradigm.

Researchers in computer graphics have been interested in modeling human faces/heads for facial animation. We briefly review three major approaches to modeling human faces. DeCarlo et al. [26] use the anthropometric measurements to generate a general face model. This approach starts with manually constructed B-spline surfaces and then applies surface fitting and constraint optimization to these surfaces.

In the second approach, facial measurements are directly acquired from 3D digitizers or structured light range sensors. Water's [27] face model is a well-known instance. A morphable model [28] was constructed from a linear combination of eigenshapes and a linear combination of eigentextures, based on laser scans of 200

subjects. The third approach, in which models are reconstructed from photographs, only requires low-cost and passive input devices (video cameras). For instance, Chen and Medioni [34] build face models from a pair of stereo images. However, currently it is still difficult to extract sufficient information about the facial geometry only from 2D images. This difficulty is the reason why Guenter et al. [29] utilize a large number of fiducially points to capture 3D face geometry for photo realistic animation. Even though we can obtain dense 3D measurements from high-cost 3D digitizers, it still takes too much time to scan every face. Hence, advanced modeling methods, which incorporate some prior knowledge of facial geometry, are needed. Reinders et al. [30] use a fairly coarse wire-frame model, compared to Water's model, to do model adaptation for image coding. Lee et al. [31] modify a generic model from two orthogonal pictures (frontal and side views), or from range data for animation. Lengagne et al. [32] and Fua [33] fit a range animation model to uncalibrated videos using bundle-adjustment and least squares fitting, given five manually selected features points and initial camera positions. Zhang [35] deforms a generic mesh model to an individual's face based on two images, each of which contains five manually picked markers.

A face modeling method is proposed , which adapts an existing generic face model (a priori knowledge of human face) to an individual's facial measurements. Our goal is to employ the learned 3D model to verify the presence of an individual in a face image database/video, based on the estimates of head pose and illumination.

2.8.1 Face Modeling

An individual face model is starting with a generic face model, instead of extracting isosurfaces directly from facial measurements (range data or disparity maps), which are often noisy (e.g., near ears and nose) as well as time consuming, and usually generates equal-size triangles. Our modeling process aligns the generic model using facial measurements in a global-to-local way so that feature points/ regions that are crucial for recognition are fitted to the individual's facial geometry.

2.8.2 Generic Face Model

The Water's animation model has been chosen [36], which contains 256 vertices and 441 facets for one half of the face. The use of triangular meshes is suitable for the free-form shapes like faces and the model captures most of the facial features that are

needed for face recognition. Figure (2.8) shows the frontal and a side view of the model, and features such as eyes, nose, mouth, face border, and chin. There are openings at both the eyes and the mouth.

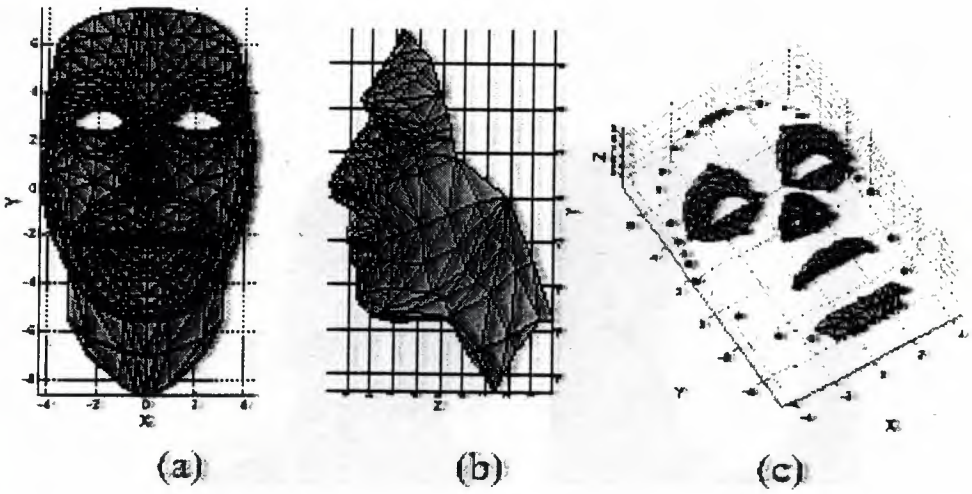


Figure 2.8 3D triangle-mesh model and its feature components;(a) Frontal view;(b) Slide view;(c) feature components.

2.8.3 Facial Measurements

Facial measurements include information about face shape and face texture. 3D shape information can be deri ed from a stereo pair combined with shape from shading, a sequence of frames in a video, or obtained directly from range data. The range database of human faces used here [37] was acquired using a Minolta Vivid 700 digitizer. It generates a registered200× 200 range map and a 400× 400color image. Figure (2.9)(a,b) shows a range map and a color image of a frontal view, and the texture-mapped appearance. The locations of face and facial features such as eyes and mouth in the black and white image can be obtained by our face detection algorithm [38]. The corners of eyes, mouth, and nose can be easily obtained based on the locations of detected eyes and mouth. Figure (2.10)(c,d,e) shows the detected feature points.

2.8.4 Model Construction

Our face modeling process consists of global alignment and local adaptation. Global alignment brings the generic model and facial measurements into the same coordinate system. Based on the head pose and face size, the generic model is translated

and scaled to fit the facial measurements. Figure (2.10) shows the global alignment results in two different modes. Local adaptation consists of local alignment and Local feature refinement. Local alignment involves translating and scaling several model features, such as eyes, nose, mouth, and chin to fit the extracted facial features. Local feature refinement makes use of displacement propagation and 2.5D active contours to smoothen the face model and to refine local features. Both the alignment and the refinement of each feature (shown in Fig. 2.8(c)) is followed by displacement (of model vertices) propagation, in order to blend features in the face model.

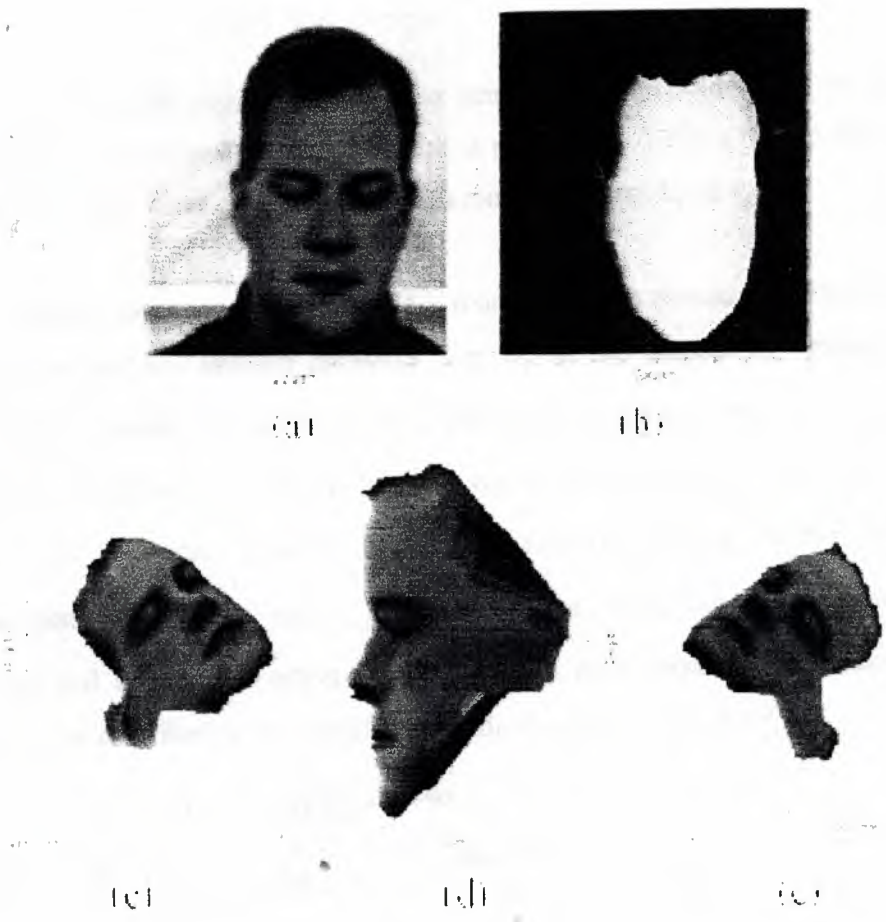


Figure 2.9 Range data of a face (a) Color texture; (b) Range map; and with texture mapping of (c) A left view; (d) A profile view; (e) A right view.

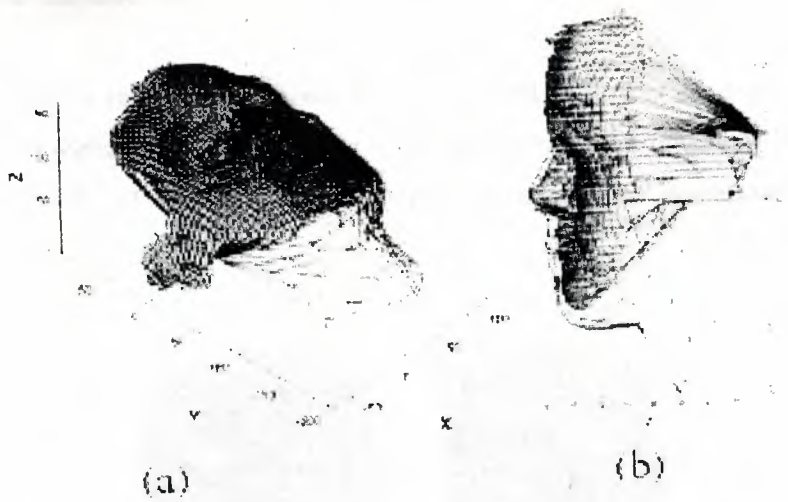


Figure 2.10 Global alignment from the generic model (Bold lines) to the facial measurements (gray lines): the target mesh is plotted in (a) For a hidden line removal mode for a frontal view; (b) For see-through mode for a profile view.

Displacement propagation inside a triangular mesh mimics the transmission of message packets in computer networks. Let N_i be the number of vertices that are connected to a vertex i, J_i be the set of all the indices of vertices that are connected to a vertex i, w_i be the sum of weights from all vertices that are connected to vertex i, and d_{ij} be the Euclidean distance between a vertex V_i and a vertex V_j ΔV_j is the displacement of vertex V_j, and $\alpha\_$ is a decay factor, which can be determined by the face size and the size of active facial feature in each coordinate. Equation.(2.20) computes the contribution of vertex V_j to the displacement of vertex V_i

$$\Delta V_{ij} = \begin{cases} \Delta V_j \cdot \frac{w_i - d_{ij}}{w_i \cdot N_i - 1} \cdot e^{-\alpha d_{ij}}, & N_i > 1,\ w_i = \sum_{j \in J_i} d_{ij} \\ \Delta V_j \cdot e^{-\alpha d_{ij}}, & N_i = 1,\ j \in J_i. \end{cases} \quad (2.20)$$

The total displacement ΔV_i of $V_i\_$ can be obtained by summing up all the displacement contributed from its neighbor vertices. The displacement will decay during propagation and it continues for a few iterations, which is determined by the number of edge connections from the current feature to the nearest neighbor feature. In the future implementation, we will include symmetric property of a face and facial topology in

computing this displacement. Figure (2.11) shows the results of local alignment for a frontal view.

Local feature refinement follows local alignment to further adapt the results of alignment to an individual face by using 2.5D active contours (snakes). We modify Amini et al.'s [39] 2D snakes for our 3D active contours on boundaries of facial features.

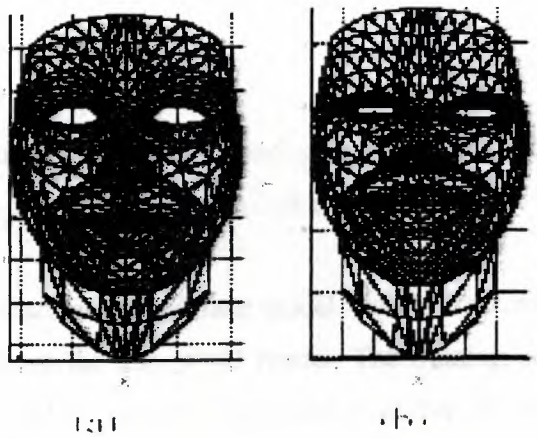


Figure 2.11 Local Feature alignment and displacement and displacement propagation shown for frontal views: (a) The generic model;(b) The model adapted to eyes, nose, mouth, and chin.

Hence, the crucial initial contours for fitting the snakes are known in our generic face model. Another important point for fitting snakes is to find appropriate external energy maps that contain local maximum/minimum at the boundaries of facial features. For the face and the nose, the external energy is computed by the maximum magnitude of vertical and horizontal gradients from range measurements. These two facial features have steeper borders than others. For features such as eyes and the mouth, the product of the magnitude obtains the external energy of the luminance gradient and the squared luminance. Figure (2.12) shows the results of local refinement for the left eye and nose.

Although our displacement propagation smoothes nonfeature skin regions in local adaptation, they can be further updated if a dense range map is available. Figure (2.13) shows the overlay of the final adapted face model in red and the target facial measurements in blue. For a comparison with

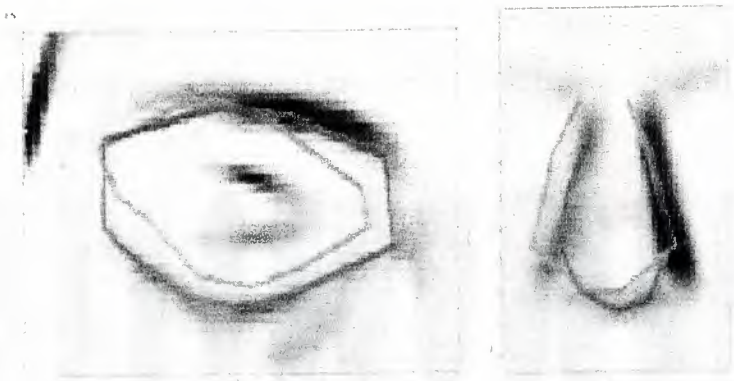


Figure 2.12 Boundary alignment: initial (inner) and refined (outer) contours overlaid on the energy maps for (a) Left eye and (b) Nose.

Figure (2.9) shows the texture-mapped face model. Further face recognition algorithm [40] is used to demonstrate the use of 3D model. The training database contains504 image from 28 subjects and15 _ images generated from our 3D face model, shown in Figure (2.14). All the 10 _ test images were correctly matched.

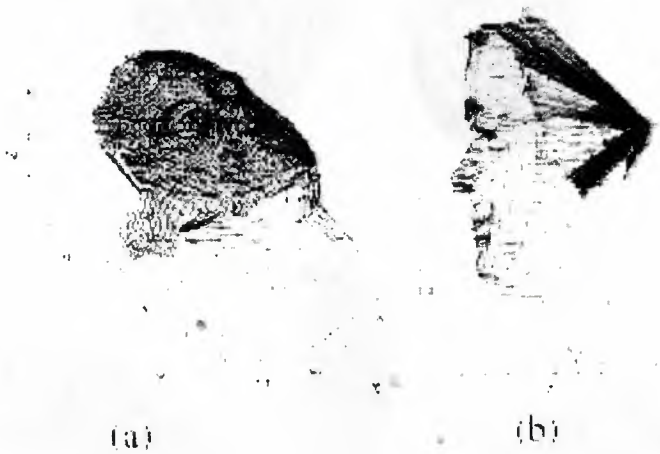


Figure 2.13 The adapted model (gray lines) overlapping the target measurement(Dark lines):The adapted model plotted (a) in 3D;(b) With colored facets at a profit view.

2.8.5 FUTURE WORK

Face modeling plays a crucial role in face recognition systems. A method had been adapted for generic face.

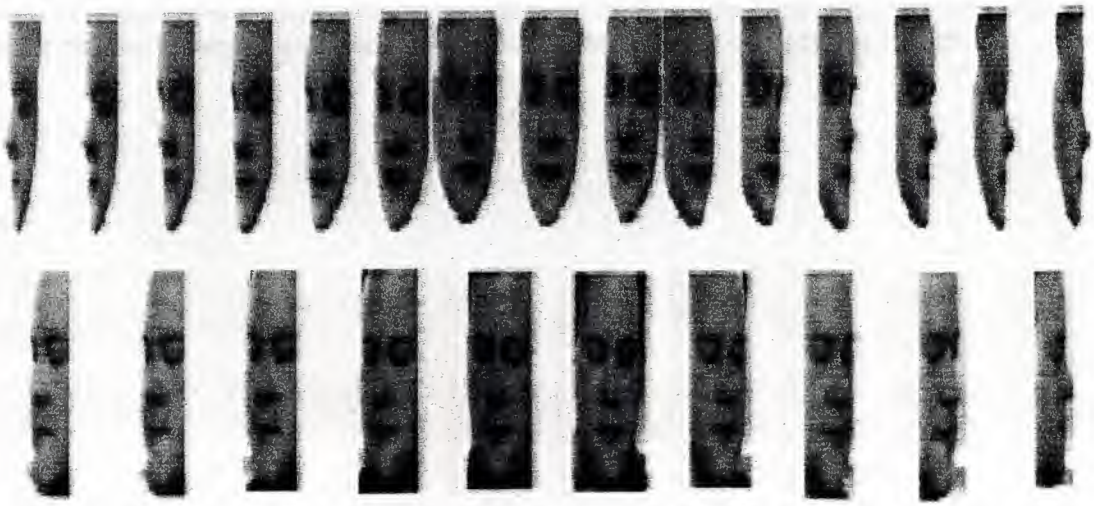


Figure 2.14 Face matching: The first row shows the 15 training images from the 3D model; the second shows 10 test images captured from a CCD camera.

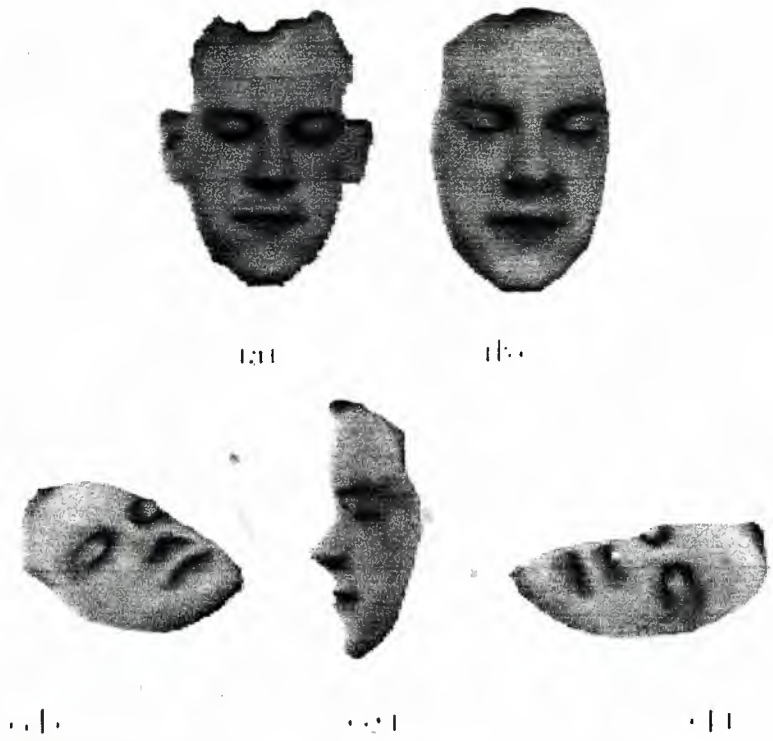


Figure 2.15 The texture-mapped (a) Input range image; adapted mesh model (b) From a frontal view;(d) From a left view; (e) From a profile view; (f) Form a right view.

Model to input facial features in a global-to-local fashion. The model adaptation first aligns the generic model globally, and then aligns and refines each facial feature locally using displacement (of model vertices) propagation and active contours associated with facial features. The final texture mapped model is visually similar to the original face. Initial matching experiments based on the 3D face model show encouraging results.

2.9 Summary

The chapter details about the techniques that are implemented now a day for recognition. Eigen faces and local feature analysis. Eigen faces are sensitive to scale reduction of less than 88% and rotations of more than 10 degrees. Hence it is essential that good scale and rotation normalization algorithms be applied prior to recognition.

The learning procedure consists of two steps. First a modified version of NMF(Non-negative matrix factorization), namely local NMF (LNMF), is applied to select a small number of local features. Second, a learning algorithm based on AdaBoost is used to select a small number of local features and yields extremely efficient classifiers. Experiments are presented which show the face detection performance is comparable to the state-of-art face recognition systems.

CHAPTER THREE

FACE RECOGNITION: A NEURAL NETWORK APPROACH

3.1 Overview

Faces represent complex, multidimensional, meaningful visual stimuli and developing a computational model for face recognition is difficult . A hybrid neural network solution which compares favorably with other methods. The system combines local image sampling, a self-organizing map neural network, and a convolutional neural network. The self-organizing map provides a quantization of the image samples into a topological space where inputs that are nearby in the original space are also nearby in the output space, thereby providing dimensionality reduction and invariance to minor changes in the image sample, and the convolutional neural network provides for partial invariance to translation, rotation, scale, and deformation. The convolutional network extracts successively larger features in a hierarchical set of layers. We present results using the Karhunen-Lo`eve transform in place of the self-organizing map, and a multi-layer perceptron in place of the convolutional network. The Karhunen-Lo`eve transform performs almost as well (5.3% error versus 3.8%). The multi-layer perceptron performs very poorly (40% error versus 3.8%). The method is capable of rapid classification, requires only fast, approximate normalization and preprocessing, and consistently exhibits better classification performance than the eigenfaces approach on the database considered as the number of images per person in the training database is varied from 1 to 5. With 5 images per person the proposed method and eigenfaces result in 3.8% and 10.5% error respectively. The recognizer provides a measure of confidence in its output and classification error approaches zero when rejecting as few as 10% of the examples. A database of 400 images of 40 individuals has been used which contains quite a high degree of variability in expression, pose, and facial details.

3.2 Introduction

The requirement for reliable personal identification in computerized access control has resulted in an increased interest in biometrics. Face recognition has the benefit of being a passive, non-intrusive system for verifying personal identity. The techniques used in the best face recognition systems may depend on the application of the system. We can identify at least two broad categories of face recognition systems:

1. We want to find a person within a large database of faces (e.g. in a police database). These systems typically return a list of the most likely people in the database [42]. Often only one image is available per person. It is usually not necessary for recognition to be done in real-time.
2. We want to identify particular people in real-time (e.g. in a security monitoring system, location tracking system, etc.), or we want to allow access to a group of people and deny access to all others (e.g. access to a building, computer, etc.) . Multiple images per person are often available for training and real-time recognition is required. In this paper, we are primarily interested in the second case. We are interested in recognition with varying facial detail, expression, pose, etc. We do not consider invariance to high degrees of rotation or scaling – we assume that a minimal preprocessing stage is available if required. We are interested in rapid classification and hence we do not assume that time is available for extensive preprocessing and normalization.

The ORL database has been used which contains a set of faces taken between April 1992 and April 1994 at the Olivetti Research Laboratory in Cambridge, *UK*[3]. There are 10 different images of 40 distinct subjects. For some of the subjects, the images were taken at different times. There are variations in facial expression (open/closed eyes, smiling/non-smiling), and facial details (glasses/no glasses). All the images were taken against a dark homogeneous background with the subjects in an up-right, frontal position, with tolerance for some tilting and rotation of up to about 20 degrees. There is some variation in scale of up to about 10%. Thumbnails of all of the images are shown in figure 1 and a larger set of images for one subject is shown in figure 2. The images are greyscale with a resolution of 92×112.

3.3 Related Work

3.3.1 Geometrical Features

Many people have explored geometrical feature based methods for face recognition. Kanade [48] presented an automatic feature extraction method based on ratios of distances and reported a recognition rate of between 45-75% with a database of 20 people. Brunelli and Poggio [47] compute a set of geometrical features such as nose width and length, mouth position, and chin shape. They report a 90% recognition rate

on a database of 47 people. However, they show that a simple template matching scheme provides 100% recognition for the same database. Cox et al. [49] have recently introduced a mixture-distance technique which achieves a recognition rate of 95% using a query database of 95 images from a total of 685 individuals. Each face is represented by 30 manually extracted distances. Systems, which employ precisely measured distances between features, may be most useful for finding possible matches in a large mugshot database. For other applications, automatic identification of these points would be required, and the resulting system would be dependent on the accuracy of the feature location algorithm. Current algorithms for automatic location of feature points do not provide a high degree of accuracy and require considerable computational capacity .

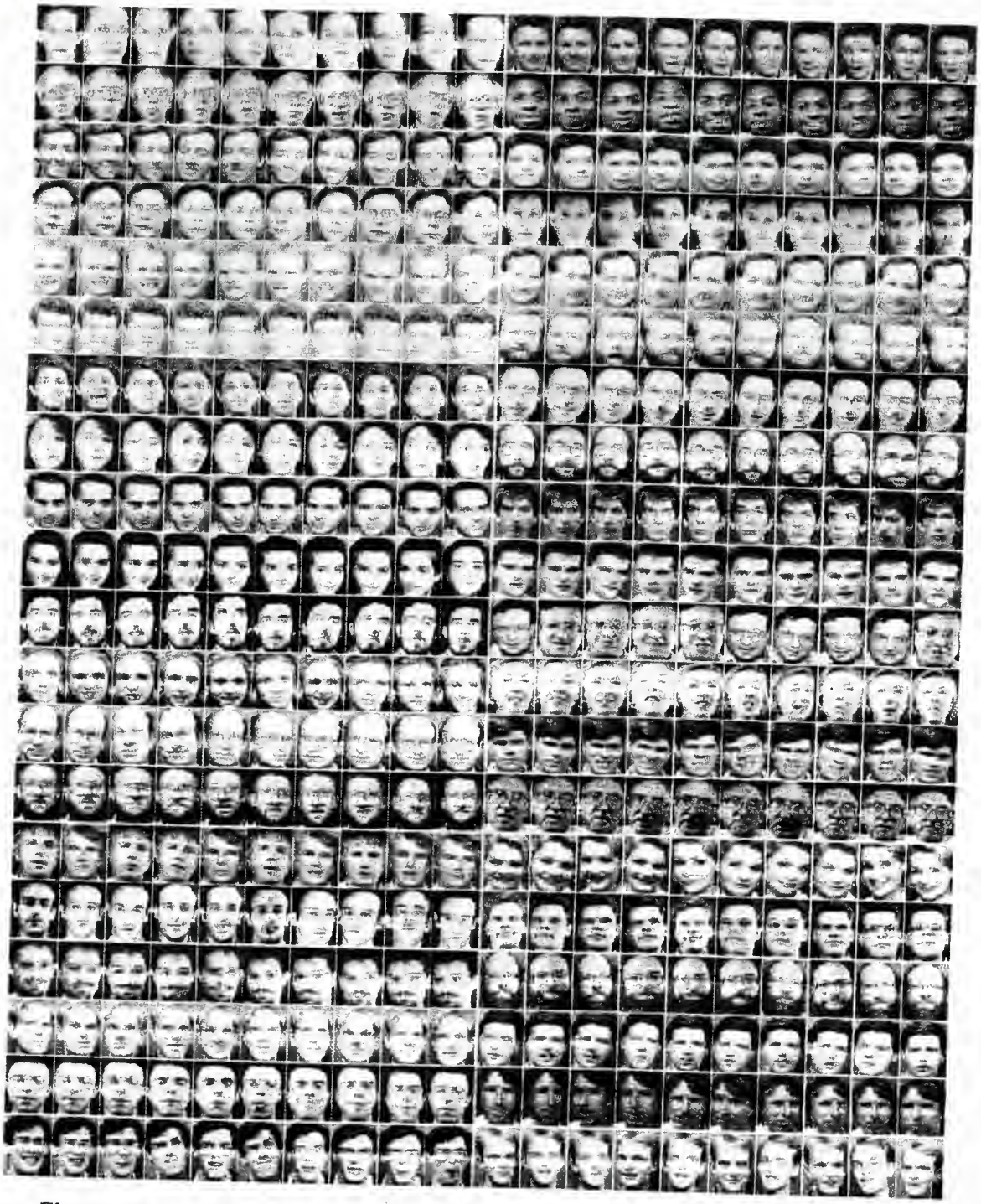


Figure 3.1 The ORL face database. There are 10 images each of the 40 subjects.

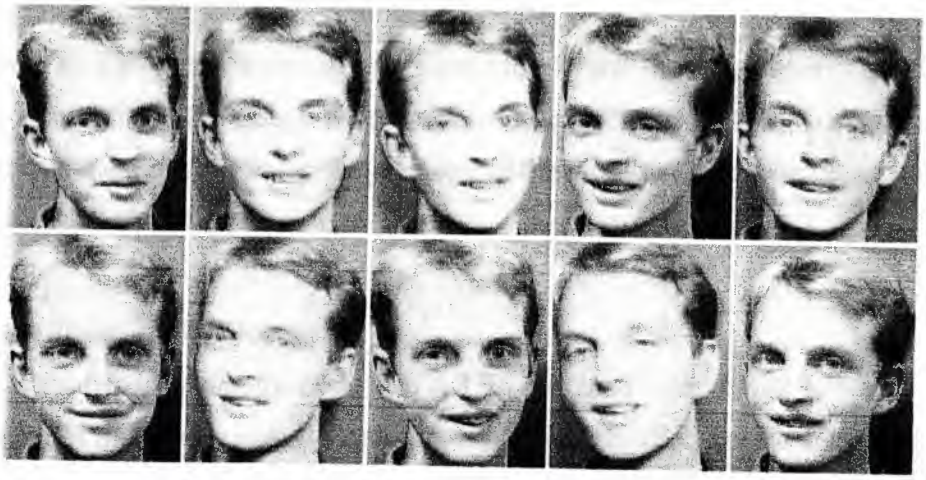


Figure 3.2 The set of 10 images for one subject. Considerable variation can be seen.

3.4 Eigenfaces

High-level recognition tasks are typically modeled with many stages of processing as in the Marr paradigm of progressing from images to surfaces to three-dimensional models to matched models . However, Turk and Pentland [1] argue that it is likely that there is also a recognition process based on low-level, two dimensional image processing. Their argument is based on the early development and extreme rapidity of face recognition in humans, and on physiological experiments in monkey cortex which claim to have isolated neurons that respond selectively to faces . However, it is not clear that these experiments exclude the sole operation of the Marr paradigm.

Turk and Pentland [1] present a face recognition scheme in which face images are projected onto the principal components of the original set of training images. The resulting *eigenfaces* are classified by comparison with known individuals. Turk and Pentland present results on a database of 16 subjects with various head orientation, scaling, and lighting. Their images appear identical otherwise with little variation in facial expression, facial details, pose, etc. For lighting, orientation, and scale variation their system achieves 96%, 85% and 64% correct classification respectively. Scale is renormalized to the eigenface size based on an estimate of the head size. The middle of the faces is accentuated, reducing any negative affect of changing hairstyle and backgrounds.

In Pentland et al. [2] good results are reported on a large database (95% recognition of 200 people from a database of 3,000). It is difficult to draw broad conclusions as many of the images of the same people look very similar, and the database has accurate

registration and alignment [43]. In Moghaddam and Pentland [3], very good results are reported with the FERET database – only one mistake was made in classifying 150 frontal view images. The system used extensive preprocessing for head location, feature detection, and normalization for the geometry of the face, translation, lighting, contrast, rotation, and scale.

3.5 Template Matching

Template matching methods such as [50] operate by performing direct correlation of image segments. Template matching is only effective when the query images have the same scale, orientation, and illumination as the training images [42].

3.6 Graph Matching

Another approach to face recognition is the well known method of Graph Matching. A Dynamic Link Architecture for distortion invariant object recognition which employs elastic graph matching to find the closest stored graph. Objects are represented with sparse graphs whose vertices are labeled with a multi-resolution description in terms of a local power spectrum, and whose edges are labeled with geometrical distances. They present good results with a database of 87 people and test images composed of different expressions and faces turned 15 degrees. The matching process is computationally expensive, taking roughly 25 seconds to compare an image with 87 stored objects when using a parallel machine with 23 transputers. An updated version of the technique is then used which compares 300 faces against 300 different faces of the same people taken from the FERET database. They report a recognition rate of 97.3%. The recognition time for this system was not given.

3.7 Neural Network Approaches

Much of the present literature on face recognition with neural networks presents results with only a small number of classes. In the first 50 principal components of the images are extracted and reduced to 5 dimensions using an autoassociative neural network. The resulting representation is classified using a standard multi-layer perceptron. Good results are reported but the database is quite simple: the pictures are manually aligned and there is no lighting variation, rotation, or tilting. There are 20 people in the database.

A hierarchical neural network which is grown automatically and not trained with gradient-descent was used for face recognition by Weng and Huang [40]. They report good results for discrimination of ten distinctive subjects.

3.8 The ORL Database

In [51] a HMM-based approach is used for classification of the ORL database images. The best model resulted in a 13% error rate. Samaria also performed extensive tests using the popular eigenfaces algorithm [43] on the ORL database and reported a best error rate of around 10% when the number of eigenfaces was between 175 and 199. We implemented the eigenfaces algorithm and also observed around 10% error. In [51] Samaria extends the top-down HMM of [51] with pseudo two-dimensional HMMs. The error rate reduces to 5% at the expense of high computational complexity – a single classification takes four minutes on a Sun Sparc II. Samaria notes that although an increased recognition rate was achieved the segmentation obtained with the pseudo two-dimensional HMMs appeared quite erratic. Samaria uses the same training and test set sizes as we do (200 training images and 200 test images with no overlap between the two sets). The 5% error rate is the best error rate previously reported for the ORL database that we are aware of.

3.9 System Components

3.9.1 Local Image Sampling

Two different methods of representing local image samples has been evaluated. In each method a window is scanned over the image as shown in figure 3.3.

1. The first method simply creates a vector from a local window on the image using the intensity values at each point in the window. Let x_{ij} be the intensity at the i th column, and the j th row of the given image. If the local window is a square of sides long, centered on, then the vector associated with this window is simply

$$[x_i -, j - w, x_i - w, j - w + 1, \ldots, x_{ij}, \ldots, x_i + w, j + w - 1, x_i + w, j + w]$$

2. The second method creates a representation of the local sample by forming a vector out of a) the intensity of the center pixel x_{ij} and b) the difference in intensity between the center pixel and all other pixels within the square window. The vector is given by

$[x_{ij}-x_i-w, j-w, x_{ij}-x_i-w, j-w+1,\ldots, w_{ij}, x_{ij},\ldots, x_{ij}-x_i+w, j+w-1, x_{ij}-x_i+w, j+w]$
The resulting representation becomes partially invariant to variations in intensity of the complete sample. The degree of invariance can be modified by adjusting the weight w_{ij} connected to the central intensity component.

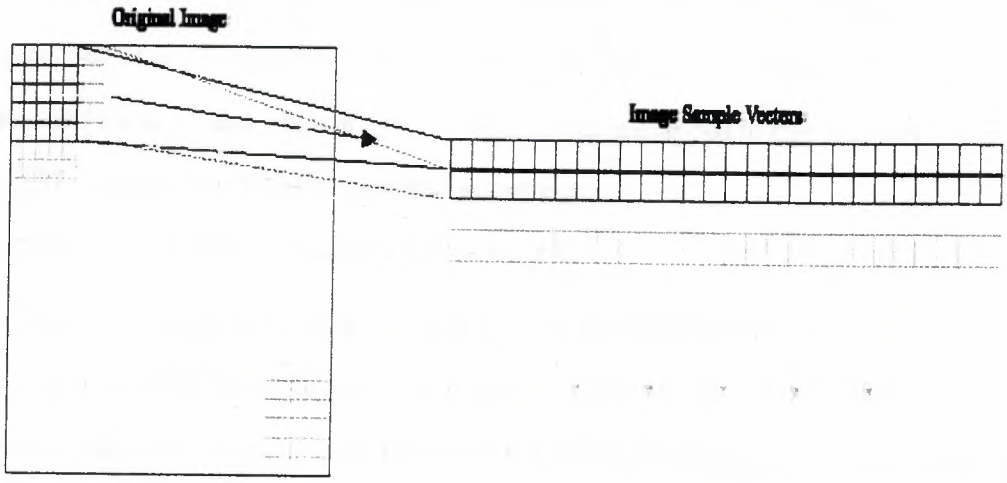


Figure 3.3 A depiction of the local image sampling. A window is stepped over the image and a vector is created at each location.

3.10 The Self-Organizing Map

Maps are an important part of both natural and artificial neural information processing systems . Examples of maps in the nervous system are retinotopic maps in the visual cortex , tonotopic maps in the auditory cortex , and maps from the skin onto the somato sensoric cortex . The self-organizing map, or SOM, introduced by Teuvo Kohonen [1] is an unsupervised learning process which learns the distribution of a set of patterns without any class information. A pattern is projected from an input space to a position in the map – information is coded as the location of an activated node. The SOM is unlike most classification or clustering techniques in that it provides a topological ordering of the classes. Similarity in input patterns is preserved in the output of the process. The topological preservation of the SOM process makes it especially useful in the classification of data which includes a large number of classes. In the local image sample classification, for example, there may be a very large number of classes in which the transition from one class to the next is practically continuous (making it difficult to define hard class boundaries).

3.10.1 Algorithm

The SOM defines a mapping from an input space R^n onto a topologically ordered set of nodes, usually in a lower dimensional space. An example of a two-dimensional SOM is shown in figure 4. A reference vector in the input space, $m_i = [\mu_{i1}, \mu_{i2}, ..., \mu_{in}]^T \varepsilon R^n$,is assigned to each node in the SOM. During training, each input vector,x , is compared to all of the m_i , obtaining the location of the closest match($\|x - m_c\| = \min_i \{\|x - m_i\|$). The input point is mapped to this location in the SOM. Nodes in the SOM are updated according to:

$$m_i(t+1) = m_i(t) + h_{ci}(t)[x(t) - m_i(t)] \tag{3.1}$$

Where t is the time during learning and $h_{ci}(t)$ is the *neighbourhood function*, a smoothing kernel which is maximum at m_c . Usually, $h_{ci}(t) = h(\|r_c - r_i\|, t)$, where r_c and r_i represent the location of the nodes in the SOM output space. r_c c is the node with the closest weight vector to the input sample and r_i i ranges over all nodes. $h_{ci}(t)$ approaches 0 as $\|r_c - r_i\|$ jj increases and also as t approaches ∞ A widely applied neighborhood function is:

$$h_{ci} = \alpha(t) \exp\left(-\frac{\|r_c - r_i\|^2}{2\sigma^2(t)}\right) \tag{3.2}$$

Where $\alpha(t)$ is a scalar valued learning rate and $\sigma(t)$ defines the width of the kernel. They are generally both monotonically decreasing with time. The use of the neighborhood function means that nodes which are topographically close in the SOM structure activate each other to learn something from the same input x. A relaxation or smoothing effect results which leads to a global ordering of the map. Note that $\sigma(t)$ should not be reduced too far as the map will lose its topographical order if neighboring nodes are not updated along with the closest node. The SOM can be considered a non-linear projection of the probability density p(x).

3.10.2 Improving the Basic SOM

The original self-organizing map is computationally expensive due to.

1. In the early stages of learning, many nodes are adjusted in a correlated manner. Luttrel [6] proposed a method which we use that starts by learning in a small

network, and doubles the size of the network periodically during training. When doubling, new nodes are inserted between the current nodes. The weights of the new nodes are set equal to the average of the weights of the immediately neighboring nodes.

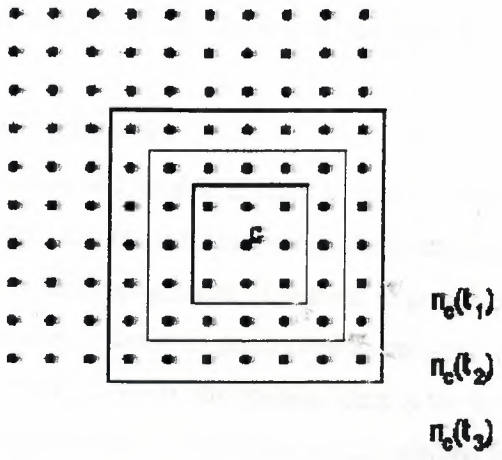


Figure 3.4. A two-dimensional SOM showing a square neighborhood function which starts as $n_c(t_1)$ and reduces in size to $n_c(t_3)$ over time.

2. Each learning pass requires computation of the distance of the current sample to all nodes in the network, which is O(N).However, this may be reduced to O(logN) using a hierarchy of networks which is created from the above node doubling strategy.

3.11 Karhunen-Lo`eve Transform

The optimal linear method6 for reducing redundancy in a dataset is the Karhunen-Lo`eve (KL) transform or eigenvector expansion via Principle Components Analysis (PCA)[6] . PCA generates a set of orthogonal axes of projections known as the principal components, or the eigenvectors, of the input data distribution in the order of decreasing variance. The KL transform is a well known statistical method for feature extraction and multivariate data projection and has been used widely in pattern recognition, signal processing, image processing, and data analysis. Points in an n-dimensional input space are projected into an m-dimensional space, $m \leq n$. The KL transform has been used for comparison with the SOM in the dimensionality reduction

of the local image samples. The use of the KL transform here is not the same as in the eigenfaces approach

because we operate on small local image samples as opposed to the entire images. The KL technique is fundamentally different to the SOM method, as it assumes the images are sufficiently described by second order statistics, while the SOM is an attempt to approximate the probability density as shown in Kohonen [1].

3.12 Convolutional Networks

Theoretically, we should be able to train a large enough multi-layer perceptron neural network to perform any required mapping , including that required to perfectly distinguish the classes in face recognition. However, in practice, such a system is unable to form the required features in order to generalize to unseen inputs (the class of functions which can perfectly classify the training data is too large and it is not easy to constrain the solution to the subset of this class which exhibits good generalization). In other words, the problem is ill-posed - there is not enough training points in the space created by the input images in order to allow accurate approximation of class probabilities throughout the input space. Additionally, there is no invariance to translation or local deformation of the images [23]. Convolutional networks (CN) incorporate constraints and achieve some degree of shift and deformation invariance using three ideas: local receptive fields, shared weights, and spatial sub sampling. The use of shared weights also reduces the number of parameters in the system aiding generalization. .

A typical convolutional network for recognizing characters is shown in figure(3.5) . The network consists of a set of layers each of which contains one or more planes. Approximately centered and normalized images enter at the input layer. Each unit in a plane receives input from a small neighborhood in the planes of the previous layer. The idea of connecting units to local receptive fields dates back to the 1960s with the perceptron and Hubel and Wiesel's [18] discovery of locally sensitive, orientation-selective neurons in the cat's visual system . The weights forming the receptive field for a plane are forced to be equal at all points in the plane. Each plane can be considered as a feature map which has a fixed feature detector that is convolved with a local window which is scanned over the planes in the previous layer. Multiple planes are usually used in each layer so that multiple features can be detected. These layers are called convolutional layers. Once a feature has been detected, its exact location is less

important. Hence, the convolutional layers are typically followed by another layer which does a local averaging and sub sampling operation (e.g. for a sub sampling factor of 2: $y_{ij} = (x_{2i,2j} + x_{2i+1,2j} + x_{2i,2j+1} + x_{2i+1,2j+1})/4$ where y_{ij} is the output of a sub sampling plane at position i, j and x_{ij} is the output of the same plane in the previous layer). The network is trained with the usual back propagation gradient-descent procedure .

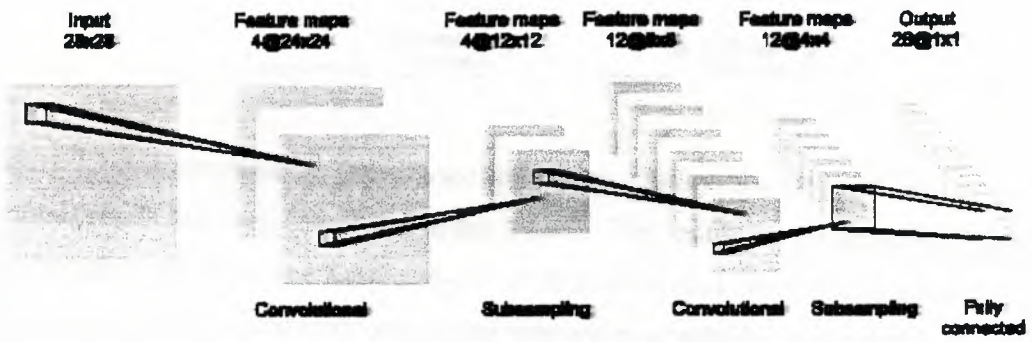


Figure 3.5 A typical convolutional network for recognizing characters.

3.13 System Details

The system we have used for face recognition is a combination of the preceding parts - a high-level block diagram is shown in figure 3.6 and figure 3.7 shows a breakdown of the various subsystems.

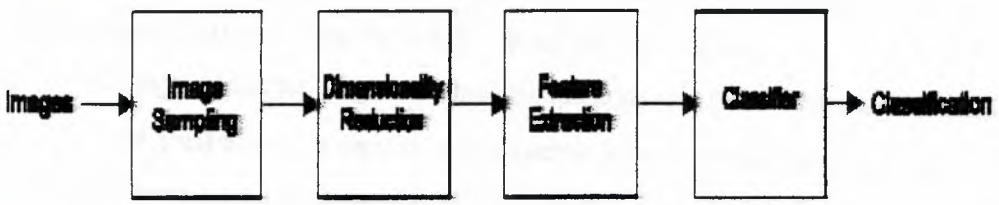


Figure 3.6 A high-level block diagram of the system used for face recognition.

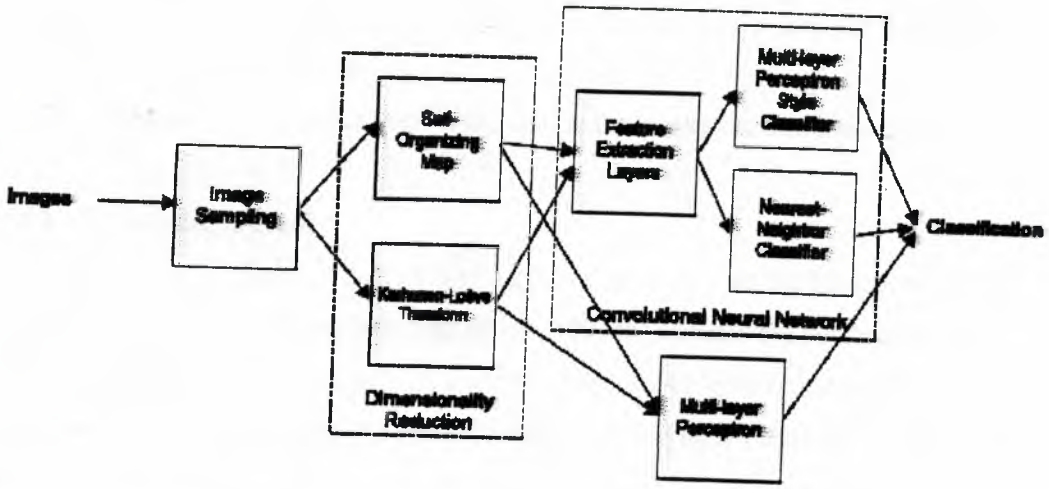


Figure 3.7 A diagram of the system used for face recognition showing alternative methods which had been considered in this chapter. The results are presented with either a self-organizing map or the Karhunen-Lo`eve transform used for dimensionality reduction, and either a convolutional neural network or a multi-layer perceptron for classification. The e possibility of replacing the final classification stage in the convolutional neural network with a nearest neighbor or related classifier had been considered. A complete recognizer consists of only one path through the diagram.

The system works as follows (The complete details of dimensions etc.).

1. For the images in the training set, a fixed size window (e.g. 5x5) is stepped over the entire image as shown in figure (3.3) and local image samples are extracted at each step. At each step the window is moved by 4 pixels.
2. A self-organizing map (e.g. with three dimensions and five nodes per dimension, $5^3 = 125$ total nodes) is trained on the vectors from the previous stage. The SOM quantizes the 25-dimensional input vectors into 125 topologically ordered values. The three dimensions of the SOM can be thought of as three features. We also experimented with replacing the SOM with the Karhunen-Lo`eve transform. In this case, the KL transform projects the vectors in the 25-dimensional space into a 3-dimensional space.
3. The same window as in the first step is stepped over all of the images in the training and test sets. The local image samples are passed through the SOM at each step, thereby creating new training and test sets in the output space created by the self-organizing map. (Each input image is now represented by 3 maps,

each of which corresponds to a dimension in the SOM. The size of these maps is equal to the size of the input image (92x112) divided by the step size (for a step size of 4, the maps are 23x28).)

4. A convolutional neural network is trained on the newly created training set.

3.13.1 Simulation Details

For the SOM, training is split into two phases as recommended by Kohonen [21] - an ordering phase, and a fine-adjustment phase. 100,000 updates are performed in the first phase, and 50,000 in the second. In the first phase, the neighborhood radius starts at two-thirds of the size of the map and reduces linearly to 1. The learning rate during this phase is: $0.7 \times \left(n/N\right)$ where n is the current update number and N is the total number of updates. In the second phase, the neighborhood radius starts at 2 and is reduced to 1. The learning rate during this phase is: $0.02 \times \left(n/N\right)$.

The convolutional network contained five layers excluding the input layer. A confidence measure was calculated for each classification: $y_m(y_m - y_{2m})$ where y_m ym is the maximum output, and y_{2m} is the second maximum output (for outputs which have been transformed using the *softmax* transformation: $y_i = \frac{\exp(u_i)}{\sum_{j=1}^{k} \exp(u_j)}$ where u_i are the original outputs, y_i are the transformed outputs, and k is the number of outputs). The number of planes in each layer, the dimensions of the planes, and the dimensions of the receptive fields are shown in table (3.1). The network was trained with back propagation [13] for a total of 20,000 updates. Weights in the network were updated after each pattern presentation, as opposed to batch update where weights are only updated once per pass through the training set. All inputs were normalized to lie in the range minus one to one. All nodes included a bias input which was part of the optimization process. The best of 10 random weight sets was chosen for the initial parameters of the network by evaluating the performance on the training set. Weights were initialized on a node by node basis as uniformly distributed random numbers in the range $\left(-2.4/F_i, 2.4/F_i\right) where F_i$ is the fan-in of neuron i . Target outputs were -0.8 and 0.8 using the tanh output activation function.

3.14 Experimental Results

All experiments were performed with 5 training images and 5 test images per person for a total of 200 training images and 200 test images. There was no overlap between the training and test sets. A system which guesses the correct answer would be right one out of forty times, giving an error rate of 97.5%. For the following sets of experiments, vary only one parameter in each case. The error bars shown in the graphs represent plus or minus one standard deviation of the distribution of results from a number of simulations9. The constants used in each set of experiments were: number of classes: 40, dimensionality reduction method: SOM, dimensions in the SOM: 3, number of nodes per SOM dimension: 5, texture extraction: original intensity values, training images per class: 5. The constants in each set of experiments may not give the best possible performance as the current best performing system was only obtained as a result of these experiments. The experiments are as follows:

1. *Variation of the number of output classes* – Table (3.2) and figure (3.9) show the error rate of the system as the number of classes is varied from 10 to 20 to 40. No attempt has been made to optimize the system for the smaller numbers of classes. Performance improves with fewer classes to discriminate between (if we continue to add new classes then the chance of a new class being very similar to an existing class increases).
2. *Variation of the dimensionality of the SOM* – Table 3.3 and figure 3.10 show the error rate of the system as the dimension of the self-organizing map is varied from 1 to 4. The best performing value is three dimensions.
3. *Variation of the quantization level of the SOM* – Table 3.4 and figure 3.11 show the error rate of the system as the size of the self-organizing map is varied from 4 to 8 nodes per dimension. The SOM has three dimensions in each case. The best error rate occurs for 8 nodes per dimension. This is also the best error rate of all experiments.

Table 3.1. Dimensions for the convolutional network. The connection percentage refers to the percentage of nodes in the previous layer which each node in the current layer is connected to - a value less than 100% reduces the total number of weights in the network and may improve generalization. The connection strategy used here is similar to that used by Le Cun et al. for character recognition. As an example of how the precise connections can be determined from the table - the size of the first layer planes (21x26) is equal to the total number of ways of positioning a 3x3 receptive field on the input layer planes (23x28).

Layer	Type	Units	x	y	Receptive field x	Receptive field y	Connection Percentage
1	Convolutional	20	21	26	3	3	100
2	Subsampling	20	9	11	2	2	-
3	Convolutional	25	9	11	3	3	30
4	Subsampling	25	5	6	2	2	-
5	Fully connected	40	1	1	5	6	100

Table 3.2. Error rate of the face recognition system with varying number of classes (subjects). Each result is the average of three simulations.

Number of classes	10	20	40
Error rate	1.33%	4.33%	5.75%

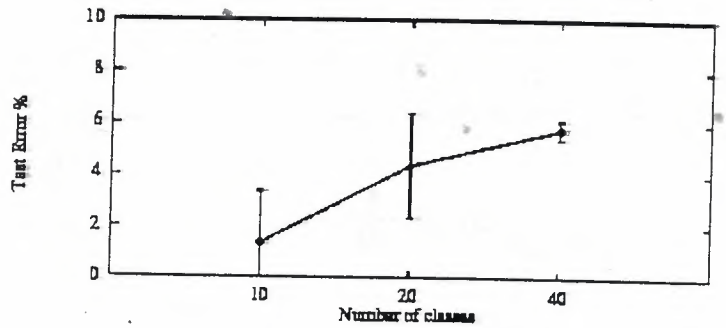


Figure 3.8 The error rate as a function of the number of classes. We did not modify the network from that used for the 40 class case.

Table 3.3 Error rate of the face recognition system with varying number of dimensions in the self-organizing map. Each result given is the average of three simulations.

SOM Dimension	1	2	3	4
Error rate	8.25%	6.75%	5.75%	5.83%

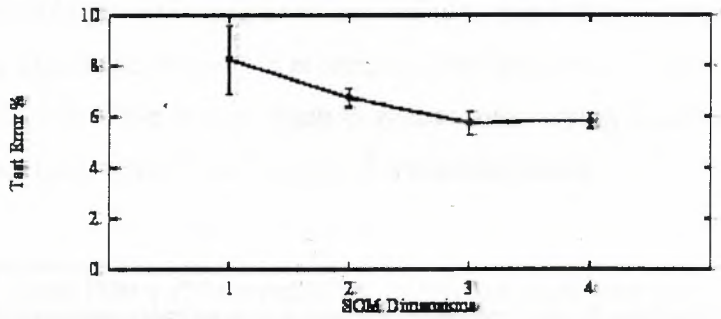


Figure 3.9 The error rate as a function of the number of dimensions in the SOM.

Table 3.4 Error rate of the face recognition system with varying number of nodes per dimension in the self-organizing map. Each result given is the average of three simulations.

SOM Size	4	5	6	7	8
Error rate	8.5%	5.75%	6.0%	5.75%	3.83%

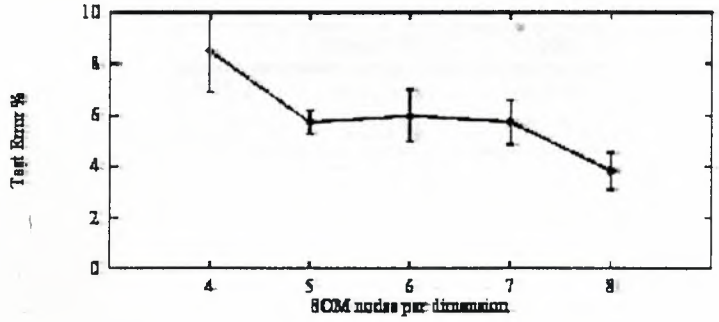


Figure 3.10 The error rate as a function of the number of nodes per dimension in the SOM.

4. *Variation of the texture extraction algorithm* – Table 3.5 shows the result of using the two local image sample representations described earlier. We found that using the original intensity values gave the best performance. We tried altering the weight assigned to the central intensity value in the alternative representation but were unable to improve the results.
5. *Substituting the SOMwith the KL transform* – table 6 shows the results of replacing the self-organizing map with the Karhunen-Lo`eve transform. We tried using the first one, two, or three eigenvectors for projection. Surprisingly, the system performed best with only 1 eigenvector. The best SOM parameters we tried produced slightly better performance. The quantization inherent in the SOM could provide a degree of invariance to minor image sample differences and quantization of the PCA projections may improve performance.

Table 5. Error rate of the face recognition system with varying image sample representation. Each result is the average of three simulations.

Input type	Pixel intensities	Differences w/base intensity
Error rate	5.75%	7.17%

Table 6. Error rate of the face recognition system with linear PCA and SOM feature extraction mechanisms. Each result is the average of three simulations.

Dimensionality reduction	Linear PCA	SOM
Error rate	5.33%	3.83%

Table3.7 Error rate comparison of the various feature extraction and classification methods. Each result is the average of three simulations.

	Linear PCA	SOM
MLP	41.2%	39.6%
CN	5.33%	3.83%

6. *Replacing the CN with an MLP* – Table 3.7 shows the results of replacing the convolutional network with a multi-layer perceptron. Performance is very poor, as we expect due to the loss of shift and deformation invariance. We tried a number of different hidden layer sizes for the multi-layer perceptron in the range 20 to 100. Note that the best performing KL parameters were used while the best performing SOM parameters were not.
7. *The tradeoff between rejection threshold and recognition accuracy* – Figure 3.11 shows a histogram of the recognizer's confidence for the cases when the classifier is correct and when it is wrong for one of the best performing systems. From this graph we expect that classification performance will increase significantly if we reject cases below a certain confidence threshold. Figure 3.12 shows the system performance as the rejection threshold is increased. We can see that by rejecting examples with low confidence we can significantly increase the classification performance of the system. If we consider a system which used a video camera to take a number of pictures over a short period, we could expect that a high performance would be attainable with an appropriate rejection threshold.

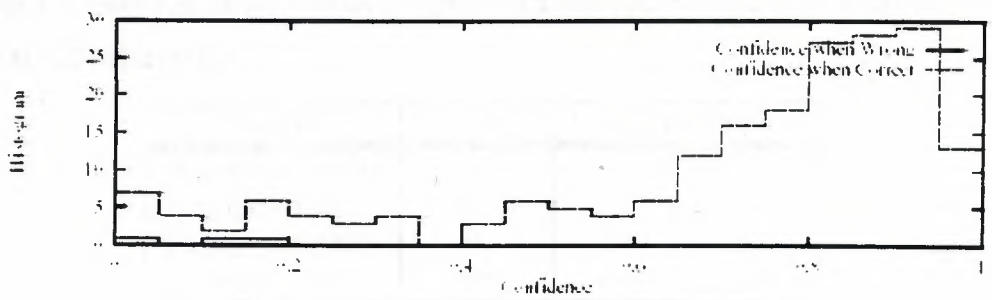


Figure 3.11 A histogram depicting the confidence of the classifier when it turns out to be correct, and the confidence when it is wrong. The graph suggests that we can improve classification performance considerably by rejecting cases where the classifier has a low confidence.

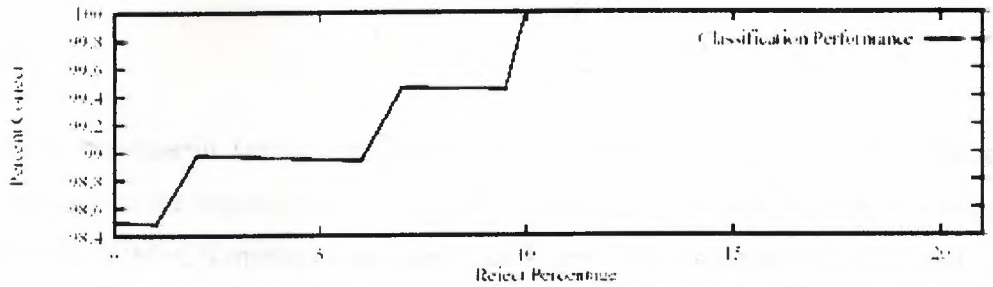


Figure 3.12 The test set classification performance as a function of the percentage of samples rejected. Classification performance can be improved significantly by rejecting cases with low confidence.

8. *Comparison with other known results on the same database* – Table 3.8 shows a summary of the performance of the systems for which we have results using the ORL database. In this case, we used a SOM quantization level of 8. Our system is the best performing system10 and performs recognition roughly 500 times faster than the second best performing system - the pseudo 2D-HMMs of Samaria. Figure 3.13 shows the images which were incorrectly classified for one of the best performing systems.

Table 3.8. Error rate of the various systems. On a Sun Sparc II. On an SGI Indy MIPS R4600 100 MHz system.

System	Error rate	Classification time
Top-down HMM	13%	n.a
Eigenfaces	10.5%	n.a
Pseudo 2D-HMM	5%	240 seconds[1]
SOM+CN	3.8%	< 0.5 seconds[2]

9. *Variation of the number of training images per person.* Table 3.9 shows the results of varying the number of images per class used in the training set from 1 to 5 for PCA+CN, SOM+CN and also for the eigenfaces algorithm. Two versions of the eigenfaces algorithm are implemented - the first version creates vectors for each class in the training set by averaging the results of the eigenface representation over all images for the same person. This corresponds to the algorithm as described by Turk and Pentland [42]. However, that using separate training vectors for each training image resulted in better performance. It has

eigenfaces resulted in similar performance. The PCA+CN and SOM+CN methods are both superior to the eigenfaces technique even when there is only one training image per person. The SOM+CN method consistently performs better than the PCA+CN method.

Figure 3.13 Test images. The images with a thick white border were incorrectly classified by one of the best performing systems.

Table 3.9. Error rate for the eigenfaces algorithm and the SOM+CN as the size of the training set is varied from 1 to 5 images per person. Averaged over two different selections of the training and test sets.

Images per person	1	2	3	4	5
Eigenfaces - average per class	38.6	28.8	28.9	27.1	26
Eigenfaces - one per image	38.6	20.9	18.2	15.4	10.5
PCA+CN	34.2	17.2	13.2	12.1	7.5
SOM+CN	30.0	17.0	11.8	7.1	3.5

Figure 3.14 shows the randomly chosen initial local image samples corresponding to each node in a two dimensional SOM, and the final samples which the SOM converges to. Scanning across the rows and columns we can see that the quantized samples represent smoothly changing shading patterns. This is the initial representation from which successively higher level features are extracted using the convolutional network. Figure 3.15 shows the activation of the nodes in a sample convolutional network for a particular test image.

Using both fixed feature extraction (the representation of local image samples), and a trainable feature extractor (the convolutional network). Can this trainable feature extractor form the optimal set of features? The answer is negative - it is unlikely that the network could extract an optimal set of features for all images. Although the exact process of human face recognition is unknown, there are many features which humans

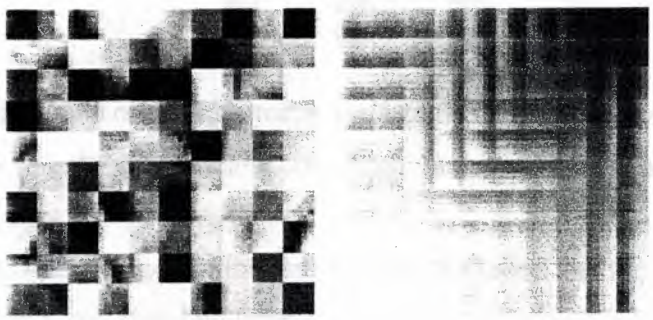


Figure 3.14 SOM image samples before training (a random set of image samples) and after training.

may use but our system is unlikely to discover optimally - e.g. a) knowledge of the three-dimensional structure of the face, b) knowledge of the nose, eyes, mouth, etc., c) generalization to glasses/no glasses, different hair growth, etc., and d) knowledge of facial expressions.

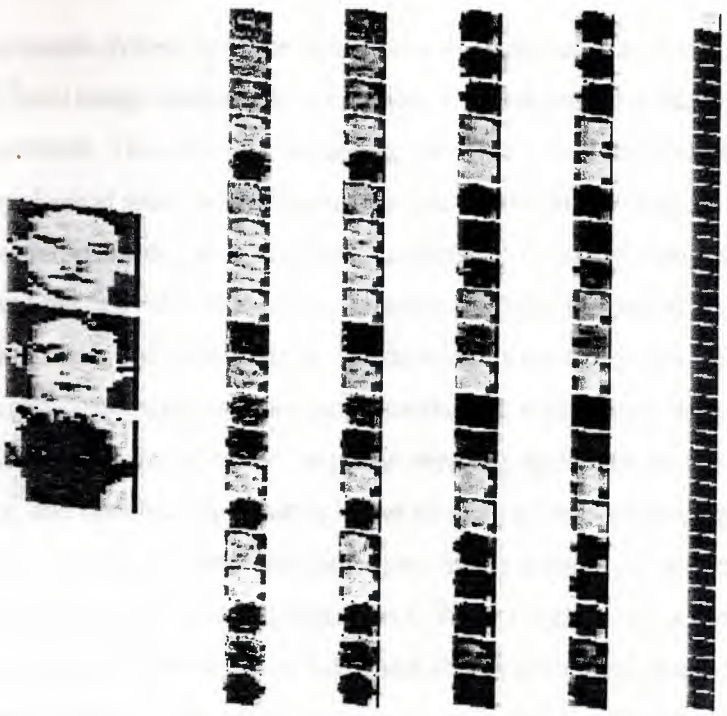


Figure 3.15 A depiction of the node maps in a sample convolutional network showing the activation values for a particular test image. In this case the image is correctly classified with only one activated output node (the top node). From left to right, the layers are: the input layer, convolutional layer 1, sub sampling layer 1, convolutional layer 2, sub sampling layer 2, and the output layer.

3.15 Summary

A fast, automatic system for face recognition is presented which is a combination of a local image sample representation, a self-organizing map network, and a convolutional network. The self-organizing map provides a quantization of the image samples into a topological space where inputs that are nearby in the original space are also nearby in the output space, which results in invariance to minor changes in the image samples, and the convolutional neural network provides for partial invariance to translation, rotation,scale, and deformation. Substitution of the Karhunen-Lo`eve transform for the self-organizing map produced similar but slightly worse results. The method is capable of rapid classification, requires only fast, approximate normalization and preprocessing, and consistently exhibits better classification performance than the eigenfaces approach [42] on the database considered as the number of images per person in the training database is varied from 1 to 5. With 5 images per person the proposed method and eigenfaces result in 3.8% and 10.5% error respectively. The recognizer provides a measure of confidence in its output and classification error approaches zero when rejecting as few as 10% of the examples. Training is computationally expensive (around four hours on a MIPS R4600 100Mhz system), however we have shown that retraining of the complete system may not be required in order to add new classes to the recognizer.

There are no explicit three-dimensional models in the system, however we have found that the quantized local image samples used as input to the convolutional network represent smoothly changing shading patterns. Higher level features are constructed from these building blocks in successive layers of the convolutional network. In comparison with the eigenfaces approach, we believe that the system presented here is able to learn more appropriate features in order to provide improved generalization. The system is partially invariant to changes in the local image samples, scaling, translation and deformation by design.

CHAPTER FOUR

FACE RECOGNITION APPLICATION

4.1 Overview

In this chapter we had focused on biometric applications that give the user some control over data acquisition. These applications recognize subjects from mug shots, passport photos, and scanned fingerprints. Examples not covered include recognition from surveillance photos or from latent fingerprints left at a crime scene. Of the biometrics that meet these constraints, voice, face, and fingerprint systems have undergone the most study and testing.

4.2 The Technology and its Application

In the 1990s, automatic-face-recognition technology moved from the laboratory to the commercial world largely because of the rapid development of the technology, and now many applications use face recognition. These applications include everything from controlling access to secure areas to verifying the identity on a passport.

In the wake of the September 11, 2001 terrorist attacks on America, the security industry is tasked with delivering technologies that could be used to help prevent future terrorist activities. Society is asking for solutions that will foster an efficient and safe travel environment. Our best defenses rest in our ability — within the context of a free and open society — to prevent terrorists and other dangerous individuals from boarding planes in the first place. The events of September 11 call into review our entire airport security system and our attitude towards what societal controls are acceptable from a civil liberties perspective. The ease by which the terrorists had gained access to four planes on September 11, unhindered or challenged, points to fundamental weaknesses in airport security systems in the U.S.

Protecting our airports and preventing a repeat of the September 11 tragedy is a matter of national security. We believe this will require not only a drastic overhaul of the entire security infrastructure of airports, ports of entry and transportation centers in this country, but also around the world. International flights bound to the U.S. could be targets for hijacking and used much in the same way as flights originating and/or operating domestically.

It is unrealistic to expect airlines to be able to address travel security effectively, consistently and on their own. The cost of traditional security is in conflict with bottom-

line interests. Therefore, airport security is a matter that needs to be driven by the federal government, and powered by our intelligence community. It demands substantial financial resources. In every sense, as we have unfortunately learned, airport security is a matter of national defense and should be treated on the same footing as other national defense initiatives. We must emphasize that we are not calling for a national ID system. The threat of terrorism is not solely an internal one. What is needed then, is technology that can be implemented immediately to spot terrorists and prevent their actions.

4.3 Security through Intelligence-Based Identification

The time has come where we must view boarding a plane, not as a right granted to all, but as a privilege accorded to those who can be cleared as having no terrorist or dangerous affiliations. This means that our defenses lie squarely with the ability to properly identify those who pose a threat to our national security and on that basis, deny them free movement.

Biometrics are the only means available to achieve this. They are technologies that conveniently and automatically establish human identity based on a measurable physical characteristic, such as the geometry of the face, hand, the patterns of the finger or the iris of the eye. Biometrics have been under development for more than a decade. Nevertheless, wide scale adoption has in the past been hampered by technical immaturity, hardware cost as well as legitimate concerns over privacy. Today, the technology has reached sufficient levels of maturity and scalability and, by adhering to industry standards for responsible use, can be deployed without posing a threat to our privacy. Because biometrics provide the foundation for security through intelligent identification, they can and should be considered as a key ingredient in the development of a more effective international security framework. [It is important to emphasize that a comprehensive security structure involves many additional elements beyond biometrics. These include other non-biometric technologies as well as training programs and the deployment of trained security officers capable of spotting pathological behavior. Biometrics do not provide the whole solution, they are simply components and tools that must be integrated into an overall system of intelligence and security.]

Biometrics can be used in five key applications related to airport security.

These are:

- Facial Screening and Surveillance
- Automated Biometric-Based Boarding

• Employee Background Screening

• Physical Access Control

• Intelligence Data Mining

The goal must be to improve security, restore public confidence but without creating additional obstacles and hindrances for travelers.

1. Facial Screening and Surveillance

Terrorist organizations rely on indoctrination and training of their members for the effective execution of their mission. This is a process that takes time, which thereby affords the intelligence agencies the opportunity to establish knowledge of identity of membership.

As such, terror is not faceless. Intelligence agencies around the world should be able to build databases of terrorists' faces and identities. These can be used to track them through computerized facial recognition as they travel from country to country.

Facial recognition is most suited because it functions from a distance, in a crowd and in real-time and without subject participation. (Similar databases on gangs have been compiled by domestic authorities and can serve as parallels to the international policing community.)

Facial recognition is ideally suited in this context for:

(a) Facial Screening at Border Control

Preventing terrorists from international travel : Four scenarios.

• Scan for faces of known criminals and terrorists before entering the country: In conjunction with surveillance systems, facial recognition technology can capture faces and match them against intelligence databases, either as passengers are disembarking or while they queue for passport control.

• Prevent the issuance of visas to known terrorists and their affiliates: Facial recognition technology can be used to ensure that visa and other travel documents are only provided if a photo search of the applicant results in no match against intelligence watch lists.

• Prevent the issuance of duplicate identification documents: By searching for duplicate photos in a database, facial recognition technology can be used to determine if subjects apply for travel documents (i.e., passports and visas) multiple times under multiple aliases, and perhaps serve as an early warning signal of future criminal intentions.

• Analyze travel documents for fraud and tampering: Facial recognition technology can be used in conjunction with full-page passport readers that are able to capture the entire image of the passport including the photo of its holder. In addition to ensuring that the passport has not been tampered with, these facial recognition-enabled systems can be used (a) to search the photo against a criminal watch list, (b) to corroborate that the name and photo on the document in fact belong to the same person, and (c) to verify that the identity of the person holding the document matches the identity of the person to whom the document has been issued in the first place.

(b) General Crowd Surveillance

Alerting authorities to the presence of terrorists among large crowds at airports. Facial surveillance can also be used as part of a general security system in conjunction with standard CCTV equipment. Technology such as FaceIt face recognition continuously captures faces from live video and analyzes them by converting the image of the face into an identity specific code known as a face print. The face print is searched against the terrorist watch list; if a match occurs above a certain threshold of confidence, an alarm will sound, alerting security to investigate further. The ability to operate a facial surveillance system over a large network is paramount to implementation. Visionics recently achieved the scalability of facial surveillance in the crowd via completion of the Biometric Network Platform (BNP). This is a hardware platform that plugs into standard CCTV and allows for the addition of facial recognition capability to as many security cameras as desired. It uses the power of network connectivity to build complex distributed security systems. The first such commercially available system is called the FaceIt Argus System, which is a real-time surveillance system capable of handling massive crowds.

2. Biometric-Based Boarding Process:

Preventing terrorists from boarding planes The boarding process must be modified to require an on-the-spot background check on each passenger in addition to the customary proof of identity via a reliable ID document. This could be conducted instantaneously at check-in through facial recognition on all boarding passengers. It is important to point out that the intent is not to identify or track the whereabouts of each passenger, but to check passengers against intelligence databases for possible matches.

The system could work by having check-in terminals equipped with cameras and network connectivity, very similar to point-of-sale terminals deployed for processing credit card transactions. In this case a passenger receives a boarding pass and is allowed to board the plane only after the transaction is authorized by the system (i.e., identity has been verified and no links to criminal organizations have been established).

An important component of this process is reconciling the persons boarding the plane against the passenger manifest, or in other words, to ensure that the person boarding the plane is indeed the same individual who was granted a boarding pass for that particular flight. In deploying this concept, cameras at check-in areas could be linked to cameras at boarding gates, which are themselves linked to flight manifest on a back-end computer network.

3. Screening of Airport Employees:

The integrity of a security system is as good as the integrity of each individual people operating within it. It is therefore imperative that airport employees be subjected to criminal and terrorist background checks prior to their employment. The Airport Security Improvement Act that was enacted into law in November 2000 requires background checks of all airport employees that have access to the tarmac or baggage handling facilities. Most of the Category X airports in the U.S. (the large international airports) are in compliance by now. However, it should be noted that all domestic airports can serve as potential security problems and so it is imperative that this law be modified to include all airports that handle commercial flights. In addition, the following points should be considered:

a) The background checks mandated by the Aviation Security Improvement Act are conducted only against criminal databases (they may not be linked to terrorist data or international criminal databases);

b) The checks involve fingerprint technology, and not photo image searches. Since it is easier for intelligence agencies to capture photos of terrorists clandestinely, this suggests that the background checks mandated by the Aviation Security Improvement Act are insufficient;

c) This requirement for criminal background checks should be expanded to all airport employees.

4. Physical Security

Access to the tarmac, baggage handling and other secure areas at airports must be restricted to authorized personnel. Access control systems today are unreliable since they utilize passwords and tokens (such as cards and badges) that can be lost, stolen or manipulated. Biometric technologies solve this problem by associating access with the measured identity. With biometrics one could be certain that only those authorized access are granted access. The human face, finger, or iris becomes the key that unlocks doors to secure areas.

Related to physical security is the issue of accessing airport/airline computer networks. As stated, biometric technologies associate access with the measured identity. Therefore, these technologies not only prevent unauthorized access to the system in general, but can also prevent access to certain areas on networks and to the entire system at certain times (i.e., employees should only be allowed access to networks during their working shifts).

5. Intelligence Data Mining

Despite all the measures outlined above, perhaps the most critical of all to the effectiveness of an identification based security system is the success of intelligence agencies in developing and maintaining terrorist watch lists in a manner that can be shared across international, federal and individual agency jurisdictions.

As we have said before, terrorism is not faceless and is not without identity. Establishing the identity of terrorists, their collaborators and sympathizers requires ongoing intelligence work. It is clear that a comprehensive program requires an investment in technology as well as in human resources. On the technology front, the development of an "Intelligence Data Mining Infrastructure" is of highest priority. This is an information system that would be capable of gathering, sifting through huge amounts of data and linking information from disparate sources.

These sources may include broadcast video, audio, electronic communications, intercepted messages as well as covert photos and video footage supplied by operatives in the field. The infrastructure should be a platform for an alliance between the world's intelligence agencies that allows for sharing of information across different organizations and jurisdictions. It is this infrastructure that will form the basis for the other components of the comprehensive international security scheme that has been outlined in this document.

4.4 The Biometric Network Platform

The Shield For Protecting Civilization from the Faces of Terror: As stated throughout this document, our ability to identify terrorists in real-time is connected with our ability to access intelligence data from any place, and at any time, and to instantaneously relay that information back to the proper authorities. Visionics' Biometric Network Platform (BNP) makes this possible. It allows for the scalable deployment of facial recognition technology over multiple surveillance systems. It makes it feasible to rapidly, and in an automated manner, use video feeds from an unlimited number of cameras and search all faces against databases created from various intelligence sources and formats, and then alert law enforcement in real-time if a match exists. In essence, this platform becomes our complex defense shield in the fight against terrorism.

The BNP has recently been completed and is currently in deployment phase. In its simplest form, the platform consists of dedicated hardware components that incorporate the functionalities of Visionics' FaceIt face recognition technology. These components are plug-and-play appliances (very much like cable boxes and point of- sale communication boxes) that are connected via a network (public or private internet). The central component in the BNP is a hardware box called FaceGrabber which is able to analyze the continuous video feed from a standard security camera, detect all faces, and "ship" them to other components for further processing (template conversion and processing).

There is no technological limit to how many cameras can be part of the network. For example, cameras at security checkpoints and portals, at the arrivals or departure lounges and at boarding gates can all be linked through the BNP to a central database, making possible each of the large-scale implementation scenarios outlined above. Connected to a FaceGrabber, each camera becomes a web page that ships information, and just like the internet, this web of cameras is scalable, unlimited in scope and provides real-time information. The more comprehensive the camera network and the intelligence data behind it, the more effective our defense shield becomes.

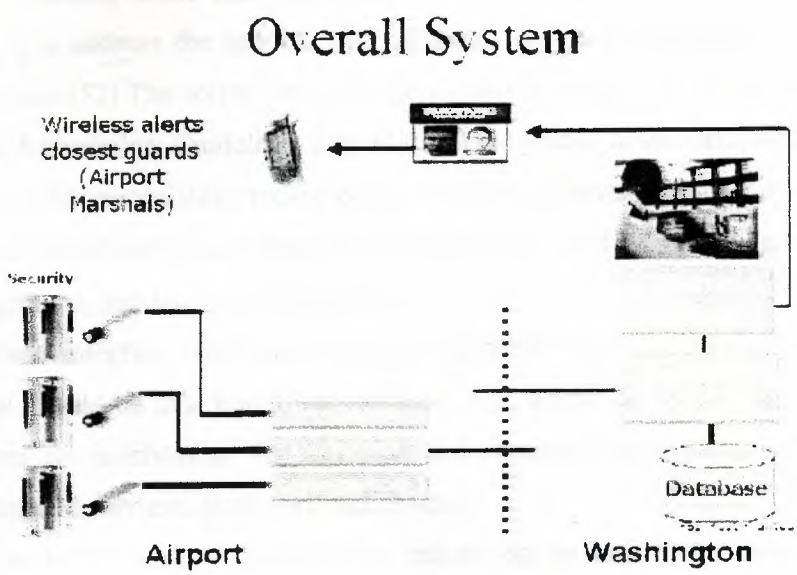


Figure 4.1 Schematic of a simple BNP set-up: Cameras feed into FaceGrabbers which detect faces in video feed. The faces are "shipped" to other FaceIt appliance boxes (BNAs) which convert the faces into 84-byte templates and then via the internet, transferred to the central atabase for matching.

4.5 Implications to Privacy

There is no doubt that an identificationbased security infrastructure using biometrics raises privacy concerns. But we must emphasize that we are not calling for the development of a national ID system. That process would take too long and is truly unnecessary. We believe we can improve our security without giving up our rights to privacy. The key is to ensure responsible use so that systems that are intended for spotting terrorists do not end up being misused down the line for purposes they were not intended for. It is for this reason that the biometrics industry has formulated responsible use guidelines, secured their acceptance by those adopting its technologies, been vigilant in ensuring compliance and, where possible, built technical measures to maintain control over the installations. In fact, the International Biometrics Industry Association (IBIA) was formed in 1998 with the express mission of advocating on behalf of the industry to create responsible use guidelines and public policy. In going a step further,

Visionics itself has recently endorsed a specific set of privacy principles designed to address the specific use of facial recognition technology in surveillance applications.[52] The cornerstone of responsible use policies lies in the following:

• Public Knowledge: Guidelines that establish the proper communication mechanisms to the public that surveillance technologies are in use (in boarding areas and parking lots, etc.) and the circumstances under which exceptions could be made (e.g. undercover investigations, intelligence gathering, etc.).

• Database Integrity: Guidelines must be established for database protocols as to who can and should be in a watch list database (e.g., terrorists, felons, etc.) with particular emphasis on justification for inclusion and removal, valid duration of information, dissemination, review, disclosure and sharing.

• No Match-No Memory: Guidelines to ensure that no audit trail is kept of faces that do not match a known terrorist, affiliate or someone under active investigation. Non-matches should be purged instantly.

• Authorized Operation & Access: Procedures must be established on how to handle respond to and record system alerts (both true and false). Furthermore, technical and physical safeguards such as logon, encryption, control logs, and security to ensure that only authorized trained individuals have access to the system and to the database.

• Enforcement & Penalty: Oversight procedures and penalties for violation of the above principles should be formulated.

There are numerous applications for face recognition technology:

• **Government Use**

- *Law Enforcement.* Minimizing victim trauma by narrowing mug shot searches, verifying identify for court records, and comparing school surveillance camera images to known child molesters.

- *Security/Counterterrorism.* Access control, comparing surveillance images to known terrorists.

- *Immigration.* Rapid progression through Customs.

- *Legislature.* Verify identity of Congressmen prior to vote.

- *Correctional institutions/prisons.* Inmate tracking, employee access.

• **Commercial Use.**

- *Day Care.* Verify identity of individuals picking up the children.

- *Missing Children/Runaways.* Search surveillance images and the internet for missing children and runaways.

- *Gaming Industry.* Find card counters and thieves.
- *Residential Security.* Alert homeowners of approaching personnel.
- *Internet, E-commerce.* Verify identity for Internet purchases.
- *Healthcare.* Minimize fraud by verifying identity.
- *Benefit payments.* Minimize fraud by verifying identity.
- *Voter verification.* Minimize fraud by verifying identity.
- *Banking.* Minimize fraud by verifying identity.

4.6 Summary

Evaluations in general—and technology evaluations in particular—have been instrumental in advancing biometric technology. By continuously raising the performance bar, evaluations encourage progress. Although improving biometric technologies can improve performance, inherent performance limitations remain that are nearly impossible

to work around, except perhaps by combining multiple biometric techniques.

Evaluations typically move from the general to the specific. The first step is to decide which scenarios or applications need to be evaluated. Once the evaluators determine the scenarios, they decide upon the performance measures, design the evaluation protocol, and then collect the data. An example is face recognition systems that verify the identity of a person entering a secure room. The primary purpose of this evaluation type is to determine whether a biometric technology is sufficiently mature to meet performance requirements for a class of applications. Scenario evaluations test complete biometric systems under conditions that model real-world applications. Because each system has its own data acquisition sensor, each system is tested with slightly different data.

CONCLUSION

Face recognition is one of the several approaches for recognizing people. There are several methods that can be used for that purpose. Some of the most common are using Local features or Eigenfaces. Thoughts there are other new techniques more simple to understand use and implement but also with very good performance.

Face recognition technology has come a long way in the last twenty years. Today, machines are able to automatically verify identify information for secure transactions, for surveillance and security tasks, and for access control to buildings. These applications usually work in controlled environments and recognition algorithms that can take advantage of the environmental constraints to obtain high recognition accuracy. However, next generation face recognition systems are going to have wide spread applications in smart environments, where computers and machines are more like helpful assistants. A major factor of that evolution is the use of neural networks in face recognition. A different field of science that also is very fast becoming more and more efficient, popular and helpful to other applications.

The combination of these two fields of science manage to achieve the goal of computers to be able to reliably identify nearby people in a manner that fits naturally within the pattern of normal human interactions. They must not require special interactions and must conform to human intuitions about when recognition is likely. This implies that future smart environments should use the same modalities as humans, and have approximately the same limitations. These goals now appear in reach however, substantial research remains to be done in making person recognition technology work reliably, in widely varying conditions using information from single or multiple modalities.

The importance of face recognition is shown with many applications in which the face recognition is approached, using eigenfaces and local feature analysis we described the work for an automatic system detection, recognition and classification. Also we described how we can perform a face recognition by neural network approach which involves covolutional network and related work and also the system components and system details.

References

[1] Turk, M., and Pentland, A., (1991) Eigenfaces for Recognition, Journal of Cognitive Neuroscience, Vol. 3,No. 1, pp. 71-86.

T. Kohonen, *Self-organization and Associative Memory*, Springer-Verlag, Berlin, 1989.

[2] Pentland, et. al., (1991) Experiments with Eigenfaces, MIT VISMOD TR-194.

[3] Moghaddam, B., and Pentland, A., (1994) Face Recognition using View-Based and Modular Eigenspaces,Automatic Systems for the Identification and Inspection of Humans, SPIE Vol. 2277.

[4] US Patent 5,163,094, Technology Recognition Systems.

[5] A. Pentland, B. Moghaddam, and T. Starner. View-based and modular eigenspaces for face recognition. *IEEE Conference on Computer Vision and Pattern Recognition*, 1994.

[6] L. Sirovich and M. Kirby. Low-dimensional procedure for the characterization of human faces. *Journal of the Optical Society of America*, March 1987

[7] S.E. Palmer, "Hierachical structure in perceptual representation", *Cogn. Psychol.* 9, 441-474, 1977.

[8] N.K. Logothetis; D.L. Sheinberg, "Visual object recognition", *Annu. Rev. Neurosci.* 19, 577-621, 1996.

[9] E. Wachsmuth; M.W. Oram; D.I. Perrett, "Recognition of objects and their component parts: responses of single units in the temporal cortex of the macaque", *Cereb. Cortex* 4, 509-522, 1994.

[10] S.E. Palmer, "Hierachical structure in perceptual representation", *Cogn. Psychol.* 9, 441-474, 1977.

[11] A.J. Colmenarez and T.S. Huang, "Face detection with information-based maximum discrimination", *IEEE CVPR*, 782-787, 1997.

[12] H. Schneiderman, "A statistical approach to 3D object detection applied to faces and cars", Ph.D. thesis, CMURI- TR-00-06, May 2000.

[13] M.C. Burl and P. Perona, "recognition of planar object classes." pp.223-230, CVPR'96.

[14] H. Rowley; S. Baluja; and T. Kanade, "Neural Network- based face detection", *IEEE PAMI.*, 20(1), 23-38, 1998.

[15] P. Viola and M.J. Jones, "Robust real-time object detection", *Technical Report Series, Compaq Cambridge Research Laboratory, CRL* 2001/01, Feb. 2001.

[16] C.P. Papageorgiou; M. Oren and T. Poggio, "A general framework for object detection", ICCV '98.

[17] P. Penev and J. Atick, "Local feature analysis: A general statistical theory for object representation", *Neural Systems*, vol.7, no.3, 477-500, 1996.

[18] P. Comon, Hubel and Wiesel's "Independent component analysis – a new concept?", *Signal Processing*, vol.36, pp.287-314, 1994.

[19] C. Jutten and J. Herault, "Blind separation of sources, part I: An adaptive algorithm based on neuromimetic architecture", *Signal Processing*, vol.24, pp.1-10, 1991.

[20] A.J. Bell and T.J. Sejnowski, "The 'independent components' of natural scenes are edge filters", *Vision Research*, vol.37, pp.3327-3338, 1997.

[21] D. Lee; H.S. Seung, "Learning the parts of objects by non-negative matrix factorization", *Nature* V.401, 21, 788-791, Oct. 1999.

[22] S.Z. Li, X.W. Hou and H.J. Zhang, "Learning Spatially Localized, Parts-Based Representation", *IEEE CVPR*, 2001.

[23] Y. Freund and R.E. Schapire. "A decision-theoretic generalization of no line learning and an application to boosting", *Computational-learning theory: Eurocolt'95* pp23-37, 1995.

[24] http://www.vasc.ri.cmu.edu/idb/html/face/frontal_images/index.html, MIT+CMU frontal face database.

[25] P. Phillips, H. Wechsler, J. Huang, and P. Rauss, "The FERET database and evaluation procedure for face recognition algorithms," *IVC*, vol. 16, no. 5, pp. 295-306, Mar.1998.

[26] D. DeCarlo, D. Metaxas, and M. Stone, "An anthropometric face model using variation techniques," *SIGGRAPH Conf. Proc.*, pp. 67-74, Jul. 1998.

[27] F.I. Parke and K. Waters, "Appendix 1: Three-dimensional muscle model facial animation," *Computer Facial Animation*,A.K. Peters, Sept. 1996.

[28] V. Blanz and T. Vetter, "A morphable model for the synthesis of 3D faces," *SIGGRAPH Conf. Prof.*, pp. 187-194, 1999.

[29] B. Guenter, C. Grimm, D. Wood, H. Malvar, and F. Pighin, "Making faces," *SIGGRAPH Conf.*, pp. 55-66, Jul. 1998.

[30] M.J.T. Reinders, P.J.L. van Beek, B. Sankur, and J.C.A. vander Lubbe, "Facial feature localization and adaptation of ageneric face model for model-based coding," *Signal Processing: Image Comm.*, vol. 7, no. 1, pp. 57-74, Mar. 1995.

[31] W. Lee and N. Magnenat-Thalmann, "Fast head modeling for animation," *Image and Vision Computing (IVC)*, pp. 355-364, vol. 18, no. 4, Mar. 2000.

[32] R. Lengagne, P. Fua, and O. Monga, "3D stereo reconstruction of human faces driven by differential constraints," *IVC*, vol. 18, no.4, pp. 337-343, Mar. 2000.

[33] P. Fua, "Using model-driven bundle-adjustment to model heads from raw video sequences," *Int'l Conf. Computer Vision*, pp. 46-53, Sept. 1999.

[34] Q. Chen and G. Medioni, "Building human face models from two images," *IEEE 2ndWorkshop Multimedia Signal Processing*, pp. 117-122, Dec. 1998.

[35] Z. Zhang, "Image-based modeling of objects and human faces," *Proc. SPIE*, vol. 4309, Jan. 2001.

[36] F.I. Parke and K. Waters, "Appendix 1: Three-dimensional muscle model facial animation," *Computer Facial Animation*,A.K. Peters, Sept. 1996.

[37] Range databases: _http://sampl.eng.ohio-state.edu/_sampl/data/3DDB/RID/minolta/faceimages.0300/_

[38] R.-L. Hsu, M. Abdel-Mottaleb, and A.K. Jain, "Face detection in color images," *Tech. Report MSU-CSE-01-7*, Michigan State Univ., Mar. 2001.

[39] A.A. Amini, T.E. Weymouth, and R.C. Jain, "Using dynamic programming for solving variational problems in vision," *IEEE Trans. PAMI*, vol. 12, pp. 855-867, Sept. 1990.

[40] W.-.S.Hwang and J.Weng, "Hierarchical discriminant regression," *IEEE Trans. PAMI*, vol. 22, pp. 1277-1293, Nov. 2000.

[41] .R. Chellappa, C. Wilson, and S. Sirohev, ``Human and machine recognition of faces: A survey," in *Proceedings of IEEE*, May 1995, vol. 83, pp. 705-740.

[42] 8.P. Phillips, H. Wechsler, J. Huang, and P. Rauss, ``The FERET database and evaluation procedure for face recognition algorithms," *Image and Vision Computing*, vol. 16, no. 5, pp. 295-306, 1998.

[43].L. Wiskott, J-M. Fellous, N. Kruger, and C. von der Malsburg, ``Face recognition by elastic bunch graph matching," *IEEE Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 775-779, 1997.

[44]10.K. Etemad and R. Chellappa, ``Discriminant analysis for recognition of human face images," *Journal of the Optical Society of America*, vol. 14, pp. 1724-1733, 1997.

[45].B. Moghaddam and A. Pentland, ``Probabalistic visual recognition for object recognition," *IEEE Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 696-710, 1997.

[46] .Penev P. and J. Atick, ``Local feature analysis: A general statistical theory for object representation," *Network: Computation in Neural Systems*, vol. 7, pp. 477-500, 1996.

[47] R. Brunelli and T. Poggio. Face recognition: Features versus templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(10):1042–1052, October 1993.

[48] T. Kanade. *Picture Processing by Computer Complex and Recognition of Human Faces.* PhD thesis, Kyoto University, 1973. [19] Hajime Kita and Yoshikazu Nishikawa. Neural network model of tonotopic map formation based on the temporal theory of auditory sensation. In *Proc. WCNN 93, World Congress on Neural Networks*, volume II, pages 413–418, Hillsdale, NJ, 1993. Lawrence Erlbaum.

[49] Ingemar J. Cox, Joumana Ghosn, and Peter N. Yianilos. Feature-based face recognition using mixture-distance. Technical report, NEC Research Institute, Princeton, NJ, October 1995.

[50] R. Brunelli and T. Poggio. Face recognition: Features versus templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(10):1042–1052, October 1993.

[51] Ferdinano Silvestro Samaria. *Face Recognition using Hidden Markov Models.* PhD thesis, Trinity College, University of Cambridge, Cambridge, 1994.

[52] Web site available at http://www.visionics.com/newsroom/biometrics/privacy.html.