***TOPIC:***

## FACE DETECTION SYSTEM WITH FACE RECOGNITION

**Submitted by:**

**JOY DAHIYA**

DATE: 06/05/2022

### ABSTRACT

The face is one of the easiest ways to distinguish the individual identity of each other. Face recognition is a personal identification system that uses personal characteristics of a person to identify the person's identity. Human face recognition procedure basically consists of two phases, namely face detection, where this process takes place very rapidly in humans, except under conditions where the object is located at a short distance away, the next is the introduction, which recognize a face as individuals. Stage is then replicated and developed as a model for facial image recognition (face recognition) is one of the much-studied biometrics technology and developed by experts. There are two kinds of methods that are currently popular in developed face recognition pattern namely, Eigenface method and Fisher face method. Facial image recognition Eigenface method is based on the reduction of face dimensional space using Principal Component Analysis (PCA) for facial features. The main purpose of the use of PCA on face recognition using Eigen faces was formed (face space) by finding the eigenvector corresponding to the largest eigenvalue of the face image. The area of this project face detection system with face recognition is Image processing. The software requirements for this project are MATLAB software.

Keywords: face detection, Eigen face, PCA, MATLAB

Extension: There are vast number of applications from this face detection project, this project can be extended that the various parts in the face can be detect which are in various directions.

## INTRODUCTION

Face recognition is the task of identifying an already detected object as a known or unknown face. Often the problem of face recognition is confused with the problem of face detection Face Recognition on the other hand is to decide if the "face" is someone known, or unknown, using for this purpose a database of faces in order to validate this input face.

**FACE RECOGNIZATION:**

DIFFERENT APPROACHES OF FACE RECOGNITION:

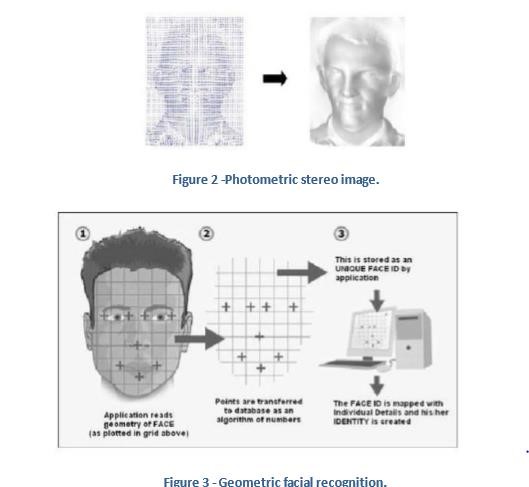
There are two predominant approaches to the face recognition problem: Geometric (feature based) and photometric (view based). As researcher interest in face recognition continued, many different algorithms were developed, three of which have been well studied in face recognition literature.

**Recognition algorithms can be divided into two main approaches:**

1. **Geometric :** Is based on geometrical relationship between facial landmarks, or in other words the spatial configuration of facial features. That means that the main geometrical features of the face such as the eyes, nose and mouth are first located and then faces are classified on the basis of various geometrical distances and angles between features. (Figure 3)
2. **Photometric Stereo:** Used to recover the shape of an object from a number of images taken under different lighting conditions. The shape of the recovered object is defined by a gradient map, which is made up of an array of surface normal (Zhao and Chalupa, 2006) (Figure 2)

**Popular recognition algorithms include:**

* + 1. Principal Component Analysis using Eigenfaces, (PCA)
    2. Linear Discriminate Analysis,
    3. Elastic Bunch Graph Matching using the Fisher face algorithm

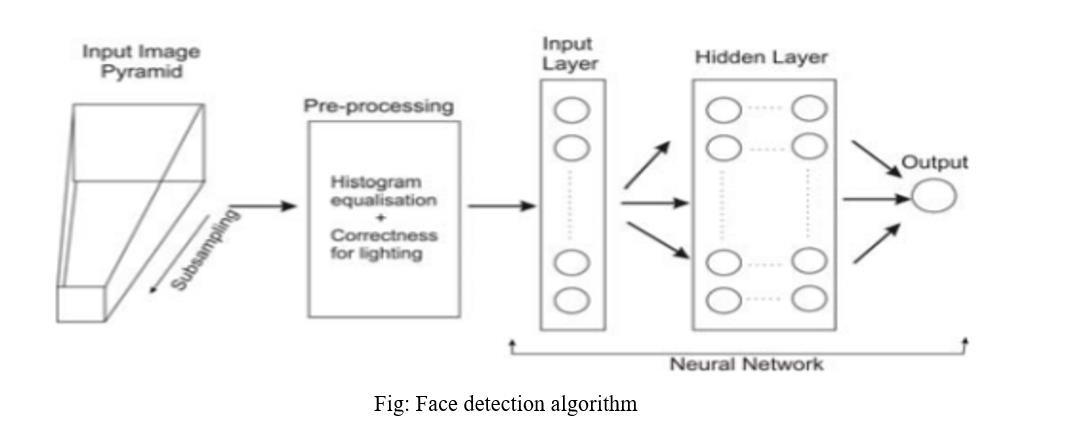


**FACE DETECTION:**

Face detection involves separating image windows into two classes; one containing faces (turning the background (clutter). It is difficult because although commonalities exist between faces, they can vary considerably in terms of age, skin color and facial expression. The problem is further complicated by differing lighting conditions, image qualities and geometries, as well as the possibility of partial occlusion and disguise. An ideal face detector would therefore be able to detect the presence of any face under any set of lighting conditions, upon any background. The face detection task can be broken down into two steps. The first step is a classification task that takes some arbitrary image as input and outputs a binary value of yes or no, indicating whether there are any faces present in the image. The second step is the face localization task that aims to take an image as input and output the location of any face or faces within that image as some bounding box with (x, y, width, height).

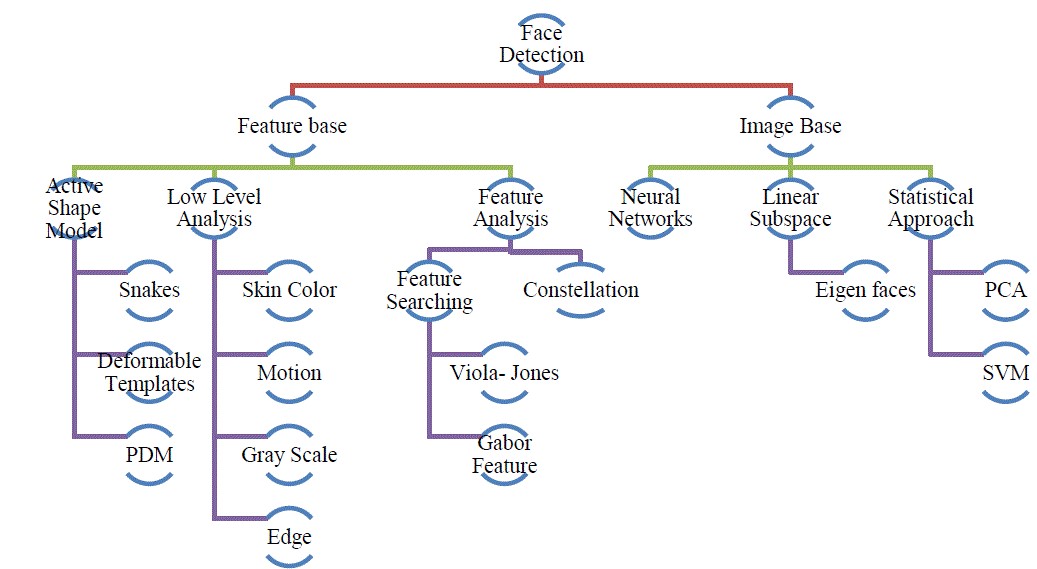
**The face detection system can be divided into the following steps: -**

1. **Pre-Processing:** To reduce the variability in the faces, the images are processed before they are fed into the network. All positive examples that is the face images are obtained by cropping images with frontal faces to include only the front view. All the cropped images are then corrected for lighting through standard algorithms.
2. **Classification:** Neural networks are implemented to classify the images as faces or nonfaces by training on these examples. We use both our implementation of the neural network and the MATLAB neural network toolbox for this task. Different network configurations are experimented with to optimize the results.
3. **Localization:** The trained neural network is then used to search for faces in an image and if present localize them in a bounding box. Various Feature of Face on which the work has done on: - Position Scale Orientation Illumination



## LITERATURE SURVEY

Face detection is a computer technology that determines the location and size of human face in arbitrary (digital) image. The facial features are detected and any other objects like trees, buildings and bodies etc. are ignored from the digital image. It can be regarded as a ‗specific ‘case of object-class detection, where the task is finding the location and sizes of all objects in an image that belong to a given class. Face detection, can be regarded as a more ‗general ‘case of face localization. In face localization, the task is to find the locations and sizes of a known number of faces (usually one). Basically, there are two types of approaches to detect facial part in the given image i.e., feature base and image base approach. Feature base approach tries to extract features of the image and match it against the knowledge of the face features. While image base approach tries to get best match between training and testing images.



**Fig 2.1 detection methods**

**FEATURE BASE APPROCH:**

Active Shape Model Active shape models focus on complex non-rigid features like actual physical and higher-level appearance of features Means that Active Shape Models (ASMs) are aimed at automatically locating landmark points that define the shape of any statistically modelled object in an image. When of facial features such as the eyes, lips, nose, mouth and eyebrows. The training stage of an ASM involves the building of a statistical

1. facial model from a training set containing images with manually annotated landmarks.

ASMs is classified into three groups i.e., snakes, PDM, Deformable templates

1. 1.1) Snakes: The first type uses a generic active contour called snakes, first introduced by Kiss et al. in 1987 Snakes are used to identify head boundaries [8,9,10,11,12]. In order to achieve the task, a snake is first initialized at the proximity around a head boundary. It then locks onto nearby edges and subsequently assume the shape of the head. The evolution of a snake is achieved by minimizing an energy function, Snake (analogy with physical systems), denoted as snake = Internal + ExternalWhere internal and External are internal and external energy functions. Internal energy is the part that depends on the intrinsic properties of the snake and defines its natural evolution. The typical natural evolution in snakes is shrinking or expanding. The external energy counteracts the internal energy and enables the contours to deviate from the natural evolution and eventually assume the shape of nearby features—the head boundary at a state of equilibria. Two main considerations for forming snakes i.e., selection of energy terms and energy minimization. Elastic energy is used commonly as internal energy. Internal energy is varied with the distance between control points on the snake, through which we get contour an elastic-band characteristic that causes it to shrink or expand. On other side external energy relay on image features. Energy minimization process is done by optimization techniques such as the steepest gradient descent. Which needs highest computations. Huang and Chen and Lam and Yan both employ fast iteration methods by greedy algorithms. Snakes have some demerits like contour often becomes trapped onto false image features and another one is that snakes are not suitable in extracting non convex features.

### FEATURE ANALYSIS

These algorithms aim to find structural features that exist even when the pose, viewpoint, or lighting conditions vary, and then use these to locate faces. These methods are designed mainly for face localization

#### Feature Searching

Paul Viola and Michael Jones presented an approach for object detection which minimizes computation time while achieving high detection accuracy. Paul Viola and Michael Jones [39] proposed a fast and robust method for face detection which is 15 times quicker than any technique at the time of release with 95% accuracy at around 17 fps. The technique relies on the use of simple Har-like features that are evaluated quickly through the use of a new image representation. Based on the concept of an ―Integral Image‖ it generates a large set of features and uses the boosting algorithm AdaBoost to reduce the overcomplete set and the introduction of a degenerative tree of the boosted classifiers provides for robust and fast interferences. The detector is applied in a scanning fashion and used on gray-scale images, the scanned window that is applied can also be scaled, as well as the features evaluated.

### CONSTELLATION METHOD

All methods discussed so far are able to track faces but still some issue like locating faces of various poses in complex background is truly difficult. To reduce this difficulty investigator, form a group of facial features in face-like constellations using more robust modelling approaches such as statistical analysis. Various types of face constellations have been proposed by Burl et al. They establish use of statistical shape theory on the features detected from a multiscale Gaussian derivative filter. Huang et al. also apply a Gaussian filter for pre-processing in a framework based on image feature analysis. Image Base Approach.

## DIGITAL IMAGE PROCESSING

### DIGITAL IMAGE PROCESSING

**Interest in digital image processing methods stems from two principal application areas:**

1. Improvement of pictorial information for human interpretation
2. Processing of scene data for autonomous machine perception

In this second application area, interest focuses on procedures for extracting image information in a form suitable for computer processing.

Examples includes automatic character recognition, industrial machine vision for product assembly and inspection, military recognizance, automatic processing of fingerprints etc.

**Image:**

Am image referring a 2D light intensity function f (x, y), where (x, y) denotes spatial coordinates and the value of f at any point (x, y) is proportional to the brightness or gray levels of the image at that point. A digital image is an image f (x, y) that has been discretized both in spatial coordinates and brightness. The elements of such a digital array are called image elements or pixels.

**A simple image model:**

To be suitable for computer processing, an image f (x, y) must be digitalized both spatially and in amplitude. Digitization of the spatial coordinates (x, y) is called image sampling. Amplitude digitization is called gray-level quantization.

The storage and processing requirements increase rapidly with the spatial resolution and the number of gray levels.

Example: A 256 gray-level image of size 256x256 occupies 64k bytes of memory.

**Types of image processing**

* Low level processing
* Medium level processing
* High level processing

Low level processing means performing basic operations on images such as reading an image resize, resize, image rotate, RGB to gray level conversion, histogram equalization etc…, The output image obtained after low level processing is raw image. Medium level processing means extracting regions of interest from output of low-level processed image. Medium level processing deals with identification of boundaries i.e.,edges. This process is called segmentation. High level processing deals with adding of artificial intelligence to medium level processed signal.

### FUNDAMENTAL STEPS IN IMAGE PROCESSING

**Fundamental steps in image processing are**

1. Image acquisition: to acquire a digital image
2. Image pre-processing: to improve the image in ways that increases the chances for success of the other processes.
3. Image segmentation: to partitions an input image into its constituent parts of objects.
4. Image segmentation: to convert the input data to a from suitable for computer processing.
5. Image description: to extract the features that result in some quantitative information of interest of features that are basic for differentiating one class of objects from another.
6. Image recognition: to assign a label to an object based on the information provided by its description.

problem

Knowledge base

Pre

-

processing

Image

acquisition

Segmentation

Representation

and description

Recognition

And

fig.3.1. Fundamental steps in digital image processing

### ELEMENTS OF DIGITAL IMAGE PROCESSING SYSTEMS

A digital image processing system contains the following blocks as shown in the figure

Communication channel

l

Storage

•

Optical discs

•

Tape

•

Video tape

•

Magnetic discs

Processing Unit

•

Computer

•

Work station

Image

acquisition

equipment’s

•

Video

•

Scanner

Display unit

•

TV monitors

•

Printers

•

Projectors

Fig.3.3. Elements of digital image processing systems

The basic operations performed in a digital image processing system include

1. Acquisition
2. Storage
3. Processing
4. Communication
5. Display

#### A simple image formation model

Image is denoted by two-dimensional function f (x, y). f (x, y) may be characterized by 2 components:

1. The amount of source illumination I (x, y) incident on the scene
2. The amount of illumination reflected r (x, y) by the objects of the scene
3. f (x, y) = I (x, y) r (x, y), where 0 <I(XXY) <and 0<r (x, y) < 1

**Typical values of reflectance r (x, y):**

* + 0.01 for black velvet
  + 0.65 for stainless steel
  + 0.8 for flat white wall paint
  + 0.9 for silver-plated metal
  + 0.93 for snow Example of typical ranges of illumination I (x, y) for visible light

(Average values)

* + Sun on a clear day: ~90,000 lm/m^2, down to 10,000lm/m^2 on a cloudy day
  + Full moon on a clear evening: -0.1 lm/m^2
  + Typical illumination level in a commercial office. ~1000lm/m^2 image Formats (supported by MATLAB Image Processing Toolbox)

|  |  |  |  |
| --- | --- | --- | --- |
| Format name | Full name | Description | Recognized extensions |
| TIFF | Tagged Image File  Format | A flexible file format supporting a variety image compression standard including  JPEG | .tiff, .tiff |
| JPEG | Joint Photographic  Experts Group | A standard for compression of images of photographic quality | .jpg, .jpeg |
| GIF | Graphics  Interchange Format | Frequently used to make small animations  on the internet | .gif |
| BMP | Windows Bitmap | Format used mainly for simple uncompressed images | .bmp |
| PNG | Portable Network  Graphics | Compresses full color images with transparency (up to  48bits/p | . pang |

Table.3.3. Image Formats Supported By MATLAB

## FACE DETECTION

The problem of face recognition is all about face detection. This is a fact that seems quite bizarre to new researchers in this area. However, before face recognition is possible, one must be able to reliably find a face and its landmarks. This is essentially a segmentation problem and in practical systems, most of the effort goes into solving this task. In fact, the actual recognition based on features extracted from these facial landmarks is only a minor last step.

There are two types of face detection problems:

1. Face detection in images and
2. Real-time face detection

### FACE DETECTION IN IMAGES

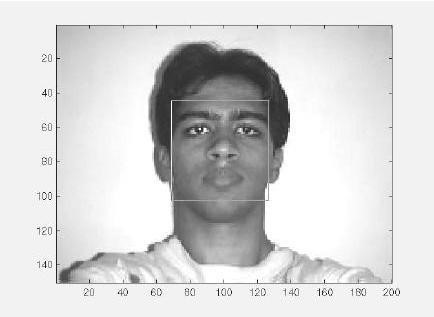


Figure 5.1 A successful face detection in an image with a frontal view of a human face.

Most face detection systems attempt to extract a fraction of the whole face, thereby eliminating most of the background and other areas of an individual's head such as hair that are not necessary for the face recognition task. With static images, this is often done by running a across the image. The face detection system then judges if a face is present inside the window (Brunelle and Poggi, 1993). Unfortunately, with static images there is a very large search space of possible locations of a face in an image

Most face detection systems use an example-based learning approach to decide whether or not a face is present in the *window* at that given instant (Sung and Poggio,1994 and sung,1995). A neural network or some other classifier is trained using supervised learning with 'face' and 'nonface' examples, thereby enabling it to classify an image (*window* in face detection system) as a 'face' or 'non-face’. Unfortunately, while it is relatively easy to find face examples, how would one find a representative sample of images which represent non-faces (Rowley et al., 1996)? Therefore, face detection systems using example-based learning need thousands of 'faces' and 'nonface' images for effective training. Rowley, Baluga, and Kaneda (Rowley et al.,1996) used 1025 face images and 8000 non-face images (generated from 146,212,178 sub-images) for their training set!

There is another technique for determining whether there is a face inside the face detection system's *window -* using Template Matching. The difference between a fixed target pattern (face) and the window is computed and thresholder. If the window contains a pattern which is close to the target pattern(face) then the window is judged as containing a face. An implementation of template matching called Correlation Templates uses a whole bank of fixed sized templates to detect facial features in an image (Bichsel, 1991 &Brunelle and Poggi, 1993). By using several templates of different (fixed) sizes, faces of different scales (sizes) are detected. The other implementation of template matching is using a deformable template (Yuille, 1992). Instead of using several fixed size templates, we use a deformable template (which is non-rigid) and there by change the size of the template hoping to detect a face in an image.

A face detection scheme that is related to template matching is image invariants. Here the fact that the local ordinal structure of brightness distribution of a face remains largely unchanged under different illumination conditions (Sinha, 1994) is used to construct a spatial template of the face which closely corresponds to facial features. In other words, the average grey-scale intensities in human faces are used as a basis for face detection. For example, almost always an individual’s eye region is darker than his forehead or nose. Therefore, an image will match the template if it satisfies the 'darker than' and 'brighter than' relationships (Sung and Poggi, 1994).

### REAL-TIME FACE DETECTION

Real-time face detection involves detection of a face from a series of frames from a video capturing device. While the hardware requirements for such a system are far more stringent, from a computer vision stand point, real-time face detection is actually a far simpler process than detecting a face in a static image. This is because unlike most of our surrounding environment, people are continually moving. We walk around, blink, fidget, wave our hands about, etc.



Figure 5.2.1: Frame 1 from camera Figure 5.2.2: Frame 2 from camera



Figure 5.2.3: Patio-Temporally filtered image

Since in real-time face detection, the system is presented with a series of frames in which to detect a face, by using spatiotemporal filtering (finding the difference between subsequent frames), the area of the frame that has changed can be identified and the individual detected (Wang and Adelson, 1994 and Adelson and Bergen 1986). Furthermore, as seen in Figure exact face locations can be easily identified by using a few simple rules, such as,

1)the head is the small blob above a larger blob -the body

2)head motion must be reasonably slow and contiguous -heads won't jump around erratically (Turk and Pentland 1991a, 1991b).

Real-time face detection has therefore become a relatively simple problem and is possible even in unstructured and uncontrolled environments using these very simple image processing techniques and reasoning rules.

### FACE DETECTION PROCESS

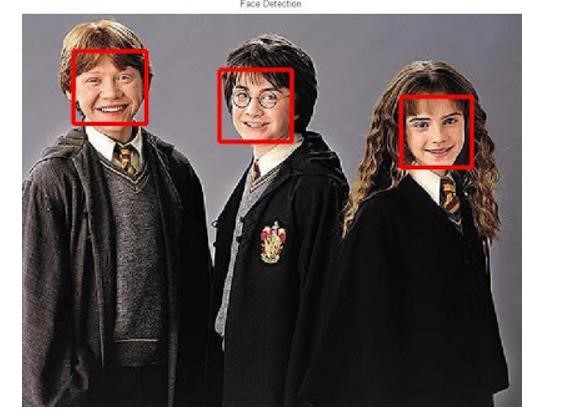
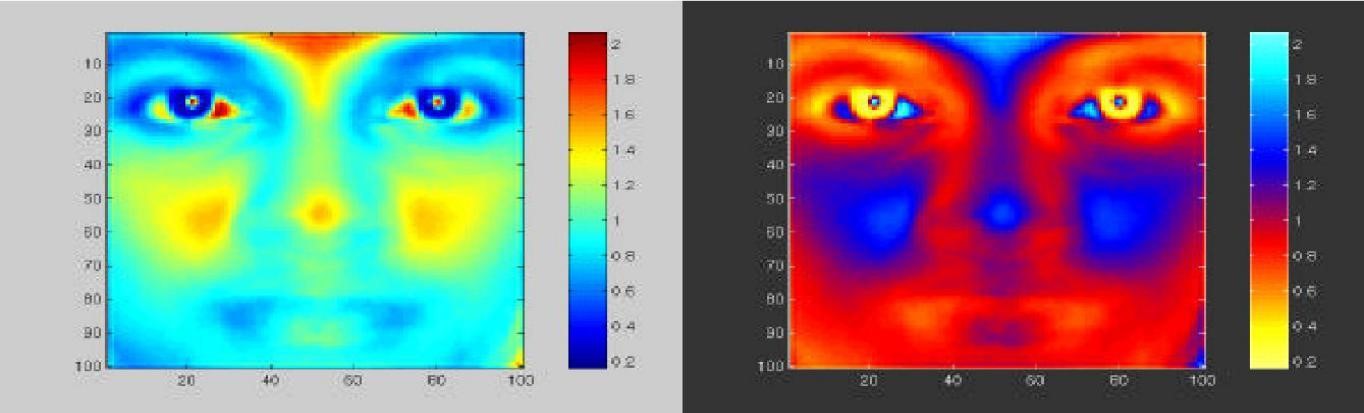


Fig 5.3 Face detection

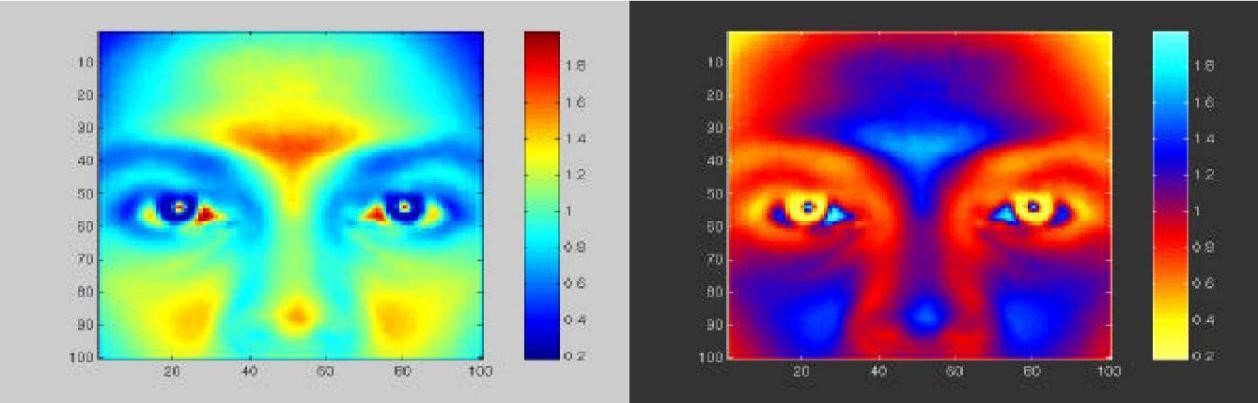
It is process of identifying different parts of human faces like eyes, nose, mouth, etc… this process can be achieved by using MATLAB codeine this project the author will attempt to detect faces in still images by using image invariants. To do this it would be useful to study the greyscale intensity distribution of an average human face. The following 'average human face' was constructed from a sample of 30 frontal view human faces, of which 12 were from females and 18 from males. A suitably scaled colormap has been used to highlight grey-scale intensity differences.



scaled colormapscaled colormap (negative)

Figure 5.3.1 Average human face in grey-scale

The grey-scale differences, which are invariant across all the sample faces are strikingly apparent. The eye-eyebrow area seems to always contain dark intensity (low) gray-levels while nose forehead and cheeks contain bright intensity (high) grey-levels. After a great deal of experimentation, the researcher found that the following areas of the human face were suitable for a face detection system based on image invariants and a deformable template.



scaled colormap scaled colormap (negative)

Figure 5.3.2 Area chosen for face detection (indicated on average human face in gray scale)

The above facial area performs well as a basis for a face template, probably because of the clear divisions of the bright intensity invariant area by the dark intensity invariant regions. Once this pixel area is located by the face detection system, any particular area required can be segmented based on the proportions of the average human faceAfter studying the above images it was subjectively decided by the author to use the following as a basis for dark intensity sensitive and bright intensity sensitive templates. Once these are located in a subject's face, a pixel area

33.3% (of the width of the square window) below this.

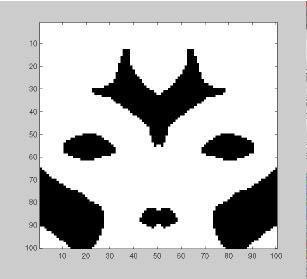


Figure 5.3.3: Basis for a bright intensity invariant sensitive template.

Note the slight differences which were made to the bright intensity invariant sensitive template (compare Figures 3.4 and 3.2) which were needed because of the pre-processing done by the system to overcome irregular lighting (chapter six). Now that a suitable dark and bright intensity invariant templates have been decided on, it is necessary to find a way of using these to make 2 A-units for a perceptron, i.e., a computational model is needed to assign neurons to the distributions displayed.



Fig 5.3.4 Scanned image

Detection

* San window over image
* Classify window as either
  1. Face
  2. Non - Face

### 5.4 FACE DETECTION ALGORITHM

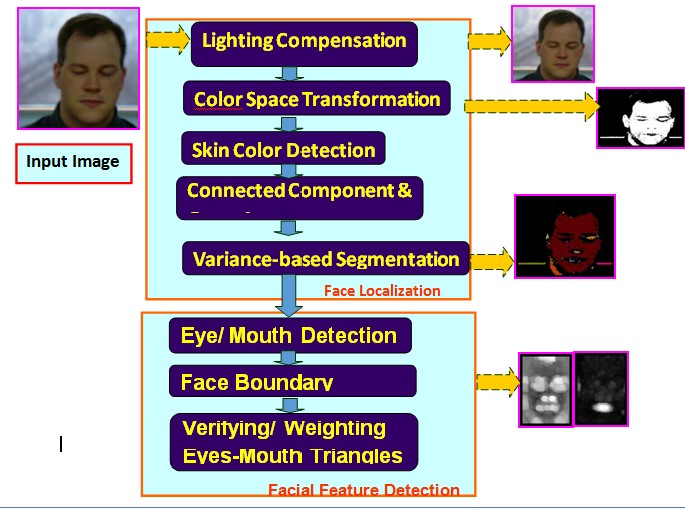


Fig 5.4 Face detection algorithm

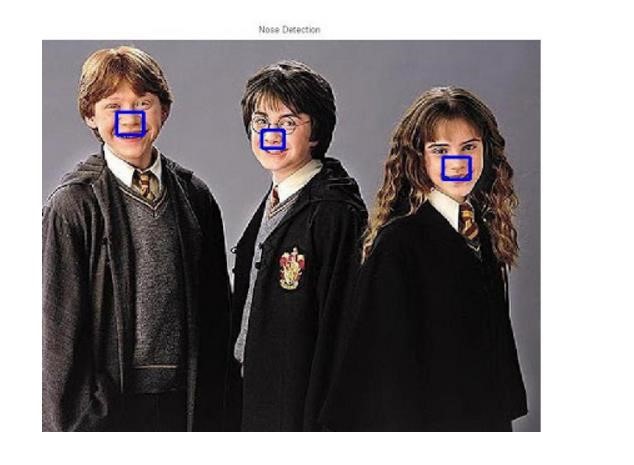


Fig 5.4.1 mouth detection Fig 5.4.2 Noise detection

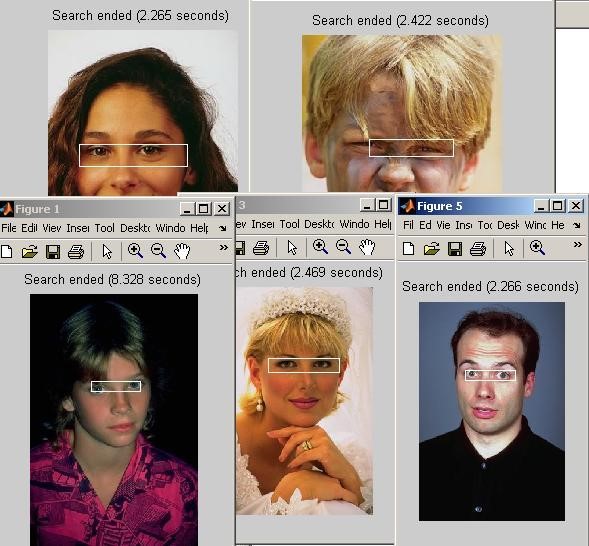


Fig 5.4.3 Eye detection

## FACE RECOGNITION

Over the last few decades many techniques have been proposed for face recognition. Many of the techniques proposed during the early stages of computer vision cannot be considered successful, but almost all of the recent approaches to the face recognition problem have been creditable. According to the research by Brunelle and Poggi (1993) all approaches to human face recognition can be divided into two strategies:

1. Geometrical features and
2. Template matching.

### FACE RECOGNITION USING GEOMETRICAL FEATURES

This technique involves computation of a set of geometrical features such as nose width and length, mouth position and chin shape, etc. from the picture of the face we want to recognize. This set of features is then matched with the features of known individuals. A suitable metric such as Euclidean distance (finding the closest vector) can be used to find the closest match. Most pioneering work in face recognition was done using geometric features (Kaneda, 1973), although Craw et al. (1987) did relatively recent work in this area.

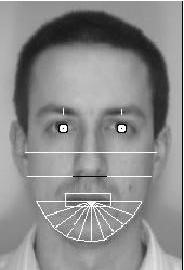


Figure 6.1 Geometrical features (white) which could be used for face recognition

The advantage of using geometrical features as a basis for face recognition is that recognition is possible even at very low resolutions and with noisy images (images with many disorderly pixel intensities). Although the face cannot be viewed in detail its overall geometrical configuration can be extracted for face recognition. The technique's maindisadvantage is that automated extraction of the facial geometrical features is very hard. Automated geometrical feature extraction-based recognition is also very sensitive to the scaling and rotation of a face in the image plane (Brunelle and Poggi, 1993). This is apparent when we examine Kenadee’s (1973) results where he reported a recognition rate of between 45-75 % with a database of only 20 people. However, if these features are extracted manually as in Goldstein et al. (1971), and Kaya and Kobayashi (1972) satisfactory results may be obtained.

#### Face recognition using template matching

This is similar the template matching technique used in face detection, except here we are

not trying to classify an image as a 'face' or 'non-face' but are trying to recognize a face.



Figure .6.11 Face recognition using template matching

Whole face, eyes, nose and mouth regions which could be used in a template matching strategy. The basis of the template matching strategy is to extract whole facial regions (matrix of pixels) and compare these with the stored images of known individuals. Once again Euclidean distance can be used to find the closest match. The simple technique of comparing grey-scale intensity values for face recognition was used by Baron (1981). However, there are far more sophisticated methods of template matching for face recognition. These involve extensive preprocessing and transformation of the extracted grey-level intensity values. For example, Turk and Pentland (1991a) used Principal Component Analysis, sometimes known as the eigenfaces approach, to pre-process the gray-levels and Wescott et al. (1997) used Elastic Graphs encoded using Gabor filters to pre-process the extracted regions.An investigation of geometrical features versus template matching for face recognition by Brunelle and Poggi (1993) came to the conclusion that although a feature-based strategy may offer higher recognition speed and smaller memory requirements, template-based techniques offer superior recognition accuracy.

### PROBLEM SCOP AND SYSTEM SPECIFICATION

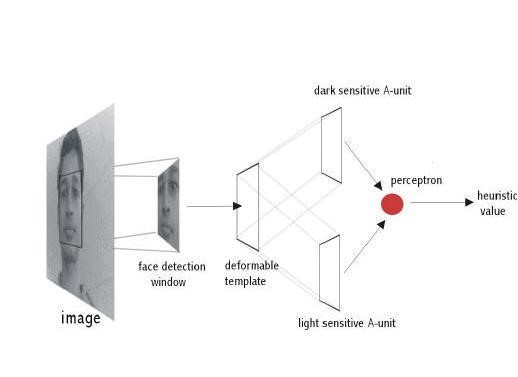
The following problem scope for this project was arrived at after reviewing the literature on face detection and face recognition, and determining possible real-world situations where such systems would be of use. The following system(s) requirements were identified

1. A system to detect frontal view faces in static images
2. A system to recognize a given frontal view face
3. Only expressionless, frontal view faces will be presented to the face detection recognition4 All implemented systems must display a high degree of lighting invariancy.
4. All systems must posses near real-time performance.
5. Both fully automated and manual face detection must be supported
6. Frontal view face recognition will be realized using only a single known image
7. Automated face detection and recognition systems should be combined into a fully automated face detection and recognition system. The face recognition sub-system must display a slight degree of invariancy to scaling and rotation errors in the segmented image extracted by the face detection sub-system.
8. The frontal view face recognition system should be extended to a pose invariant face recognition system.

Unfortunately, although we may specify constricting conditions to our problem domain, it may not be possible to strictly adhere to these conditions when implementing a system in the real-world.

### BRIEF OUT LINE OF THE IMPLEMENTED SYSTEM

Fully automated face detection of frontal view faces is implemented using a deformable template algorithm relying on the image invariants of human faces. This was chosen because a similar neural-network based face detection model would have needed far too much training data to be implemented and would have used a great deal of computing time. The main difficulties in implementing a deformable template-based technique were the creation of the bright and dark intensity sensitive templates and designing an efficient implementation of the detection algorithm.



Figure

6

.

3

Implemented fully

automated frontal view face detection model

A manual face detection system was realized by measuring the facial proportions of the average face, calculated from 30 test subjects. To detect a face, a human operator would identify the locations of the subject's eyes in an image and using the proportions of the average face, the system would segment an area from the image

A template matching based technique was implemented for face recognition. This was because of its increased recognition accuracy when compared to geometrical features-based techniques and the fact that an automated geometrical features-based technique would have required complex feature detection pre-processing.

Of the many possible template matching techniques, Principal Component Analysis was chosen because it has proved to be a highly robust in pattern recognition tasks and because it is relatively simple to implement. The author would also liked to have implemented a technique based on Elastic Graphs but could not find sufficient literature about the model to implement such a system during the limited time available for this project.

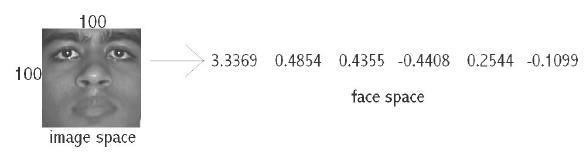


Figure 6.3.1: Principal Component Analysis transform from 'image space' to 'face space'.

Using Principal Component Analysis, the segmented frontal view face image is transformed from what is sometimes called 'image space' to 'face space'. All faces in the face database are transformed into face space. Then face recognition is achieved by transforming any given test image into face space and comparing it with the training set vectors. The closest matching training set vector should belong to the same individual as the test image. Principal Component Analysis is of special interest because the transformation to face space is based on the variation of human faces (in the training set). The values of the 'face space' vector correspond to the amount certain 'variations' are present in the test image

Face recognition and detection system is a pattern recognition approach for personal identification purposes in addition to other biometric approaches such as fingerprint recognition, signature, retina and so forth. Face is the most common biometric used by humans’ applications ranges from static, mug-shot verification in a cluttered background.



Fig 6.3.2 Face Recognition

### FACE RECOGNITION DIFFICULTIES

1. Identify similar faces (inter-class similarity)
2. Accommodate intra-class variability due to
   1. head pose
   2. illumination conditions
   3. expressions
   4. facial accessories
   5. aging effects
3. Cartoon faces

**Inter - class similarity:**

Different persons may have very similar appearance



Fig 6.4.1 Face recognition twins and other and Son

Face recognition and detection system is a pattern recognition approach for personal identification purposes in addition to other biometric approaches such as fingerprint recognition, signature, retina and so forth. The variability in the faces, the images are processed before they are fed into the network. All positive examples that is the face images are obtained by cropping images with frontal faces to include only the front view. All the cropped images are then corrected for lighting through standard algorithms.

#### Inter – class variability

Faces with intra-subject variations in pose, illumination, expression, accessories, color, occlusions, and brightness

### UNDERSTANDING EIGENFACES

Any grey scale face image I(XXY) consisting of a Nixon array of intensity values may also be consider as a vector of N2. For example, a typical 100x100 image used in this thesis will have to be transformed into a 10000-dimension vector!

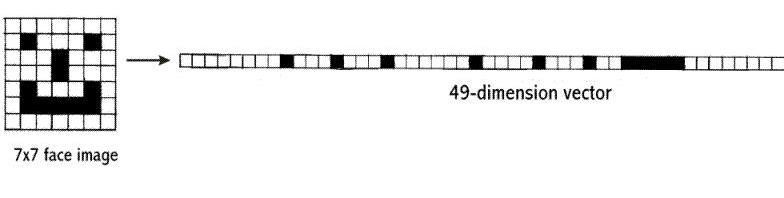
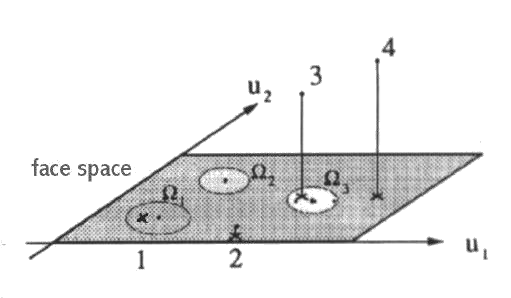
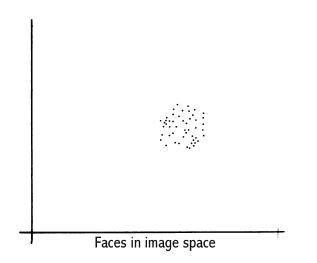


Figure 6.6.0 A 7x7 face image transformed into a 49-dimension vector

This vector can also be regarded as a point in 10000-dimension space. Therefore, all the images of subjects' whose faces are to be recognized can be regarded as points in 10000-dimension space. Face recognition using these images is doomed to failure because all human face images are quite similar to one another so all associated vectors are very close to each other in the 10000-



dimension space.



Fig 6.6.1 Eigenfaces

The transformation of a face from image space (I) to face space (f) involves just a simple matrix multiplication. If the average face image is A and U contains the (previously calculated) eigenfaces,

f = U \* (I - A)

This is done to all the face images in the face database (database with known faces) and to the image (face of the subject) which must be recognized. The possible results when projecting a face into face space are given in the following figure.

.

There are four possibilities:

1. Projected image is a face and is transformed near a face in the face database

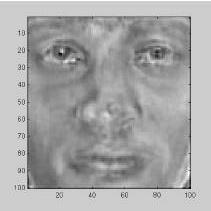
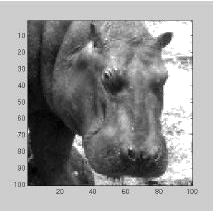
2.Projected image is a face and is not transformed near a face in the face database

3.Projected image is not a face and is transformed near a face in the face database

4. Projected image is not a face and is not transformed near a face in the face database While it is possible to find the closest known face to the transformed image face by calculating the Euclidean distance to the other vectors, how does one know whether the image that is being transformed actually contains a face? Since PCA is a many-to-one transform, several vectors in the image space (images) will map to a point in face space (the problem is that even non-face images may transform near a known face image's faces space vector). Turk and Pentland (1991a), described a simple way of checking whether an image is actually of a face. This is by transforming an image into face space and then transforming it back (reconstructing) into image space. Using the previous notation,

I' = UT \*U \* (I - A

With these calculations it is possible to verify that an image is of a face and recognize that face. O'Toole et al. (1993) did some interesting work on the importance of eigen faces with large and small eigenvalues. They showed that the eigen vectors with larger eigenvalues convey information relative to the basic shape and structure of the faces. This kind of information is most useful in categorizing faces according to sex, race etc. Eigen vectors with smaller eigenvalues tend to capture information that is specific to single or small subsets of learned faces and are useful for distinguishing a particular face from any other face. Turk and Pentland (1991a) showed that about 40 eigen faces were sufficient for a very good description of human faces since the reconstructed image have only about 2% RMS. pixel-by-pixel errors.

 0.8235

0.0661

-0.8786

-0.4727

-0.0646

0.6642

-0.4840

-0.4501 -0.2506

0.1591

0.3359

0.0048

0.0745

………..

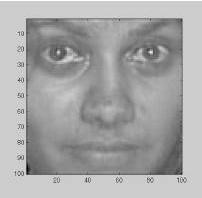
Hippo in image space Hippo in face space Reconstructed hippo in image space

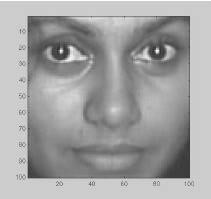
0.7253

-0.0392 0.2896

-0.1725

-0.2642

 -0.0014



-0.0814

-0.0054

-0.0623 -0.0965

-0.0879

0.0745

-0.0261

…………

Face in image space Face in face space Reconstructed face in image space

Figure 6.6.3 Images and their reconstruction. The Euclidean distance between a face image and its reconstruction will be lower than that of a non-face image

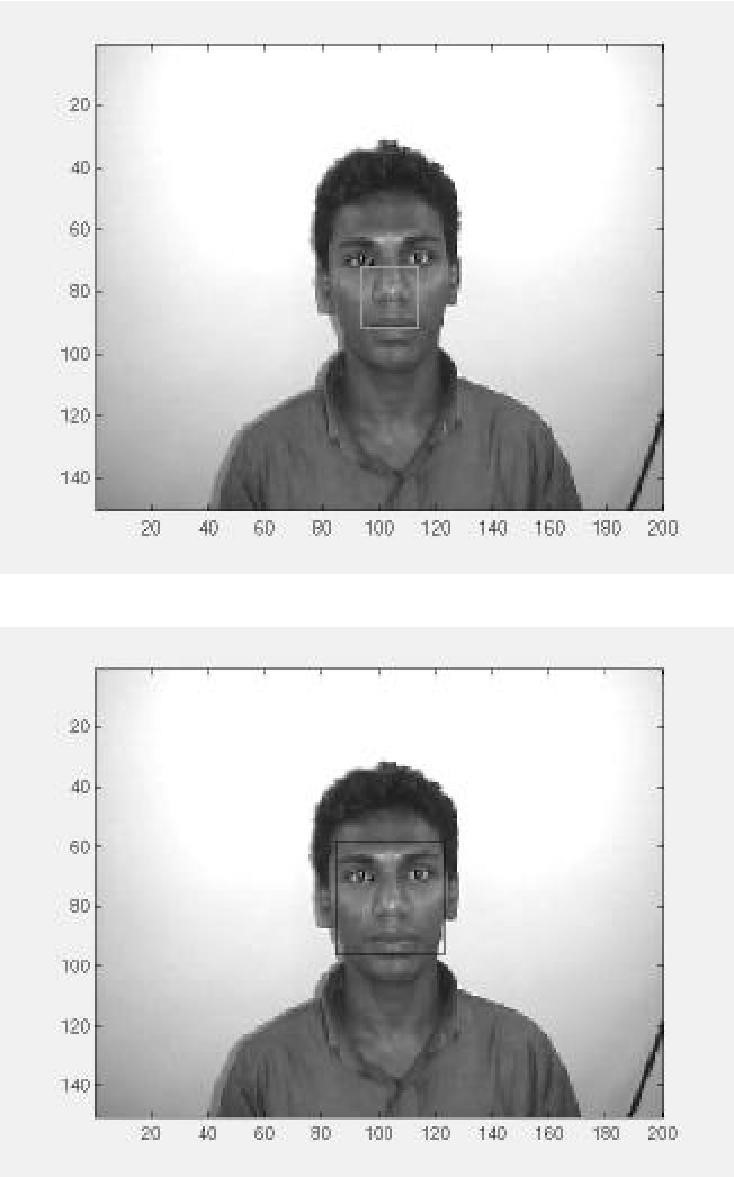
### IMPROVING FACE DETECTION USING RECONSTRUCTIN

Reconstruction cannot be used as a means of face detection in images in near real-time since it would involve resizing the face detection *window* area and large matrix multiplication, both of which are computationally expensive. However, reconstruction can be used to verify whether potential face locations identified by the deformable template algorithm actually contain a face. If the reconstructed image differs greatly from the face detection *window,* then the *window* probably does not contain a face. Instead of just identifying a single potential face location, the face detection algorithm can be modified to output many high 'fakeness' locations which can be verified using reconstruction. This is especially useful because occasionally the best 'fakeness' location found by the deformable template algorithm may not contain the ideal frontal view face pixel area.

Output from Face detection system

|  |  |  |  |
| --- | --- | --- | --- |
| Heuristic | x | y | width |
| 978 | 74 | 31 | 60 |

1872 74 33 60



Best heuristic location (94,65,20)

1994 75 32 58

2125 76 32 56

2418 76 34 56

2389 79 32 50

2388 80 33 48

2622 81 33 46

2732 82 32 44

2936 84 33 40

2822 85 58 38

2804 86 60 36

2903 86 62 36

3311 89 62 30

3373 91 63 26

3260 92 64 24

3305 93 64 22

3393 94 65 20

potential face locations that have been identified by the face detection system (the best face locations it found on its search) are checked whether they contain a face. If the threshold level (maximum difference between reconstruction and original for the original to be a face) is set correctly this will be an efficient way to detect a face. The deformable template algorithm is fast and can reduce the search space of potential face locations to a handful of positions. These are then checked using reconstruction. The number of locations found by the face detection system can be changed by getting it to output, not just the best face locations it has found so far but any location, which has a 'fakeness' value, which for example is, at least 0.9 times the best heuristic value that has been found so far. Then there will be many more potential face locations to be checked using reconstruction. This and similar speed versus accuracy trade-off decisions have to be made keeping in mind the platform on which the system is implemented.

Similarly, instead of using reconstruction to check the face detection system's output, the output's correlation with the average face can be checked. The segmented areas with a high correlation probably contain a face. Once again, a threshold value will have to be established to classify faces from non-faces. Similar to reconstruction, resizing the segmented area and calculating its correlation with the average face is far too expensive to be used alone for face detection but is suitable for verifying the output of the face detection system.

### POSE INVARIANT FACE RECOGNITION

Extending the frontal view face recognition system to a pose-invariant recognition system is quite simple if one of the proposed specifications of the face recognition system is relaxed. Successful pose-invariant recognition will be possible if many images of a known individual are in the face database. Nine images from each known individual can be taken as shown below. Then if an image of the same individual is submitted within a 30o angle from the frontal view he or she can be identified.

Nine images in face database from a single known individual



Unknown image from same individual to be identified



Fig: 6.8 Pose invariant face recognition.

Pose invariant face recognition highlights the generalization ability of PCA. For example, when an individual's frontal view and 30o left view known, even the individual's 15o left view can be recognized.

## CONCLUSION

The computational models, which were implemented in this project, were chosen after extensive research, and the successful testing results confirm that the choices made by the researcher were reliable. The system with manual face detection and automatic face recognition did not have a recognition accuracy over 90%, due to the limited number of eigenfaces that were used for the PCA transform. This system was tested under very robust conditions in this experimental study and it is envisaged that real-world performance will be far more accurate. The fully automated frontal view face detection system displayed virtually perfect accuracy and in the researcher's opinion further work need not be conducted in this area.

The fully automated face detection and recognition system was not robust enough to achieve a high recognition accuracy. The only reason for this was the face recognition subsystem did not display even a slight degree of invariance to scale, rotation or shift errors of the segmented face image. This was one of the system requirements identified in section 2.3. However, if some sort of further processing, such as an eye detection technique, was implemented to further normalize the segmented face image, performance will increase to levels comparable to the manual face detection and recognition system. Implementing an eye detection technique would be a minor extension to the implemented system and would not require a great deal of additional research. All other implemented systems displayed commendable results and reflect well on the deformable template and Principal Component Analysis strategies. The most suitable real-world applications for face detection and recognition systems are for mugshot matching and surveillance. There are better techniques such as iris or retina recognition and face recognition using the thermal spectrum for user access and user verification applications since these need a very high degree of accuracy. The real-time automated pose invariant face detection and recognition system proposed in chapter seven would be ideal for crowd surveillance applications. If such a system were widely implemented its potential for locating and tracking suspects for law enforcement agencies is immense.

The implemented fully automated face detection and recognition system (with an eye detection system) could be used for simple surveillance applications such as ATM user security, while the implemented manual face detection and automated recognition system is ideal of mugshot matching. Since controlled conditions are present when mugshots are gathered, the frontal view face recognition scheme should display a recognition accuracy far better than the results, which were obtained in this study, which was conducted under adverse conditions.

Furthermore, many of the test subjects did not present an expressionless, frontal view to the system. They would probably be more compliant when a 6'5'' policeman is taking their mugshot!In mugshot matching applications, perfect recognition accuracy or an exact match is not a requirement. If a face recognition system can reduce the number of images that a human operator has to search through for a match from 10000 to even a 100, it would be of incredible practical use in law enforcement.

The automated vision systems implemented in this thesis did not even approach the performance, nor were they as robust as a human's innate face recognition system. However, they give an insight into what the future may hold in computer vision.

THANK YOU