FACE RECOGNITION FROM MULTIPOSE IMAGE SEQUENCE

By

M. Nirmala

Reg. No. 71203405009

of

KUMARAGURU COLLEGE OF TECHNOLOGY

COIMBATORE- 641006.

A PROJECT REPORT

Submitted to the

FACULTY OF INFORMATION AND COMMUNICATION ENGINEERING

In partial fulfillment of the requirements

for the award of the degree

of

MASTER OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

June, 2005

BONAFIDE CERTIFICATE

Certified that this project report titled **FACE RECOGNITION FROM MULTIPOSE IMAGE SEQUENCE** is the bonafide work of **Ms. M.Nirmala** who carried out the research under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form part of any other project report of dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

GUIDE **HEAD OF THE DEPARTMENT**

The candidate with University Register No. **71203405009** was examined by us in Project Viva-Voce examination held on 25|6|05.

INTERNAL EXAMINER **EXTERNAL EXAMINER** 25/6/05

ABSTRACT

A novel approach of face recognition based on multipose image sequence involves faces represented by their pattern vectors in eigen space. Instead of recognizing face from a single view, a sequence of images showing face movement from left to right profile is used for recognition. Pattern vectors corresponding to multiple poses build a trajectory in eigen space where each trajectory belongs to one face sequence. New models of matching are presented and analyzed as well as influence of some parameters to recognition ratio.

Two phases are involved in face recognition

- Training phase
- Recognition phase

In the training phase sequences of poses construct prototype trajectories. In recognition phase, an unknown face trajectory is taken into comparison with prototypes.

The method provides a feasible way to locate the different features of the face image for identification. This approach would also help to improve the accuracy of face recognition. The experiments show that the method presented could locate feature points from faces exactly and quickly.

கருத்துச்சுருக்கம்

முக அடையாளம் காணுதல் என்பது பலகோண (Multipose) படிம வரிசைகளை (Image Sequence) அடிப்படையாகக் கொண்டு ஜகன் பரப்பில் (eigenspace) குறிக்கப்படுகிறது. இந்த ஆய்வில் ஒரே ஒரு கோணத்தில் வைத்து முகங்களை அடையாளம் கண்டு கொள்வதற்கு பதிலாக, இடது முதல் வலது வரை(Left to Right Profile) அடுத்தடுத்து வரும் முக உருவங்கள் பயன்படுத்தப்படுகிறது. புதுவிதமான பொருந்தும் முறை அறிமுகப்படுத்தப்பட்டு ஆராய்ச்சி செய்யப்படுகிறது.

முக அடையாளம் காணுதல் என்பது இரு நிலைகளைக் கொண்டது.

- பயிற்சி நிலை
- அடையாளம் காணுதல் நிலை

பயிற்சி நிலையில் அடுத்தடுத்து வரும் முக உருவங்களைக் கொண்டு ஒரு நகல் பாதை (Prototype trajectory) உருவாக்கப்படுகிறது.

அடையாளம் காணும் நிலையில் புது அறியாத முகம் ஒன்றை உருவாக்கப்பட்ட நகல் உருவங்களோடு ஒப்பிடுவதாகும்.

இந்த முறை, முக அடையாளத்தைக் கண்டு கொள்வதற்காக, ஒரு சாத்தியமான வழியில், முக உருவத்தின் பல மாறுபட்ட அம்சங்களை (Features) நிர்ணயிக்கிறது. இவ்வித அணுகுமுறையானது முக அடையாளம் காணுதலை, பிழையின்றி மிகச் சரியாகக் கண்டுபிடிக்க உதவுகிறது. இங்கு அளிக்கப்பட்ட முறை முகத்தின் முக்கிய அம்சங்களை மிகச் சரியாகவும், விரைவாகவும் நிர்ணயிக்கிறது.

ACKNOWLEDGEMENT

I cordially bow my head to the **Omnipotent** for his immense grace at each and every step of my life.

I sincerely thank our principal **Dr.K.K.Padmanabhan Ph.D.,** Kumaraguru College of Technology, Coimbatore, for the support and innumerable facilities provided by him for my project work.

I express my sincere thanks to our Head of the department **Dr.S.Thangasamy Ph.D.,** Department of Computer Science and Engineering, Kumaraguru College of Technology, Coimbatore, for his valuable suggestions and constant encouragement throughout my project.

I am extremely grateful to my guide **Mrs.L.Latha M.E.,** Senior Lecturer, Department of Computer Science and Engineering, Kumaraguru College of Technology, Coimbatore, for her valuable guidance, constant encouragement, timely help, monitoring and above all her humanitarian approach towards me throughout the course of my project.

I express my sincere thanks to our course coordinator **Mrs.L.S.Jayashree M.E.,** Senior Lecturer, Department of Computer Science and Engineering, Kumaraguru College of Technology, Coimbatore, for her motivation, guidance.

I record my sincere thanks to our project coordinator **Mr.R.Dinesh M.S.,** Assistant Professor, Department of Computer Science and Engineering, Kumaraguru College Of Technology , Coimbatore, for his constant encouragement and motivation.

TABLE OF CONTENTS

CONTENTS **PAGE No.**

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1INTRODUCTION

Though people are good at face identification, recognizing human face automatically by computer is very difficult. Face recognition is influenced by many complications, such as the differences of facial expression, the light directions of imaging, and the variety of posture, size and angle. Even to the same people, the images taken in different surroundings may be unlike. The problem is so complicated that the achievement in the field of face recognition by computer is not satisfied as the fingerprints. Facial feature extraction has become an important issue in the recognition of human faces. Detecting the basic features as eyes, nose and mouth exactly is necessary for most of the face recognition methods.

A face model must exhibit invariance under changes in viewing conditions if robust recognition is to be performed. Invariance to changes in illumination, scale, translations and small rotations in the image plane can be achieved through a process of normalization of face images, but changes in face pose cannot be easily normalized. 3D models and 2D model geometric features cannot be extracted robustly under large rotation in depth. Also, it is very difficult to find face features relevant at different poses.

The basic methodology adopted for recognition is largely based on matching of static face image patterns in a given feature space. However, the issue of recognizing faces from sequences of rotating head is largely unresolved. Therefore, a method has analyzed to simulate the effect of variation of some parameters to recognition ratio.

1.2APPLICATIONS OF FACE RECOGNITION

Instead of recognizing face from a single view, a sequence of images showing face movement from left to right profile is used for recognition. The issue of recognizing faces from sequences of rotating head is to be resolved.

The user authentication is increasingly important because security control is required everywhere. Traditionally, ID cards and pass words are popular for authentication. Recently biological authentication technologies across voice, iris, fingerprint, palm print, and face, etc., have been emerged. The face recognition system is economic with low cost of cameras and computers.

The face recognition system has been widely applied in security system,

- Credit card verification
- Criminal identifications
- Teleconference
- Human computer interaction
- Image and film processing

CHAPTER 2

LITERATURE REVIEW

The face recognition system developed so far has been widely classified as belonging to

- Holistic feature category
- Local feature category

Most of the systems in holistic feature category are based on PCA or similar approaches like fisher faces, LDA or PCA combined with LDA. Local feature category mostly consists of systems based on graph matching or dynamic link matching or from derivatives from these.

The face recognition system developed has the following baseline algorithms

- Principal component analysis
- Local feature analysis
- Elastic bunch graph method

2.1 PRINCIPAL COMPONENT ANALYSIS

An attempt to use eigenface representation method is based on principal component analysis PCA, to develop a program capable of face recognition. The main idea in this method is to decompose the face images into a small set of characteristic feature images called eigenfaces, which may be thought of as the principal component of the original image. These eigenface function as the orthogonal basis vector of a linear subspace called face space. Recognition is performed by projecting a new face image into this face space and then comparing its position in the face space with those of known faces.

2.1.1General Information

The central idea of PCA is to reduce the dimensionality of a data set which consists of a large number of interrelated variables, while retaining as much as possible the variation present in the dataset. This is achieved by transforming to a new set of variables, the principal components, which are ordered so that the first retain most of the variation present in all of the original variables.

PCA is a kind of Karonen-Loeve transform which aims to find the orthonormal basis to compress the information. The geometric explanation is that PCA method try to change the coordinate axes so that the ones lie along the directions on which the original dataset has the largest variations.

PCA has two important features that are

- Dimension reduction
- Coordinate transformation

Dimension reduction removes the redundant information to achieve a compact representation of the image data and Coordinate transformation rotates the axis to get a best view angle regarding data variation.

2.1.2Behavioral characteristics

Eigenface method finds the principal components of the distribution of the faces or the eigen vectors of the covariance matrix of the set of face images, treating the image as a point or a vector in a very high dimensional space. The eigen vectors are ordered in such a way that, each one accounts for a different amount of variation among the face images.

Eigen vectors can be thought of as a set of features that together characterize the variation between the face images. Each image location contributes more or less to an eigen vector, so that we can display the eigen vectors as a sort of face, which is called an eigen face. Each eigenface deviates from uniform gray where some facial feature differs among the set of training faces i.e., they are a sort of map variation among the face images.

Each individual face can be represented exactly in terms of a linear combination of the eigenfaces. Each face 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 best M eigenfaces span an M- dimensional subspace – " facespace "- of all possible images.

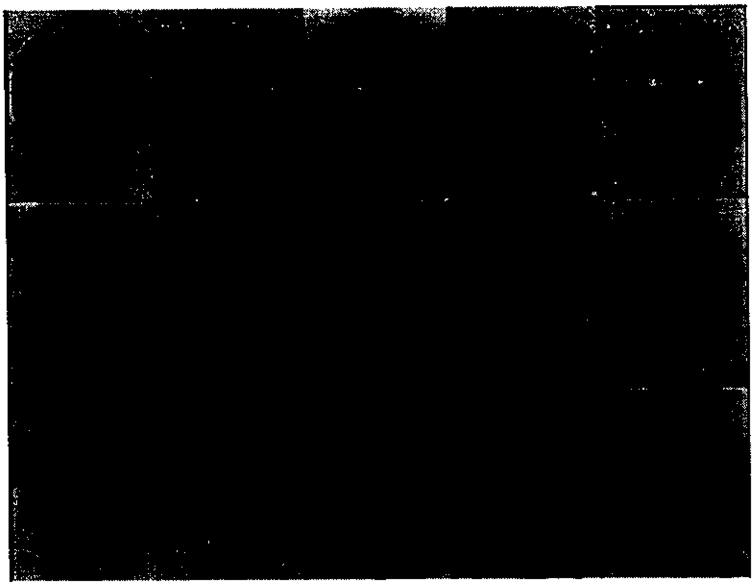


Figure 2.1: Photographs comprising the training set are from ORL database

Consider the set of all possible images, those representing a face, make up only a small fraction of it. If we represent image as very long vectors, each image will be a point in the entire image space. In this image space, faces will group at a certain and separate region from other images since they have similar structures like eyes, nose and mouth. By finding a lower dimensional space, face images can be described in shorter vectors.

In order to efficiently describe the cluster of images, we have to choose the set of directions in the image space along which the variance of the cluster is maximum. This is achieved through the standard procedure of principal component analysis or K-L Transform. The identified directions from the K-L Transform are images, or more precisely eigen -images, and in this case we call as eigenfaces.

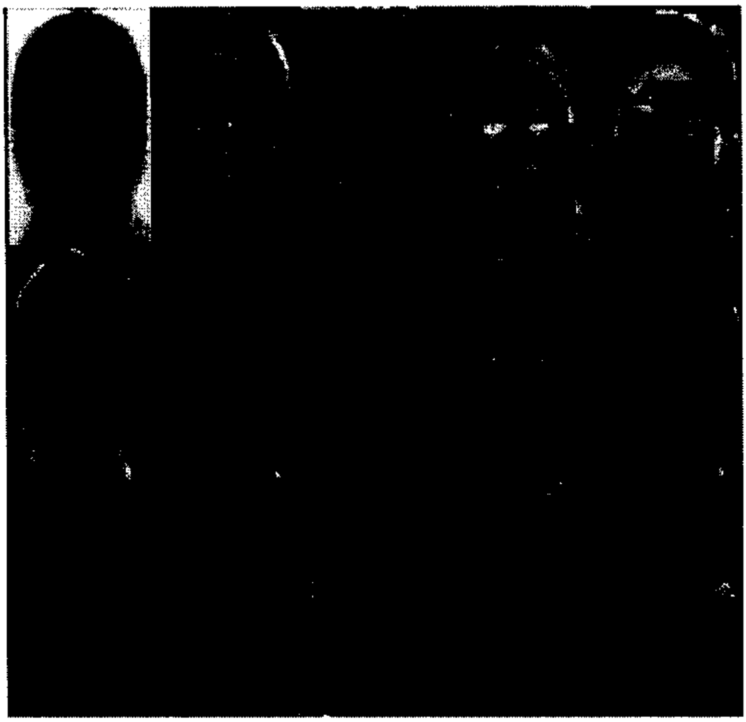
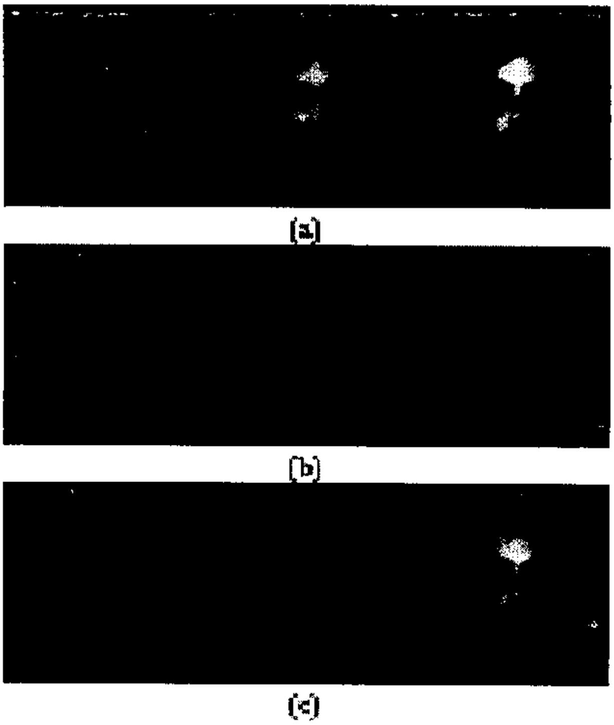


Figure 2.2: Eigenfaces that make up the face space

Recognizing similar faces is equivalent to identifying which is the closest point to the query, in the newly defined face space. If a person is represented in the database more than once, the problem is to decide to which group of images the query is most similar to. Finally if the input image is not a face at all, it's projection into the face space will give inconsistent results.



(a)

(b)

(c)

Figure 2.3: (a) Test views

(b) Eigenface matches

(c) Eigen-feature matches

The figure (a) shows the additional testing views of three individuals. These test images are indicatives of the type of variations which may lead to false matching:

A hand in the face, a painted face, a beard face

Figure (b) shows the nearest match based on a standard eigenface classification. Figure (c) shows the nearest match based on the eyes and nose, and results in correct identification in each case.

2.2Local Feature Analysis

Eigenface is one of the most famous face recognition approach. It is fast, simple and effective method, but not scale and light condition invariant. Another approach represents face as a graph, whose nodes positioned in correspondence to the facial fiducial points is labeled with a multi resolution description of the surrounding gray level image. This method is much better than the other in terms of rotation, light and scale variations.

We build three galleries, each one containing an image per person, the frontal, right and left rotated face galleries. Given a new image, the system localizes the face and facial features, extracts the facial fiducial points, determines the head pose and thus the gallery to be used for the comparison computes the face local characterization and compares the face with the gallery images. We observe that while the facial fiducial point extractor uses color information, the face analysis is done on the gray levels only.

The Rockefeller system, seen in Figure below, uses a sparse version of the eigenface transform, followed by a discriminative neural network. The FERET database testing employs faces with variable position, scale, and lighting in a manner consistent with mug shot or driver's license photography. 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 mug shot-like images before it can be considered potentially competitive.

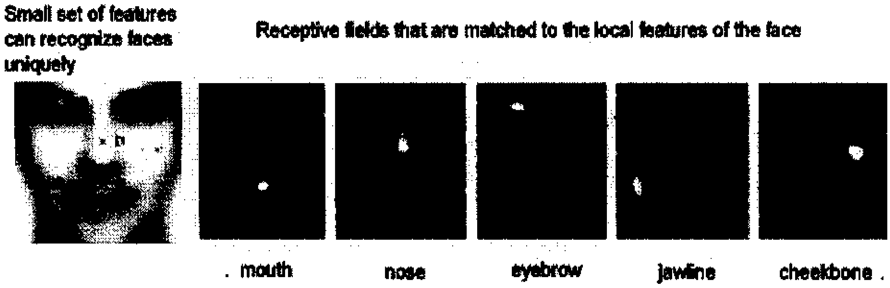

Small set of features can recognize faces uniquely
Receptive fields that are matched to the local features of the face
mouth
nose
eyebrow
jawline
cheekbone

Figure 2.4: Face recognition using Local Feature Analysis

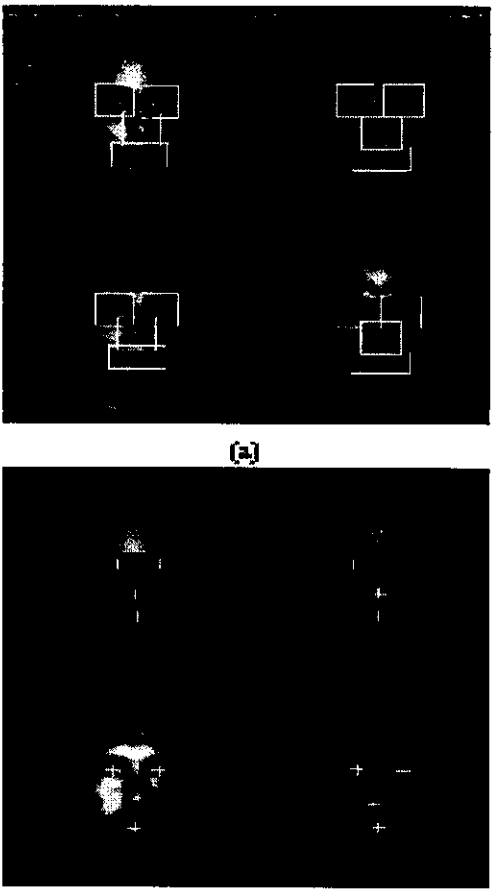
2.2.1Facial feature localization

To localize the eyes we combine two different types of information:

- Eye chromatic characterization
- Binary image obtained

When the two eyes are found we estimate the eventual head tilt determining the tilt of the line passing through the center of the two eye regions.

This allows to straight the head and deal in the next steps with a vertical head. In the straightened head the eyebrow regions are localized upon the eye regions, while the search of mouth and nose is circumscribed to the square under the eyes.



(a)

(b)

Figure 2.5: (a) Examples of facial feature
(b) Resulting typical detections

At first extract the shape information, which allows to localize the features roughly, and then use the color information to tailor the feature sub images more precisely. The chin shape information is extracted applying a non linear edge detector.

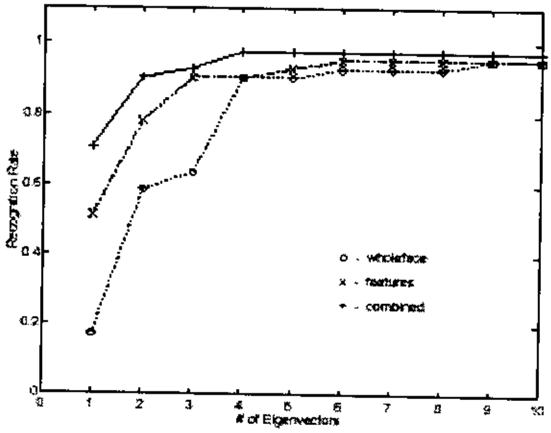


Figure 2.6: Recognition rates for eigenfaces, eigen features and combined representations

The above figure shows the recognition rates for eigenfaces as a function of the number of eigen vectors for eigenface only, eigen feature only and combined representations.

2.2.2Identification of fiducial points

The eyes and the mouth are described by two parametric models derived from the deformable templates with significant variations.

Eyes:

In the eye sub image the iris is first identified with the hough transform for circumferences and the reflux, often present in it, is eliminated. Without these preliminary steps the deformable templates finds very often wrong contours.

Mouth:

In the mouth sub image, we calculate the mouth corners and the entire border adopting a parametric model. To determine the mouth corners, we extract the mouth cut: in correspondence to it, there are high values of the vertical derivative and low values in the color space. Combining this information, we obtain the mouth cut and taking its extremes, the correct corner position. Thus we define the mouth model parameterized by { l, h, $a_{p,}$ a_{ul}, $b_{ul,}$ a_{ur}, b_{ur} } and made of one parabola p, for the lower lips and two cubes, ul and ur, for the upper lips. Two energy functions to be minimized are defined; both of them are functions of the template parameters but the fi:st, E_i, depends on the image colors, while the second Ee depends on the image edges.

Eyebrow:

The Eyebrow description consist in the best parabola which approximates it: we calculate the vertical derivative of the eyebrow gray level image, we binarize it keeping the 10% of the highest derivative values, and we extract the upper border of the obtained regions.

Nose:

The nose is characterized by very simple and generic properties which are true independently of the pose: the nose has a base which gray levels contrast significantly with the neighbor regions. Moreover the nose profile can be characterized as the set of points with the highest symmetry and high luminance values: Finally we say that the nose tip lies on the nose profile, above the nose base line and is bright.

2.3FACE CHARACTERIZATION

2.3.1Pose determination and normalization

Once the fiducial points have been extracted, we normalize the face images scaling them so that the area of triangle defined by the nose tip and the two external eye corners is of 2000 pixels. Moreover to discriminate the images on the basis of the head rotation, we compare the length of the segments which connect the nose tip and the two external eye corners:

If they are approximately of the same length we conclude that the face is frontal: on the contrary the head is left rotated if the length of the segment linking the left eye to the nose is less than 0.8 times the length of the other segment, and vice versa. This information will be used to choose the suitable gallery (frontal, right or left rotated head) for the search.

Thus finally proceed characterizing each fiducial point in terms of the surrounding gray level image adopting two techniques described.

2.3.2Gabor wavelet transform

To characterize each fiducial point we convolve the portion of the gray image around it with a bank of gabor kernels: Applying the Gabor wavelet transform to all the fiducial points, we obtain the face characterization consisting in a jet vector of 10 X N real coefficients where N is the number of fiducial points.

To recognize a face image I, we compute a similarity measure between its jet vector and the ones of all the images G, in the corresponding gallery, and we associate I with G, which maximizes the measure of similarity.

2.3.3Gaussian first derivative basis filters

In this case the filter bank is derived from the Gaussian first derivative basis filters, varying the scale and the orientation:

The jets consists in the real coefficient obtained applying the filters to the gray scale image around each fiducial point. Using these filters, the similarity measure described above has poor performance. It is thus resorted to Euclidean distance between vectors, which of course, has to be minimized.

The face recognition system has been experimented on a database of 50 subjects. For each of them three images are catalogued in the galleries according to the pose. In such images all the fiducial points are visible.

It is tried to recognize frontal, right and left rotated face images referring to the three different galleries. Thus there are three possible situations:

- All the fiducial points are visible
- The eyes and eyebrows are hidden
- The mouth and chin are hidden

The experiments are carried out using both the Gabor wavelet filters and Steerable Gaussian first derivative basis filters.Thus this system, when given a face image extracts the facial fiducial points, determines the head pose, normalizes the image, characterizes it with its jet vector and compares it with the ones in the corresponding gallery. The image is recognized to be the most similar one in the gallery.

2.4ELASTIC BUNCH GRAPH

A system for recognizing human faces from single images out of a large database containing one image per person. Faces are represented by labeled graphs, based on Gabor wavelet transform. Image graphs of new faces are extracted by an elastic graph matching process and can be compared by a simple similarity function. This system is concerned with three aspects:

- Phase information is used for accurate node positioning
- Object adapted graphs are used to handle large rotations in depth
- Image graph extraction is based on a novel data structure called the bunch graph, which is constructed from a small set of sample image graphs.

Individual faces were represented by a rectangular graph, each node labeled with a set of complex gabor wavelet coefficients, called a jet. Only the magnitudes of the wavelet coefficients were used for matching and recognition.

When recognizing the face of a new image, each graph in the model gallery was matched to the image separately and the best match indicated the recognized person. Rotation in depth was compensated for by elastic deformation of the graphs.

There are three major extensions to this system in order to handle large galleries and large variations in pose, and to increase the matching accuracy, which provides the potential for further techniques to improve recognition rate.

- Firstly, use the phase of the complex gabor wavelet coefficients, called a jet to achieve a more accurate location of the nodes and to disambiguate patterns which would be similar in their coefficient magnitudes.
- Secondly, it employs object adapted graphs, so that nodes refer to specific facial landmarks, called fiducial points. The correct correspondence between two faces can then be found across large viewpoint changes.
- Thirdly, it introduces a new data structure, called bunch graph, which serves as a generalized representation of the faces by combining jets of a small set of individual faces.

This allows the system to find the fiducial points in one matching process, which eliminates the need for matching each model graph individually. It reduces computational effort significantly.

2.4.1Jets

A jet is based on a wavelet transform, defined as a convolution of the image with a family of Gabor kernels. A jet J is defined as the set$\{ U_j \}$ of 40 complex Gabor wavelet coefficients obtained for one image point. It can be written as J_j = a_j exp(iö$_j$) with magnitudes $a_j(x)$, which slowly vary with position, and phases ö$_j$(x), which rotate with a rate set by the spatial frequency or wave vector k of the kernels.

Due to this face rotation, jets taken from image points only a few pixels apart have very different coefficients, although representing the same local feature. This can cause severe problems for matching. We therefore either ignore the phase or compensate for its variation explicitly.

2.4.2Graphs

A labeled graph G representing a face consists of N node connected by E edges. The nodes are located at facial landmarks x_n, n=1, 2,...,N, called fiducial points, e.g., the pupils, the corners of the mouth, the tip of the nose, the top and bottom of the ears, etc.

This face graph is object adapted since its geometrical structure is adapted to the structure of the object. The nodes are labeled with jets J. The edges are labeled with two dimensional distance vectors. ÄX$_{\tilde{n}}$ = X_u - $X_{u'}$ where e = 1,2,....E where edge e connects node n' with n. Graphs for different head pose differ in geometry and local features or jets.

In order to extract image graphs automatically for new faces, we need a general representation, rather than models of individual faces. This representation should cover a wide range of possible variations in the appearance of faces, such as differently shaped eyes, mouth or noses, different types of beards, variations due to sex, age and race, etc.

It is obvious that it would be too expensive to cover each feature combination by a separate graph. We instead combine a representative set of M individual model graphs into a stack like structure, called a face bunch graph (FBG). Each model has the same grid structure and the nodes refer to identical fiducial points. A set of jets referring to one fiducial point is called a bunch.

An eye bunch for instance, may include jets from closed, open, female and male eyes, etc. to cover these local variations. The corresponding FBG is then given the same grid structure as the individual graphs, its nodes are labeled with the bunches of jets and its edges are labeled with the averaged distances

$$ÄX_{ñ} = Ó_{m} ÄX_{ñ} / M$$

During the location of fiducial points in a new image of a face, the procedure selects the best fitting jet, called the local expert, from the bunch dedicated to each fiducial point. Thus the full combination of jets in the bunch graph is available, covering a much larger range of facial variation than represented in the constituting model graphs.

2.4.3Elastic Bunch Graph matching

A first set of graphs is generated manually. Nodes are located at fiducial points and edges between the nodes as well as correspondences between nodes of different poses are defined. Once the system has an FBG (possibly consisting of only one manually defined model), graphs for new images can be generated automatically by Elastic Bunch Graph matching.

Initially when the FBG contains only few faces, it is necessaıy to review and correct the resulting matches, but once the FBG is rich enough one can rely on the matching and generate large galleries of the model graphs automatically. It is described in the figure below.

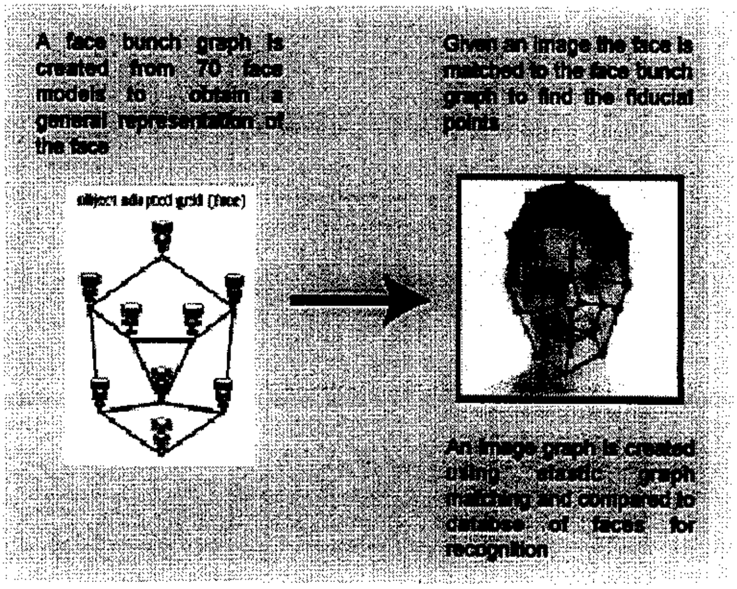

A face bunch graph is created from 70 face models to obtain a general representation of the face
object adapted grid (face)
Given an image the face is matched to the face bunch graph to find the fiducial points
An image graph is created using elastic graph matching and compared to database of faces for recognition

Figure 2.7: Face recognition using Elastic graph matching

Matching a FBG of a new Image is done by maximizing graph similarity between an image graph and the FBG of identical pose. It depends on the jet similarities and a topology term, which takes into account the distortion of the image grid relative to the FBG grid.

Since the FBG provides several jets for each fiducial point, the best one is selected and is used for comparison. These best fitting jets serve as local experts for the image face.

2.4.4Recognition

After having extracted model graphs from the gallery images and image graphs from the probe images recognition is possible with relatively little computational effort by comparing an image graph to all model graphs and selecting the one with the highest similarity value.

The highest similarity value used here for computing graphs is an average over the similarities between pairs of the corresponding jets. Some jets in one pose may not have a corresponding jet in the other pose. A person is recognized correctly if the correct model yields the highest graph similarity. i.e., it is of rank one. A comparison against a gallery of 250 individuals took slightly less than a second.

The recognition rate is very high for frontal against frontal images (first row). This is mainly due to the fact that in the database two frontal views show only little variation and any face recognition system should perform well under these circumstances.

When comparing half profiles with either frontal views or full profiles another reduction in the recognition rate is observed. The results are asymmetrical, performance being better when frontal or profile images serve as model gallery rather than if half profiles are used.

This is mainly due to the fact that both the frontal and profile poses are much more standardized than half profiles for which the angle varies between 40 and 70 degrees . A large number of such false positives considerably degrades the correct recognition rate.

2.5THE INDIVIDUALITY MODEL

Locating faces in images is a key problem in many applications, such as model – based video coding, intelligent computer interfaces and surveillance applications. As such it has to work in a 'natural' environment, i.e., no strict constraints on the position of the camera or the lighting conditions are acceptable.

Often a camera cannot be placed close to the face to be analyzed. Therefore, the technique must be able to work with a moderate resolution, where not all the details of the facial features are recognizable. The analysis is also limited to gray level images, and it had to work with individual frames.

CHAPTER 3

LINE OF ATTACK

There are hybrid approaches to handle the issues of face recognition. These three issues are as follows.

- Feature Extraction
- Discriminant analysis
- Classification

For Feature Extraction, the multi resolution wavelet transform is used to extract wavelet face. It also performs the linear discriminant analysis on wavelet faces to reinforce discriminant power. During classification, the nearest feature plane NFP and nearest feature space NFS classifiers are explored for robust decision in presence of wide facial variations. Their relationship to conventional nearest neighbor and nearest feature line classifiers are also been demonstrated.

3.1SINGLE POSE FACE IDENTIFICATION

Face recognition has a wide application field ranging from the commercial to the security one. Normally galleries are constructed containing images of the individuals. i.e., the frontal, the top and the bottom pose of the person. Thus the database consists of the single pose image of the individual.

However, during recognition the same image has to be got for matching or otherwise an error is reported. Thus from the various experiments, it has been analyzed that the performance yielded is very poor when single pose image is stored in the database. The recognition rate is also very poor when single pose images are used.

3.2 THE PROPOSED MULTIPOSE MODEL

Rather than recognizing face from a single view, a sequence of images showing face movement from left to right profile is used for recognition. Pattern vectors corresponding to multiple poses build a trajectory in eigen space where each trajectory belongs to one face sequence. New models of matching are presented and analyzed as well as influence of some parameters to recognition ratio.

Two phases are involved in face recognition

- Training phase
- Recognition phase

In the training phase sequences of poses construct prototype trajectories. This prototype involves different images corresponding to single individual. The profile includes various images from +90 to -90 degrees. In recognition phase, an unknown face trajectory is taken into comparison with prototypes. The image to be tested is compared to each and every image in the database, and if there exists a match, it is reported or results an error.

If the image contains more than one person, a predefined threshold value will be set for face finding. For each face inside the image, most of the time the code will find multiple face templates for it with small location shift and size change. Finally all the face templates are packed to give only one face for each person in the image.

CHAPTER 4

DETAILS OF METHODOLOGY

4.1FACE PREPROCESSING

The images used is gray scale images, vertically oriented, multiple views. Normal expression variation is allowed and the image is prepared under roughly constant illumination. Because usually images are bigger than the actual faces, the first problem is to find the face in the image, or face detection which is another closely related problem.

The input face should have roughly the same lighting as those in the database. To avoid strong or weak illumination, each face is normalized. The image is treated as a vector in the high dimensional space. Its vector length is adjusted to the vector length of average face in the face space.

While the face detection problem emphasizes the commonality among faces and their difference from non – faces, the interest in face recognition is the face variation among different individuals. All faces have the same facial features and are basically very similar in overall configuration. It makes the face recognition a difficult and fine discrimination problem. Another thing which makes it more complicated is that each individual's face can have many variations because of the change in orientation, expression and lighting. Therefore the input face image to be tested should be normalized.

4.2DYNAMICS OF FACES

So far we have assumed that face images are taken from a very similar view. However, mathematical formulation can be extended to multi views sequence where face is rotating from profile to profile (pose change is between - 90° – 90°).

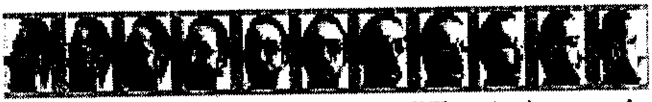


Figure 4.1Face images taken from different view angels (profile to profile).

In this method, two face images of the same person but with large view difference are more likely to be associated with two different persons. This is because that eigen faces do not explicitly register any three dimensional facial structure and images are not differentiated if they are taken from different views. A variance from different persons image are not significantly isolated from the variance from different views. But if we consider the pose information, the problem becomes different. We can form trajectory of patterns for same person. Each point of trajectory corresponds to one view angle of the same person. It is noticeable that, even for very low dimensional features space (only the first 3 PCA features used), the identities of different persons captured by their trajectories are separable.

However, trajectories of same persons lie very close to each other in features space spanned by first 3 PCA Components. Problem of face recognition can be simply matching the face trajectories. For a given sequence containing faces to be recognized, one can obtain a trajectory by projecting the face patterns into PCA space. On the other side, using the known prototype patterns, it is easy to construct trajectory for all known faces.

Therefore, the recognition problem can be solved by matching the novel trajectory to a set of prototype trajectories. To achieve more robust and accurate recognition, it is needed to perform accumulated trajectories. In other word, accumulation of positive identity information will overwhelm any misidentification over time if recognition is performed on accumulated evidence. But it may not be consistently the best match in every frame over time. This idea is illustrated in Figure 4.2 ,4.3.

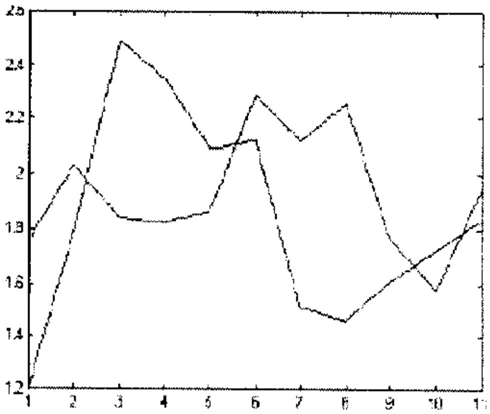


Figure 4.2: Distance measured in each of 11 frames between object patterns

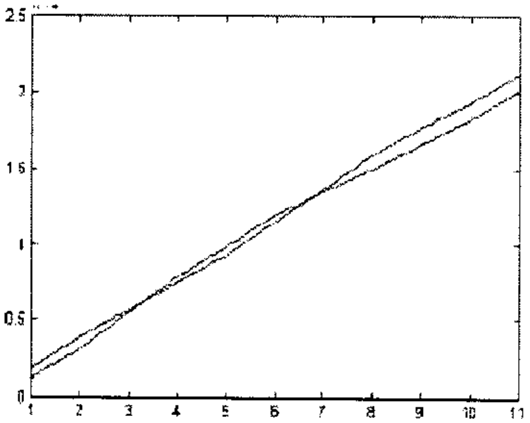


Figure 4.3: Accumulated distance after each of 11 frames between object pattern

Distance can be measured as Euclidean distance in vectors space or some other like city-block distance (also called D4 distance), etc. Therefore, it is important to take into account most suitable distance measure or extend matching criteria to perform robust and most accurate recognition. It is recommended to extend matching criteria for better recognition ratio and present exhaustive analysis of eigen faces number and views number to recognition results. Because of more views used for recognition and robustness of eigen face method, high accurate result is expected.

Face images for recognition are captured under real conditions, including very different face expression, glasses, hair, rotation in image plane, etc. to demonstrate robustness of the eigen faces method and matching criteria as shown in the figure below.

Figure 4.4 : Real conditions effect, (above) sequence used for training,(down) sequence used for recognition

First experiment is made using all of 11 frames of each subject and with varying dimension of features space. It is expected to achieve better recognition ratio when more eigen faces are used.

Table 1. Recognition results of using first n eigenfaces

Eigenfaces (n)	Used frames	Recognition rate (%)
1	11	85.7
2	11	92.8
3	11	96,4
4	11	100
5	11	100
6	11	100
27	11	100

Table 2. Recognition results of using only some frames and 5 eigenfaces

Eigenfaces (n)	Used frames	Recognition rate (%)
3	6 - 6	57.1
3	5 - 7	64.2
3	4 - 8	75.0
3	3 - 9	82.1
3	2 - 10	85.7
3	1 - 11	96.4

Recognition rate is very sensitive with respect to the available poses when it is decreased below 11. For 3 eigen faces approximation, thus, 11 or more frames are desirable. It is also evident from Tables 1,2 that poses close to left and right profiles are less significant for recognition than frontal and half profile poses.

4.3IMPLEMENTATION DETAILS

The notion of direction of variance in a high dimensional space can be extracted from the covariance matrix of the data points. The eigen-vectors of the covariance matrix points in directions of maximum variance of the data and mean square error between original and transformed image is minimized by selecting the eigenvectors associated with largest eigen value. Consider a face image among collection of *M* images, define the average image as:

$$\psi = 1/M \sum i \, \Gamma i., \; i = 1,\ldots,M \qquad (4.1)$$

The covariance matrix of the data is thus defined as:

$$C = 1/M \sum_i \Phi_i \, \Phi_i^T = AA^T, \; i = 1,\ldots,M \quad (4.2)$$

where *C* has dimension *wh x wh* where w is the width of the image and h is the height. The size of this matrix is enormous and computing the *wh* eigenvectors of *C* is computationally hard. However, since we only sum up a finite number of image vectors *M (M<< wh)* the rank of this matrix can not exceed *M-1*. Now, if we consider V_i to be the eigenvectors of matrix A^TA whilst A^TA is only a *M x M* matrix i.e. $(A^TA)V_i = \lambda_i V_i$ (4.3)

then $A(A^TA)V_i = A(\lambda_i V_i)$ (4.4)

which means

$(AA^T)(AV_i) = \lambda i\ (AV_i)$ (4.5)

and AV_i are the eigenvectors of $C = AA^T$. Therefore, the eigenvectors of C are given by:

$Ui = AV_i = \sum kV_k\ {}^i\Phi_k,\ k = 1,\ldots,M$ (4.6)

where ($i = 1,\ldots,M - 1$) and V_k is the *k*th element of *Vi*.

A face image can be represented by its pattern vector and first *M'* eigenfaces of the face space (given by *M'* most significant eigenvectors of covariance matrix *A*). To recognize a novel face, it is needed to calculate distance between its pattern vector and *P* of known face.

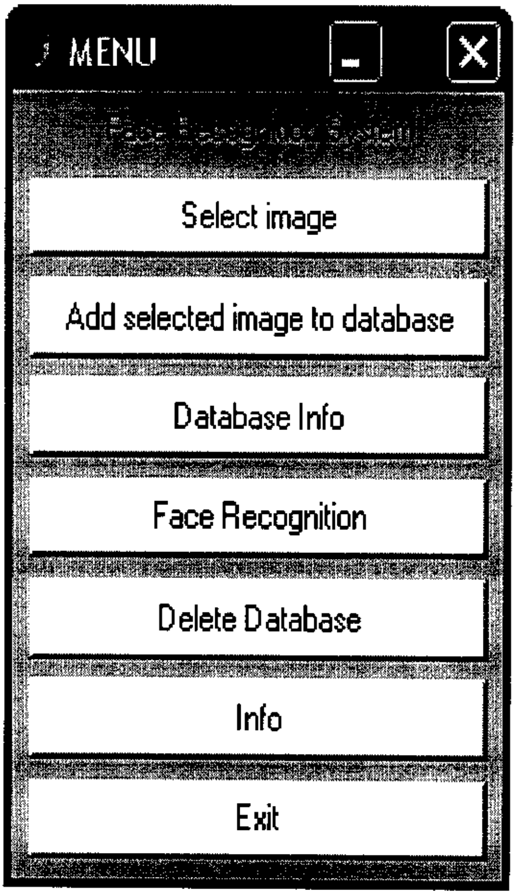
CHAPTER 5

IMPLEMENTATION RESULTS

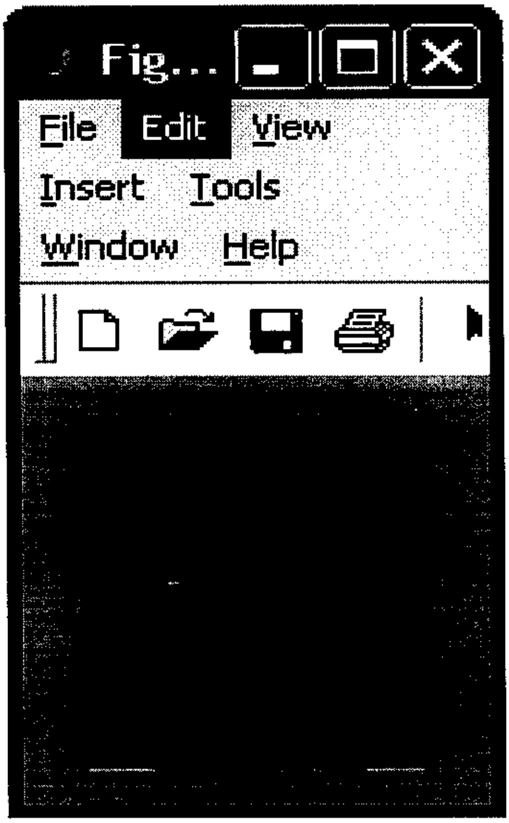
An extensive study of the characteristics of the various features of the face is made. A database of face images is to be created for different individuals. A new input image when supplied to the system is compared with the face images in the database and is reported if a match exists.

MATLAB was used to simulate this method and influence of PCA space dimensionality and pose number to recognition ratio as well as different distance measure. A face recognition system has been implemented for dynamic recognition of face across multi pose sequence by matching trajectories and also to improve the recognition rate. The various snapshots as a result of the experiment undergone are provided.

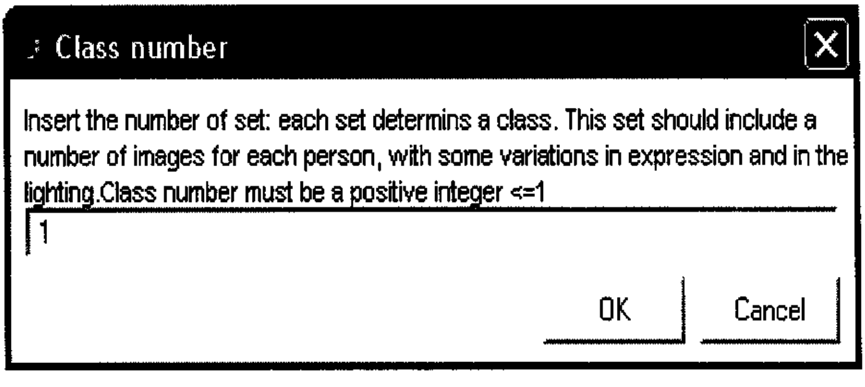
OUTCOME:


MENU
Face Recognition System
Select image
Add selected image to database
Database Info
Face Recognition
Delete Database
Info
Exit

SELECT IMAGE:



ADD SELECTED IMAGE TO DATABASE:

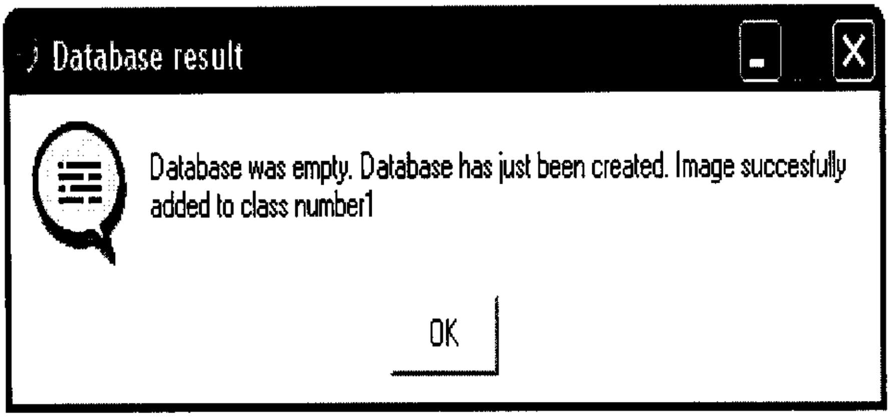


Class number

Insert the number of set: each set determins a class. This set should include a number of images for each person, with some variations in expression and in the lighting.Class number must be a positive integer <=1

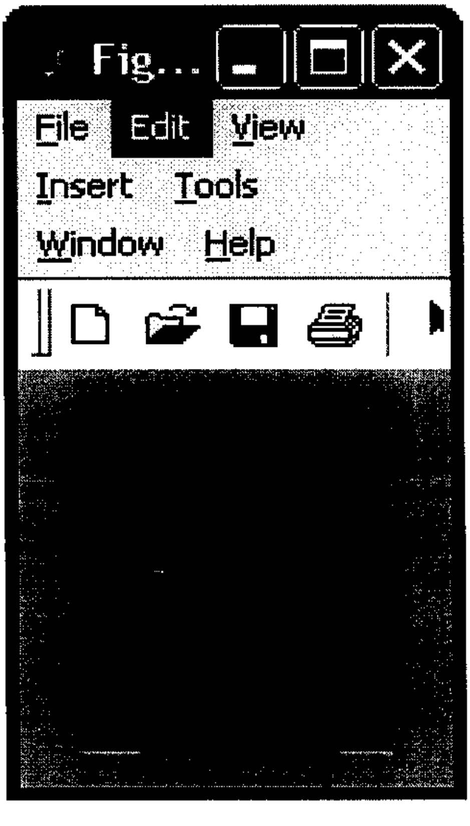
1

OK Cancel

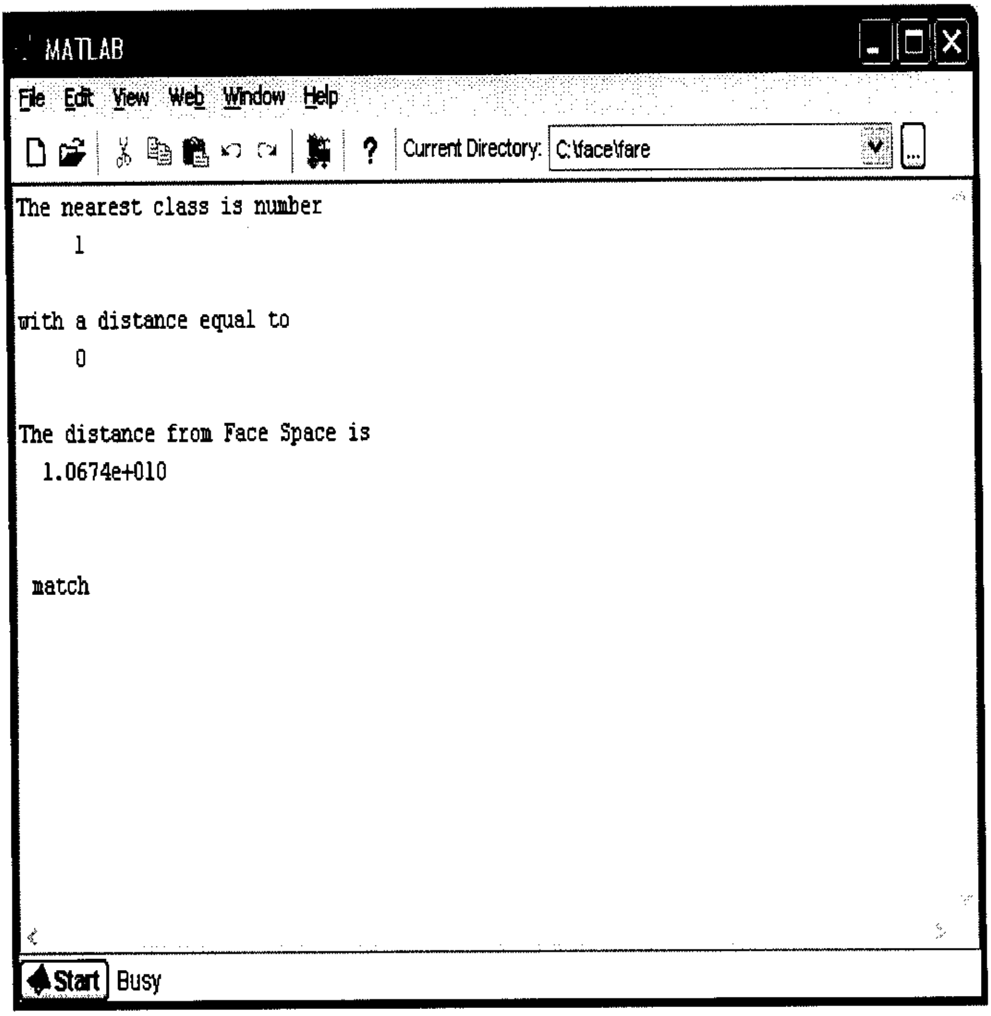


Database was empty. Database has just been created. Image succesfully added to class number1

SELECT IMAGE



FACE RECOGNITION

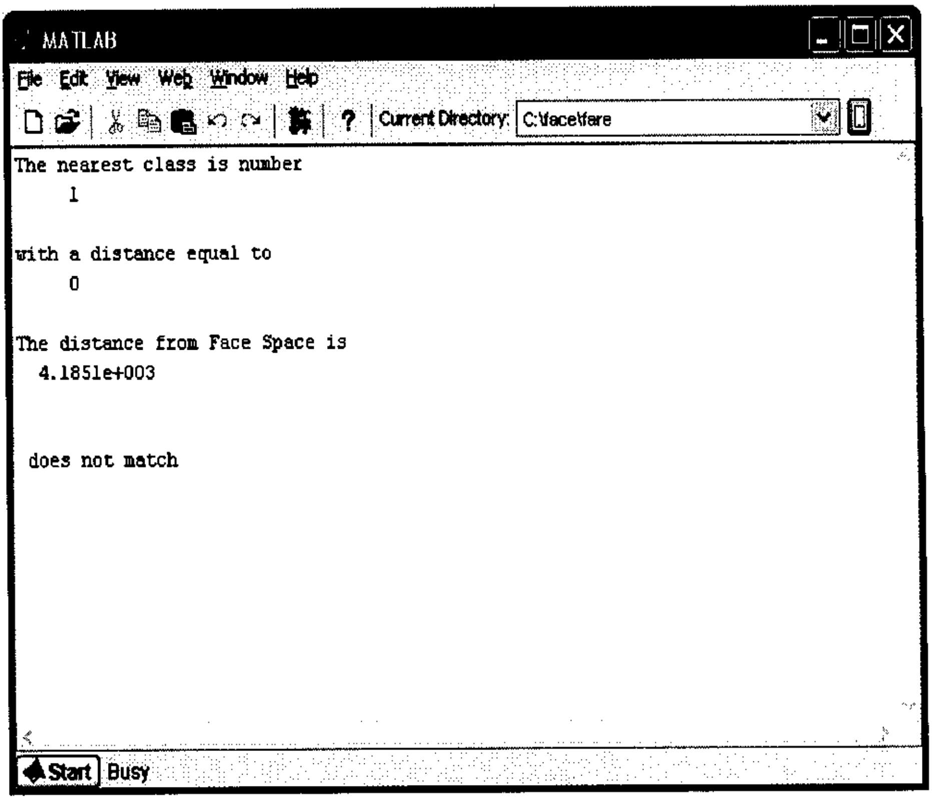


MATLAB
File Edit View Web Window Help
Current Directory: C:\face\fare
Start
Busy

```
The nearest class is number
     1

with a distance equal to
     0

The distance from Face Space is
  1.0674e+010



 match
```

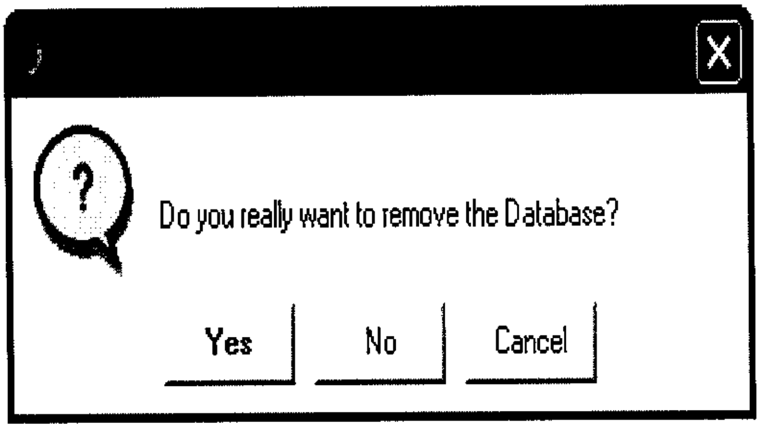
SELECT IMAGE

FACE RECOGNITION

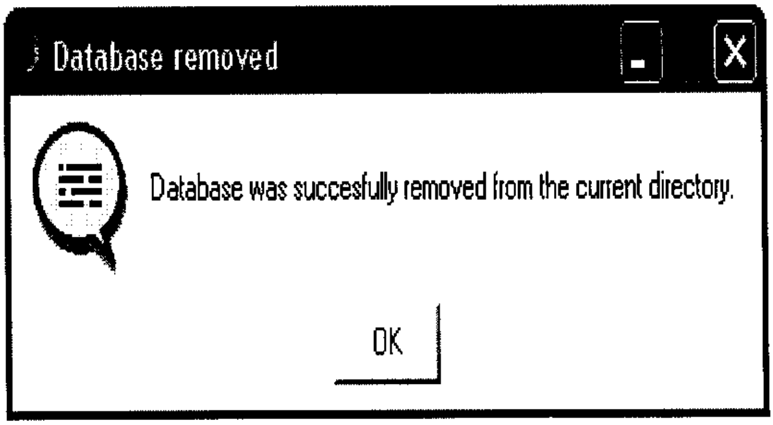

MATLAB
File Edit View Web Window Help
Current Directory: C:\face\fare
Start
Busy

```
The nearest class is number
     1

with a distance equal to
     0

The distance from Face Space is
  4.1851e+003


 does not match
```

DELETE DATABASE:



Do you really want to remove the Database?



Database removed

Database was succesfully removed from the current directory.

CHAPTER 6

CONCLUSION AND FUTURE OUTLOOK

An approach for dynamic recognition of faces across multi-pose sequence by matching trajectories in PCA features space. For a given sequence containing faces to be recognized, we can obtain a trajectory by projecting face patterns into PCA space. For prototype subjects, it is easy to construct identity model trajectories using prototype patterns. Thus, a face recognition problem can be solved by matching the object trajectory to a set of identity model trajectories.

MATLAB was used to simulate this method and influence of PCA space dimensionality and pose number to recognition ratio as well as different distance measure. For eleven or more frames and four or more eigen spaces, this method provide excellent results, but recognition ratio is decreased when either parameter is reduced. This matching criteria yields better results in case of small number of PCA components.

The work could be enhanced by considering the orientation problem. Performing the matching criteria by having the input face image at any orientation, i.e., having the input image upside down has been yet unresolved. Face recognition systems used today work very well under constrained conditions, although all systems work much better with frontal 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.

APPENDIX

SAMPLE CODE:

```
chos=menu('Face Recognition System','Select image','Add
selected image to database','Database Info','Face
Recognition','Delete Database','Info','Exit');
%----------------
if chos==1,
   clc;
   [namefile,pathname]=uigetfile('*.*','Select image');
   if namefile~=0
      [img,map]=imread(strcat(pathname,namefile));
      imshow(img);
   else
      warndlg('Input image must be selected.',' Warning ')
   end
end
if chos==2,
   if exist('img')
      if (exist('face_database.dat')==2)
         load('face_database.dat','-mat');
         face_number=face_number+1;
         data{face_number,1}=img(:);
         prompt={strcat(messaggio,'Class number must be a
         positive integer <= ',num2str(max_class))};
         title='Class number';
```

```
                      lines=1;
                      def={'1'};
                      answer=inputdlg(prompt,title,lines,def);
                      zparameter=double(str2num(char(answer)));
                      if size(zparameter,1)~=0
                          class_number=zparameter(1);

      {class_number<=0)||(class_number>max_class)||(floor(class_numbe
      r)~=class_number)||(~isa(class_number,'double'))||(any(any(imag(cla
      ss_number))))
                              warndlg(strcat('Class number must be a positive
                              integer <= ',num2str(max_class)),' Warning ')
                          else
                              if class_number==max_class;
                                  max_class=class_number+1;
                              end
                              data{face_number,2}=class_number;

      save('face_database.dat','data','face_number','max_class','-append');
msgbox(strcat('Database already exists: image succesfully added to
 class number ',num2str(class_number)),'Database result','help');
                              close all;
                              clear('img')
                          end
                      else
             warndlg(strcat('Class number must be a positive integer <=
             ',num2str(max_class)),' Warning ')
                      end
                  else
                      face_number=1;
```

```
    max_class=1;
    data{face_number,1}=img(:);
    prompt={strcat(messaggio,'Class number must be a
    positive integer <= ',num2str(max_class))};
    title='Class number';
    lines=1;
    def={'1'};
    answer=inputdlg(prompt,title,lines,def);
    zparameter=double(str2num(char(answer)));
    if size(zparameter,1)~=0
     class_number=zparameter(1);
     if
(class_number<=0)||(class_number>max_class)||(floor(class_numbe
r)~=class_number)||(~isa(class_number,'double'))||(any(any(imag(cla
ss_number))))
    warndlg(strcat('Class number must be a positive integer
    <= ',num2str(max_class)),' Warning ')
     else
      max_class=2;
      data{face_number,2}=class_number;

save('face_database.dat','data','face_number','max_class');
msgbox(strcat('Database was empty. Database has just been
created. Image succesfully added to class number
',num2str(class_number)),'Database result','help');
      close all;
      clear('img')
     end
    else
 warndlg(strcat('Class number must be a positive integer <=
',num2str(max_class)),' Warning ')
          end
        end
   else
      errordlg('No image has been selected.','File Error');
   end
end
%----------------
if chos==3,
   clc;
   close all;
   clear('img');
   if (exist('face_database.dat')==2)
      load('face_database.dat','-mat');
   else
      msgbox('Database is empty.','Database result','help');
   end
end
if chos==4,
   clc;
   close all;
   if exist('img')
      ingresso=double(img(:));
      if (exist('face_database.dat')==2)
         load('face_database.dat','-mat');

         matrice=zeros(size(data{1,1},1),face_number);
         for ii=1:face_number
            matrice(:,ii)=double(data{ii,1});
         end
        somma=sum(matrice,2);
        media=somma/face_number;
        for ii=1:face_number
           matrice(:,ii)=matrice(:,ii)-media;
        end
        matrice=matrice/sqrt(face_number);
        [V,D] = eig(elle);

        Vtrue=matrice*V;
        Dtrue=diag(D);
        [Dtrue,ordine]=sort(Dtrue);
        Dtrue=flipud(Dtrue);
        ordine=flipud(ordine);
        Vtrue(:,1:face_number)=Vtrue(:,ordine);
        Vtrue=Vtrue(:,1:max_class-1);
        Dtrue=Dtrue(1:max_class-1);
        pesi=Vtrue'*(ingresso-media);
         pesi_database=zeros(max_class-1,max_class-1);
        numero_elementi_classe=zeros(max_class-1,1);
        for ii=1:face_number
           ingresso_database=double(data{ii,1});
           classe_database=data{ii,2};
           pesi_correnti=Vtrue'*(ingresso_database-media);

pesi_database(:,classe_database)=pesi_database(:,classe_databas
e)+pesi_correnti;

numero_elementi_classe(classe_database)=numero_elementi_class
e(classe_database)+1;
        end
        for ii=1:(max_class-1)

pesi_database_mediati(:,ii)=pesi_database(:,ii)/numero_elementi_cla
sse(ii);
        end
        distanze_pesi=zeros(max_class-1,1);
        for ii=1:(max_class-1)
          distanze_pesi(ii)=norm(pesi-
pesi_database_mediati(:,ii));
        end

[minimo_pesi,posizione_minimo_pesi]=min(distanze_pesi);
        proiezione=zeros(size(data{1,1},1),1);
        for ii=1:(max_class-1)
        proiezione=proiezione+pesi(ii)*Vtrue(:,ii);
        end
        distanza_spazio_facce=norm((ingresso-media)-
proiezione);
        if((minimo_pesi>3e6)|(distanza_spazio_facce<3e9))
          fprintf('\n does not match ');
        else
          fprintf('\n match');
         end


      else
        warndlg('No image processing is possible. Database is
      empty.',' Warning ')
      end
    else
      warndlg('Input image must be selected.',' Warning ')
```

```
   end
if chos==5,
   clc;
   close all;
   if (exist('face_database.dat')==2)
      button = questdlg('Do you really want to remove the
      Database?');
      if strcmp(button,'Yes')
         delete('face_database.dat');
  msgbox('Database was succesfully removed from the current
 directory.','Database removed','help');
      end
   else
      warndlg('Database is empty.',' Warning ')
   end
end
%---------------
```

REFERENCES

[1] Turk, M., and Pentland, A., " Eigen faces for recognition ", Journal of cognitive Neuroscience, 1991., pp.3(1):71-86.

[2] B. moghaddam and A. Pentland, " Face recognition using view – based and modular eigenspaces ", In Automatic systems for the identification and Inspection of humans, SPIE, 1994., volume 2277.

[3] O. Yamaguchi, K. Fukui, and K. Maeda. " Face recognition using temporal image sequence ", In IEEE international Conference on Automatic face and Gesture Recognition, Nara, Japan, 1998., pp 318 – 323.

[4] R. Brunelli and T. Poggio, "Face recognition : Features vs.templates ", IEEE Transactions on Pattern Analysis and Machine Intelligence. October 1993., 15 (10).

[5] I. Craw,E. Ellis, and J. Lishman, "Automatic extraction of face features", Pattern Recognition Letters, 1987.

[6] K. Sung and T.Poggio, " Example – based learning for view – based human face detection ", A. I. Memo No. 1521, Artificial Intelligence Laboratory, Massachusetts, Institute Of Technology, 1994.

[7] David Beymer and T. Poggio, "Face recognition from one example view" A. I. Memo No. 1536, Artificial Intelligence Laboratory, Massachusetts, Institute Of Technology, 1995.

[8] B. S. Manjunath and R. Chellappa, and C. vonder malsburg. " A Feature Based Approach for Face recognition ", In proc. Of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 1992.

[9] A. Pentland and B. Moghaddam, and T. Starner, " View -- based and Modular Eigenspace for Face recognition ",In proc. Of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp 84 - 91, 1994.

[10] S. Y. Kung and J. S. Taur, " Decision – Based Neural Networks with signal / Image Classification applications ", IEEE transactions on Neural Networks, vol 6, pp. 170 – 181, 1995.

[11] A. L. Yuille, P.W. Hallinan, and D. S. Cohen, " Feature Extraction from faces using deformable templates", International journal of computer vision, 8(2): 99 111, 1992.

[12] M. Gargesha and S. Panchanathan , A Hybrid Technique for Facial Feature point Detection", IEEE Proceedings of SSIAI 2002,2003, pp. 134-138.