Improving Face Recognition in Video Key-Frames for e-Learning Systems

S.C. Premaratne

M.Phil Registration No: MPhil/FT/2004/001

Supervisor: Dr. D.D. Karunaratna

This dissertation is submitted to fulfill the requirement
of the Degree of M.Phil
of the
University of Colombo School of Computing

Declaration

I certify that this dissertation does not incorporate, without acknowledgement, any material previously submitted for a Degree or Diploma in any University and to the best of my knowledge and belief, it does not contain any material previously published or written by another person or myself except where due reference is made in the text. I also hereby give consent for my dissertation, if accepted, to be made available for photocopying and for inter-library loans, and for the title and summary to be made available to outside organizations.

....

Signature of Candidate Date:…../…..../…...

Name of Candidate:

To the best of my knowledge the above particulars are correct.

Approved by ____

Supervisor

____

____

____

Date:…../…..../…...

Abstract

E-learning has become an integral part in higher education in the last decade. The emerging multimedia information technologies allow researchers to identify new ways to store, retrieve, share, and manipulate complex information which are expected to be used for building exciting new e-learning applications. The key challenges in this field are related to data organization and integration, indexing and retrieval mechanisms, intelligent searching techniques, information browsing, content-based query processing, handling of heterogeneity etc.

This thesis reveals a profile based feature identification system for multimedia database systems which is designed to support the use of video clips for e-learning. The system creates profiles of presenters appearing in the video clips based on their facial features and uses these profiles to identify similar video segments based on the presenter profiles. The face recognition algorithm used by the system is based on the Principal Components Analysis (PCA) approach. The thesis addresses one of the main problems identified in profile construction over video key-frames which is the overlapping of key-frames in the eigenspace. It explains various tests carried out to explore the courses for this problem and then proposes a novel approach to overcome the problem by introducing a profile normalization algorithm. In particular, this method reveals the profile overlapping problem can be controlled by using certain parameters obtained by analyzing a collection of key-frames.

List of Publications.

This thesis is based on the work reported in the following publications.

i. S. C. Premaratne, D. D. Karunaratna, G. N. Wikramanayake K. P. Hewagamage and G. K. A. Dias. An Architecture of a Media Based System to Support E-Learning. The Bulletin of the British Computer Society Sri Lanka Section, October 2004. pp. 32-33.

ii. S. C. Premaratne, D. D. Karunaratna, G. N. Wikramanayake K. P. Hewagamage and G. K. A. Dias. Profile Based Video Segmentation System to Support E-Learning. Proceedings of the 6th International Information Technology Conference, 2004. Colombo, Sri Lanka. pp. 74-81.

iii. S. C. Premaratne, D. D. Karunaratna, G. N. Wikramanayake K. P. Hewagamage and G. K. A. Dias. Implementation of a Profile Based Video Segmentation System. Proceedings of the International Conference on Information Management in a Knowledge Society, 2005, Grand Hyatt Mumbai, Maharashtra, India. pp 89 – 100.

iv. S. C. Premaratne, D. D. Karunaratna, G. N. Wikramanayake K. P. Hewagamage and G. K. A. Dias. Efficient Profile Construction Algorithm for Video Indexing in E-Learning. Proceedings of the 11th International Conferenceon Virtual Systems and multimedia, 2005, Flanders Expo, Ghent, Belgium. pp 65-74

v. S. C. Premaratne, D. D. Karunaratna, G. N. Wikramanayake K. P. Hewagamage and G. K. A. Dias. Improvised Profile Construction for Multimedia Databases in E-Learning. Proceedings of the MMU International Symposium on Information and Communication Technology 2005. Kuala Lampur, Malaysia. TS12, pp 9-13

vi. S. C. Premaratne, D. D. Karunaratna, and K. P. Hewagamage. Profile Based Video Browsing for E-Learning. Proceedings of the 10th IASTED International Conference on Software Engineering and Applications 2006. Dallas, Texas USA. pp 489 – 494.

vii. S. C. Premaratne, D. D. Karunaratna, and K. P. Hewagamage. Collaborating Educational Videos with Presenter Profiles for Effective Content-based Video Retrieval. Proceedings of the Digital Learning Asia 2006. http://www.digitallearning.in/dlasia/2007/agenda_day3_3.asp.

viii. S. C. Premaratne and D. D. Karunaratna. An Effective Profile Based Video Browsing System for E-Learning. Electronic journal of e-Learning. http://www.ejel.org/Volume-5/v5-i2/v5-i2-art-6.htm

Acknowledgements

First and foremost, I wish to express my heartfelt gratitude to my supervisor Dr. Damitha Karunaratna for his continuous support and guidance provided throughout the research. This would not have been possible without the support of my supervisor.

I acknowledge with sincere gratitude to the other research group members, Dr G. N. Wikramanayake, Dr K. P. Hewagamage and Mr. G. K. A. Dias for their valuable suggestions and comments. And also I wish to thank Dr. Ruwan Weerasinghe, Director of the University of Colombo School of Computing for the ever present helping hand and encouragement.

This research work is initially supported by the Japan International Cooperation Agency (JICA) with technical equipment and Asian Development Bank (ADB) with registration fees and first year stipends. The second year of the research was funded by National Science Foundation (NSF) with monthly stipends. The national e-learning center provided funds for overseas traveling to present research papers. These assistance are gratefully acknowledged.

I express my thanks and appreciation to my family for their understanding, motivation and patience. Lastly, but in no sense the least, I am thankful to all colleagues and friends who made my stay at the university a memorable and valuable experience.

Table of Contents

List of Tables

List of Figures

List of Acronyms

BIT	Bachelor of Information Technology
BIC	Bayesian Information Criterion
CNN	Convolutional Neural Network
DARPA	Defense Advanced Research Products Agency
DCT	Discrete Cosine Transformation
DDL	Data definition language
DS	Description Schemes
EGM	Elastic Graph Matching
FAR	False Acceptance Rate
FERET	Facial Recognition Technology
FRR	False Rejection Rate
GMM	Gaussian Mixture Model
MPEG	Moving Picture Expert Group
PCA	Principal Components Analysis
XML	Extensible Markup Language

Chapter: 1

1 Introduction

1.1 Integration of e-learning and multimedia databases

Today we live in a knowledge society where knowledge has become a necessary factor for the development. In today's rapidly changing electronic world, the key to maintain the appropriate momentum in organizations and academic environments is knowledge. Education has always been considered as a life-long activity. Therefore, continuous, convenient and economical access to training material assumes the highest priority for the ambitious individual or organization. This requirement is met by electronic learning (e-learning). E-learning is one of the fastest growing areas of the advanced technology sector today. It is interactive and involves the use of multimedia. The term E-learning covers computer-based learning, web-based learning and virtual classrooms. E-learning can be delivered via numerous electronic mediums such as the Internet, intranets, extranets, satellite broadcast, audio/videotape, interactive television, and CD-ROM. When students are using e-learning they play an active role rather than the passive role of recipient of information transmitted by a teacher, textbook, or broadcast. At its best, e-learning is individual, customized learning that allows learners to choose and review material at their own pace at anytime anywhere. At its worst, it may disempower and demotivate learners by leaving them lost and unsupported in an immensely confusing electronic realm. Leveraging the most advanced technology, multimedia have raised the learners' interest and provide methods to learn effectively.

Multimedia includes more than one form of media such as text, graphics, animation, audio, video and video conferencing. The term Interactivity (interactive learning) means, a computer is used actively in the delivery of learning materials in the context of education and training. A person can navigate through a computer-based interactive learning environment, in order to select relevant information, respond to questions using input devices such as a keyboard, mouse, touch screen, or voice command system, complete tasks, communicate with others, and receive feedback on assessment. Integration of heterogeneous data as content for e-learning applications is crucial, since the amount and versatility of processable information is the key to a successful system.

Prototypes of the Knowledge Management (KM) systems which simulate people have recently been developed in education and research institutions for e-learning applications, and such knowledge management systems due to the fact of their low cost motivate people to continue learning. In recent times those applications that allow semantic enrichment of data and a loose categorization of the presented content have become popular, since they put forward an unchanged presentation of the e-content.

1.2 Major issues on present e-learning systems

Several approaches have been proposed to increase the acceptance and usage of existing e-learning platforms in education, but most of them are restricted in flexibility with regard to the content and adaptation to the user's skills [Hauptmann 1999, Lorente and Torres 1998, Spaniol et al. 2002]. In our research, we have recognized the need to provide an e-learning system to satisfy requirements of users with different learning objectives and learning patterns. Also it was discovered that low bandwidth is an impediment for the success of an e-learning system. Therefore techniques must be developed for efficient utilization on the available bandwidth. One solution to this problem is to provide facilities for the user to browse and select what he actually required before delivering the material. This can be done by categorizing and clustering various types of educational materials by using ontologies and indices.

1.3 Problem Statement and the Scope of the Project

In our effort to deliver educational video materials for the Bachelor of Information Technology external degree program conducted by the University of Colombo School of Computing through the Internet we have faced with the issues stated in the previous section. In our attempt to integrate video clips into e-learning we have realized that building an index on top of the video library is a requirement to provide efficient access to the video library. This will provide an easy mechanism for a student to navigate through the available video clips without downloading the entire clips and thus provides a solution to the limited bandwidth problem as well.

The focus of this thesis is on video based educational materials where presenters deliver educational content. To provide content based retrieval of digital video information, we employ a set of tools developed by us to segment video clips semantically into shots by using low level features. Then we identify those segments where presenters appear and extract the relevant information in key frames. These information are then encoded and compared with a database of similarly encoded key frames. The feature information in video frames of a face is represented as an eigenvector which is considered as a profile of a particular person [Turk and Pentland 1991]. In this system, a feature selection and a feature extraction sub-system have been used to construct presenter profiles. The feature extraction process transforms the video key-frame data into a multidimensional feature space as feature vectors. These profiles are then used to construct an index over the video clips to support efficient retrieval of video shots.

One difficulty we came up with is the profile overlapping when the faces of the presenters are projected to the eigenspace. This problem has degraded the indexing process and also reduced the accuracy of the profile identification process with the increase in the number of presenters. Thus in this research, the main emphasis is to investigate the causes for the profile overlapping and to develop a technique to eliminate it.

1.4 Methodology

The structure of profiles are prepared using the Principle Component analysis (PCA) [Pentland el al. 1994, Turk and Pentland 1991, Zhang et al. 1997]. By using this algorithm, the presenter’s facial features are transformed into the feature space. We have observed that variation in lighting conditions as one of the main causes for the overlapping of faces in the eigenspace. Thus, we have tried with different parameters to determine how these parameters affect on the lighting variations in the video key-frames. This variation in lighting conditions cannot be eliminated as the video clips are filmed on different manner and at different times by different technical staff. Also the variation in lighting conditions has adverse effect on profile classification. Thus our efforts were to identify parameters to control this lighting effect and to construct an algorithm to overcome this problem.

After evaluating on different data sets with different parameter settings we have identified that certain parameters could be used effectively to normalize the profile with respect to lighting and hence to resolve profile overlapping. As a result, a novel profile normalization algorithm is introduced to avoid the profile overlapping problem when the faces are projected to the eigenspace. The effectiveness of the normalizing algorithm was tested by comparing Total Error Rate (TER) with and without the normalization process.

1.5 Thesis outline

The remainder of the thesis is organized as follows. Chapter two reviews a number of techniques related to our work. The system architecture is shown in Chapter three. Chapter four explains the technique for segmenting face regions and describes the use of PCA for our work. The experiment results are shown in Chapter five and the evaluation of the method is shown in Chapter six. Finally, Chapter seven gives our conclusions and address the future work possible based on this project.

Chapter 2

2 Related Work

With the development of Internet and multimedia technologies, new systems to support e-learning, such as e-learning systems are becoming more popular. These e-learning systems improve effectiveness of teaching in and out of classrooms [Abowd et al. 1998, Dorai et al. 2001, Deshpande and Hwang 2001]. The e-learning infrastructure upgrade requirements tend to increase as the e-learning content becomes more complex and media-rich. Also with the increase popularity of e-learning, the e-learner traffic increases. However very few have tried to explore possibilities to minimize the usage of internet bandwidth and also to locate what the learner wants with easy, when the e-learning systems are extended to capture video clip libraries [Spaniol et al. 2002].

In the field of digital image processing, the focus of research has been not on just detection but also identification of faces, people or some specific objects in video images or video footages. Our research is focused on how these techniques, especially the face recognition techniques can be adopted in the area of e-learning to provide customized services to the e-learning. Face recognition can be divided into two areas: face identification and face verification (also known as authentication). A face verification system verifies the claimed identity based on images (or a video sequence) of the claimant's face; this is in contrast to an identification system, which attempts to find the identity of a given person out of a pool of several people.

Generally, a full face recognition system can be thought of as being comprised of two stages:

1. Face segmentation
2. Face identification.

Face identification can be further subdivided into:

- Feature extraction

- Classification

2.1 Face Localization and Segmentation

The first step of any face processing system is detecting the locations in images where faces are present. The main challenges associated with face detection can be attributed to the following factors:

- Pose: The images of a face vary due to the relative camera-face pose (frontal, 45 degree, profile, upside down), and some facial features such as an eye or the nose may become partially or wholly occluded.
- Presence or absence of structural components: Facial features such as beards, mustaches, and spectacles may or may not be present and there is a great deal of variability among these components including shape, color, and size.
- Facial expression: The appearance of a face is directly affected by a person's facial expression.
- Occlusion: A Face may be partially occluded by other objects. In an image with a group of people, some faces may partially occlude other faces.
- Image orientation: Face images directly vary for different rotations around the camera's optical axis.
- Imaging conditions: When the image is formed, factors such as lighting (spectra, source distribution and intensity) and camera characteristics (sensor response, lenses) affect the appearance of a face.

Due to the variability of the above factor, face detection from a single image is a challenging task because of variability in scale, location, orientation (up-right, rotated), and pose (frontal, profile) [Yang et al. 2002]. Facial expression, occlusion and lighting conditions also change the overall appearance of faces.

2.2 Detecting a Face in a Single Image or Video frame

In general we can classify single face detection methods into four categories, however these methods clearly overlap category boundaries.

- Knowledge-based methods: These rule-based methods encode human knowledge of what constitutes a typical face. Usually, the rules capture the relationships between facial features. These systems are based on the evaluation of coarse forms (eyes, mouth and nose) to detect faces filling up the major part of the image and having good resolution. These algorithms are sometimes based on simple averages of pixel along the lines or columns [Brunelli and Poggio 1993].

- Feature invariant approaches: 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 [Vezhnevets 1998]. The human skin color as well as the eyes are also being used as additional parameters. Movement is sometimes used to locate the presence of a person in the image [Park et al. 2003]. These algorithms make it possible to detect faces of medium size (50 pixels width) in the image but are not very robust in case of the detection of small faces (20 pixels width) when the background is complex [Sandeep and Rajagopalan 2002, Feris et al. 2000].

- Template matching methods: Several standard patterns of a face are stored to describe the face as a whole or the facial features separately [Zhu and Cutu 2002]. The correlations between an input image and the stored patterns are computed for detection. These methods have been used for both face localization and detection. Given an input image, the correlation values with the standard patterns are computed for the face contour, eyes, nose, and mouth independently. The existence of a face is determined based on the correlation values. This approach has the advantage of being simple to implement. However, it has proven to be inadequate for face detection since it cannot effectively deal with variation in scale, pose, and shape.

- Appearance-based methods: In contrast to template matching, the models (or templates) are learned from a set of training images which should capture the representative variability of facial appearance. These learned models are then used for detection. Some use statistical techniques "to learn" what is a face on a basis of examples. The techniques most usually used are the Principal Components Analysis [Turk and Pentland 1991, Pissarenko 2002], the Support Vector Machines [Guo et al. 2000] and the Neural Networks [Rowley et al. 1998, Feraud 1998]. The effectiveness of detecting several small faces in a complex background is sometimes astonishing.

The algorithms of the first category are simple. It is generally possible to carry out them in real time on small systems [Fr¨oba et al. 2001]. Most of the time, the algorithms of the second and fourth categories are implemented on expensive workstations dedicated to image processing and employee real time processing in tracking mode in which a small part of the image is analyzed [Kawato and Ohya 2000].

Our face detection procedure classifies key-frames based on the value of simple features. There are many motivations for using features rather than the pixels directly [Kawato and Ohya 2000, Papageorgiou et al. 1998, Viola and Jones 2001 a]. The main reason for using this method is that features of the image can act to encode ad-hoc domain knowledge that is difficult to learn when using a finite quantity of training data.

2.3 Face Recognition Approaches

After the first two stages of the full face recognition system we shall concentrate on the last stage of face recognition. Why computer-based face recognition is challenging? To begin with, a recognition system has to be invariant both to external changes, like environmental light, and the person's position and distance from the camera, and internal deformations, like facial expression and aging. Because most commercial applications use large databases of faces, recognition systems have to be computationally efficient. Given all these requirements, mathematical modeling is not so simple. There are many approaches to face recognition ranging from the Principal Component Analysis (PCA) approach (also known as eigenfaces) [Turk and Pentland 1991], Elastic Graph Matching (EGM) [Lades et al. 1993], Artificial Neural Networks [Lawrence et al. 1997 , Palanivel et al. 2003], to Hidden Markov Models (HMM) [Bicego 2003]. All these systems differ in terms of the feature extraction procedure and/or the classification technique used. The face recognition systems relevant to our work are described in the sections below.

2.3.1 Geometric Features and Templates

Brunelli and Poggio compared the performance of a system utilizing automatically extracted geometric features combined with a classifier based on the squared Mahalanobis distance (similar to a single-Gaussian GMM) against a system using a template matching strategy [Brunelli and Poggio 1993, Sun et al. 2000]. In the former system, the geometrical features included:

- Eyebrow thickness and vertical position at the eye center position.
- Coarse description of the left eyebrow’s arches.
- Vertical position and width of the nose.
- Vertical position of the mouth as well as the width and height.
- Set of radii describing the chin shape.
- Face width at nose position.
- Face width halfway between nose tip and eyes.

In the system, four sub-images (automatically extracted from the frontal face image), representing the eye, nose, mouth and face area (from eyebrows downward), were used by a classifier based on normalized cross correlation with a set of template images. The size of the face image was first normalized. Brunelli and Poggio found that the template matching approach obtained superior identification performance and was significantly simpler than the geometric feature based approach [Brunelli and Poggio 1993]. Moreover, they have also found that the face areas can be sorted by discrimination ability as follows: eyes, nose and mouth; where eyes has the highest ability differenciate a face and they further noted that this ordering is consistent with human ability of identifying familiar people from a single facial characteristic.

2.3.2 Principal Component Analysis (PCA).

Turk and Pentland presented a face recognition scheme in which face images are projected onto the principal components of the original set of training images [Turk and Pentland 1991]. The resulting eigenfaces are classified by comparison with known individuals. These eigenvectors can be thought of as a set of features that together characterize the variation between face images. The idea behind eigenfaces is to find a lower dimensional space which is capable of describing faces.

Any Gray scale face frame of *N*x*N* array of intensity values may also be considered as a vector *of* N^2. For an example, a simple 7x7 image can be transformed into a 49 dimension vector as shown in Figure 2.1.

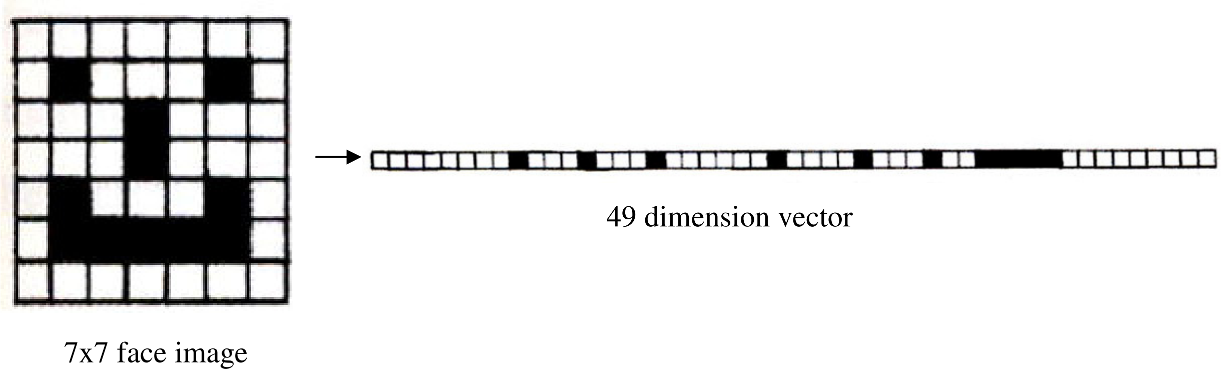


Figure 2.1: A 7x7 dimension face image transformed into a 49 dimension vector

This vector can be considered as a point in 49 dimensional space which is called the eigenspace. Therefore, all the faces once transformed into such vectors can be regarded as a set of points in 49 dimensional eigenspace (Figure 2.2).

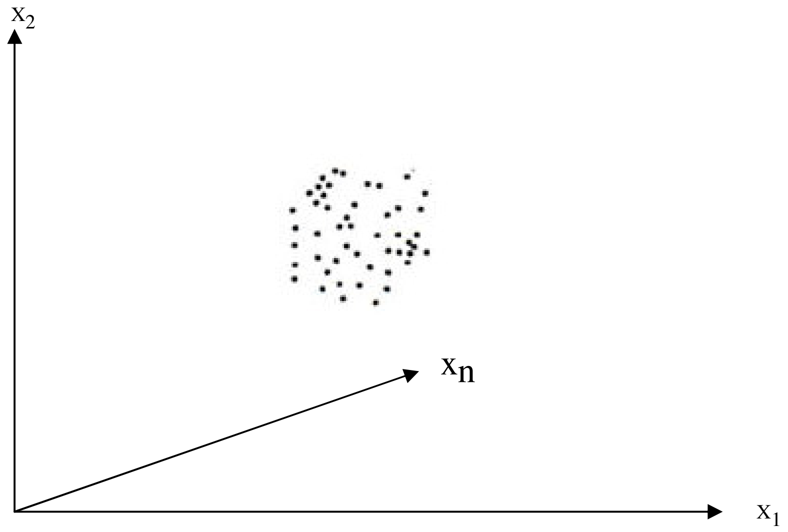


Figure 2.2: Faces in Eigenspace

In PCA, the recognition system is based on the representation of the faces using the so called eigenfaces. In the eigenface representation, every training face is considered a vector of pixel gray values (i.e. the training images are rearranged using row ordering).

An eigenvector of a matrix (**A**) is a vector (***u***) given in the equation 2.1, if multiplied with the matrix, the result is always an integer multiple of that vector. This integer value (***λ***) is the said to be the eigenvalue corresponding ot the eigenvector (***u***).

$$A \times u = \lambda \times u \quad -(2.1)$$

Eigenvectors possess following properties:

- There are ***n*** eigenvectors (and corresponding eigenvalues) in an ***n × n*** matrix.
- All eigenvectors are perpendicular.

However eigenvectors can be determined only for square matrices. If there is ***M*** total eigenvectors in the eigenspace, the average matrix Ψ is calculated and then subtracted from the original faces (Γ_i) as given in the following equation (2.2) and (2.3), and the result is stored in the variable Φ_i:

$$\Psi = \frac{1}{M}\sum_{n=1}^{M}\Gamma_n \qquad \text{- (2.2)}$$

$$\Phi_i = \Gamma_i - \Psi \qquad \text{- (2.3)}$$

Then the eigenvectors (eigenfaces) and the corresponding eigenvalues are calculated using the equation 2.1. The eigenvectors (eigenfaces) constructed in this way are normalized so that they are unit vectors of length 1. From all the ***M*** eigenvectors (eigenfaces) created for a person, only an abstract of ***M*** eigenfaces of highest eigenvalues are chosen. The higher the eigenvalue, the more characteristic features of a face does the particular eigenvector describe. Eigenfaces with low eigenvalues can be omitted, as they explain only a small part of characteristic features of the faces.

$$C = \frac{1}{M}\sum_{n=1}^{M}\Phi_n\Phi_n^T = AA^T \qquad \text{- (2.4)}$$

$$L = A^TA \quad L_{n,m} = \Phi_m^T\Phi_n \qquad \text{- (2.5)}$$

$$u_l = \sum_{k=1}^{M} v_{lk}\Phi_k \quad l = 1,\ldots,M \qquad \text{- (2.6)}$$

Where L is an $M \times M$ matrix and v are M eigenvectors of L and u are eigenfaces.
The covariance matrix C is calculated using the formula $C = AA^T$ (equation 2.4). The advantage of this method is that one has to evaluate only M numbers and not N^2. Usually, $M << N^2$ as only a few principal components (eigenfaces) will be relevant. The amount of calculations to be performed is reduced from the number of pixels ($N^2 \times N^2$) to the number of key-frames in the training set (M) (equation 2.6). We will use only a subset of M eigenfaces, the M' eigenfaces with the largest eigenvalues. Eigenvector selection process is explained in detail on section 2.6. After M' eigenfaces are determined, the ”training” phase of the algorithm can be accomplished.

There is a problem with the algorithm described in equation 2.4. The covariance matrix C has a dimensionality of $N^2 \times N^2$, so one would have N^2 eigenfaces and eigenvalues. For a 128 × 128 key-frame means that one must compute a 16,384 × 16,384 matrix and calculate 16,384 eigenfaces. Computationally, this is not very efficient as most of those eigenfaces are not useful for our task.

The process of classification of a new (unknown) face Γ_{new} to one of the faces (known faces) proceeds in two steps. First, the new key-frame is transformed into its eigenface components. The resulting weights $\boldsymbol{w}$ form the weight vector are computed by using the equation given in (2.7) and (2.8) below.

$$\omega_k = u_k^T(\Gamma_{\text{new}} - \Psi) \quad k = 1 \ldots M' \qquad \text{- (2.7)}$$

$$\Omega_{\text{new}}^T = [\ \omega_1 \quad \omega_2 \quad \ldots \quad \omega_{M'}\ ] \qquad \text{- (2.8)}$$

The Euclidean distance between two weight vectors $d(\Omega_i, \Omega_j)$ provides a measure of similarity between the corresponding key-frames $\boldsymbol{i}$ and $\boldsymbol{j}$. If the Euclidean distance between Γ_{new} and other faces exceeds on average some threshold value $\boldsymbol{\theta}$, we can assume that Γ_{new} is not a known face. $d(\Omega_i, \Omega_j)$ also allows one to construct ”clusters” of faces such that similar faces are assigned to one cluster. Let an arbitrary instance ***x*** be described by the feature vector

$$x = [a_1(x), a_2(x), \ldots, a_n(x)] \qquad \text{- (2.9)}$$

Where $a_r(x)$ denotes the value of the ***r*** th attribute of instance ***x***. Then the distance between two instances x_i and x_j is defined to be $d(x_i, x_j)$:

$$d(x_i, x_j) = \sqrt{\sum_{r=1}^{n} (a_r(x_i) - a_r(x_j))^2} \qquad \text{- (2.10)}$$

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

2.3.3 Elastic Graph Matching (EGM)

Another approach to face recognition is the well known method of Graph Matching. Martin Lades present a Dynamic Link Architecture for distortion invariant object recognition which employs elastic graph matching to find the closest stored graph [Lades et al. 1993]. Objects are represented with sparse graphs where vertices are labeled with a multi-resolution description in terms of a local power spectrum, and edges are labeled with geometrical distances.

They present good results with a database of 87 people and test images composed of different expressions and faces turned 15 degrees. The matching process is computationally expensive, taking roughly 25 seconds to compare an image with 87 stored objects when using a parallel machine with 23 transputers. Wiskott [Wiskott et al.1995, Wiskott et al. 1997] use an updated version of the technique and compare 300 faces against 300 different faces of the same people taken from the Facial Recognition Technology (FERET) database. One drawback of this system is that they haven't tested the robustness to variations such as illumination changes or orientation variations.

2.3.4 Neural Network Approaches

Much of the present literature on face recognition with neural networks presents results with only a small number of classes (often below 20). Steve Lawrence presented a hybrid neural network solution [Lawrence et al. 1997] which can be superior to other methods. In Neural Networks, the knowledge is not encoded by a programmer into a program, but is embedded in the weights of the neurons. Whilst Expert Systems and Knowledge-Based Systems try to emulate human conceptual mechanisms at a high level, Neural Networks try to simulate these mechanisms at a lower level. They attempt to reproduce not only the input/output behavior of the human brain, but also its internal structure. Knowledge is then stored in a non-symbolic fine-grained way. The weights can be set through a learning process, the goal of which is to obtain values which give the network the desired input/output behaviour. The system combines local image sampling, a self-organizing map neural network, and a Convolutional Neural Network (CNN) [Szlávik and Szirányi 2003].

The Self-Organizing Map (SOM), introduced by Teuvo Kohonen is an unsupervised learning process which learns the distribution of a set of patterns without any class information [Kohonen 1988]. A pattern is projected from an input space to a position in the map – information is coded as the location of an activated node. The SOM is unlike most classification or clustering techniques, provides a topological ordering of the classes. Similarity in input patterns is preserved in the output of the process. CNN incorporate constraints and achieve some degree of shift and deformation invariance using three ideas: local receptive fields, shared weights, and spatial subsampling. The use of shared weights also reduces the number of parameters in the system aiding generalization. Steve Lawrence performed various experiments. In most cases experiments were performed with 5 training images and 5 test images per person for a total of 200 training images and 200 test images. One drawback of Neural Networks is its slow rate of learning, making it less than ideal for real-time use.

2.3.5 Independent Component Analysis (ICA)

ICA can be seen as an extension to principal component analysis and factor analysis. It's a statistical and computational technique for revealing hidden factors that underlie sets of random variables, measurements, or signals. The goal of ICA is to recover independent sources given only sensor observations that are unknown linear mixtures of the unobserved independent components [Bartlett and Sejnowski 1997]. In contrast to correlation-based transformations such as PCA, ICA reduces higher-order statistical dependencies, attempting to make the signals as independent as possible. ICA for face recognition has been applied only relatively recently. In that work, a subset of ICA components were selected by a heuristic employing PCA to perform dimensionality reduction and conducting ICA on the principal component basis. The ICA method computes independent components by maximizing non-Gaussianity of whitened data distribution using a kurtosis maximization process. The kurtosis measures the non-Gaussianity and the sparseness of the face representations [Bartlett et al. 2002].

Previous results of applying ICA to human face recognition on the FERET database and the Olivetti and Yale databases showed that ICA outperforms PCA [Yuen and Lai 2000, Liu and Wechsler.1999]. Another report claimed that there is no performance difference between ICA and PCA [Moghaddam 1999]. Baek et al. found, that PCA significantly outperforms ICA when the best performing distance metric is used for each method [Baek et al. 2000].

2.3.6 Other Approaches

Baback Moghaddam proposes a new technique for the purposes of face recognition using a probabilistic measure of similarity, based primarily on a Bayesian analysis of image differences [Moghaddam et al. 2000]. The work was based on probabilistic similarity measure based on the Bayesian belief on image intensity differences. The system was tested with Defense Advanced Research Products Agency (DARPA) and Facial Recognition Technology (FERET) databases. The performance of the probabilistic matching technique over standard Euclidean nearest-neighbor eigenface matching was demonstrated using results from DARPA's 1996 "FERET" face recognition competition, in which this Bayesian matching algorithm was found to be performing well.

2.4 Issues in Face recognition

Despite the successes of some face recognition systems there are many issues remain to be addressed. Among those issues the following two are prominent for most systems:

- Pose
- Illumination

Difficulties due to illumination and pose variations have been documented in many evaluations of face recognition systems [Adnin et al. 1997]. It is more difficult to solve when both pose and illumination variations are combined. These problems are difficult to eliminate in some situations where face images are acquired in uncontrolled environments, for instance, in surveillance video clips.

The pose problem occurs where the same face appears differently due to changes in viewing condition. However the pose problem is not discussed in detail, hence it's not within the scope of our research.

The illumination problem occurs where the same face appears differently due to the change in lighting. More specifically, the changes induced by illumination could be larger than the differences between individuals, causing systems based on comparing images to misclassify the identity of the input image [Romdhani et al. 2002]. This has been reported in with a dataset of 25 individuals. The conclusions suggest that significant illumination changes cause dramatic changes in the system, and will reduce the performance of subspace-based methods [Romdhani et al. 2002, Wang et al. 2003].

As a fundamental problem in image understanding literature, illumination is generally quite difficult and has been receiving consistent attentions [Chennubhotla et al 2002, Dinggang and Horace 1997, Finlayson et al. 1998, Phillips et al. 2005]. Alper Yilmaz proposed a new approach to overcome the problems in face recognition associated with illumination changes by utilizing the edge images rather than intensity values [Yilmaz and Gokman 2001]. The methodology introduced “hills” which obtain by covering edges with a membrane. Each hill image is then described as a combination of most descriptive eigenvectors, called “eigenhills”, spanning hills space when they are projected into a graph. This approach is based on the hypothesis that edges do not change considerably in varying illumination. However, edges bring their own problems; they are very sensitive to pose and orientation changes of the face. To overcome these problems edges are converted with a membrane, which is related to regularization theory. Comparison of recognition performances of eigenface, eigenedge and eigenhills methods by considering illumination and orientation changes showed that eigenhills approach performs will. However, a drawback of edge-based approach is the locality of edges. Any change in facial expression or a shift in edge locations due to small rotation of the face will degrade the recognition performance.

Within the eigen-subspace domain, it has been suggested that by discarding the three most significant principal components, variations due to lighting can be reduced and it has been experimentally verified in that discarding the first few principal components seems to work reasonably well for images under variable lighting [Belhumeur et al. 1997]. However, in order to maintain system performance for normally lighted images, and improve performance for images acquired under varying illumination, an assumption has to be made that the first three principal components capture the variations only due to lighting.

To handle the rotation problem, researchers have proposed multiple images based methods when multiple images per person are available [Georghiades et al. 1999, Beymer 1997]. Beymer proposed a template based correlation matching scheme. In this work, pose estimation and face recognition are coupled in an iterative loop [Beymer 1997]. For each hypothesized pose, the input image is aligned to database images corresponding to a selected pose. The main restrictions of this method are

- Many images of different views per person are needed in the database.
- No lighting variations (pure texture mapping) or facial expressions are allowed.
- The computational cost is high since it is an iterative searching approach.

More recently, an illumination-based image synthesis method [Georghiades et al. 1999] has been proposed as a potential method for robust face recognition handling both pose and illumination problems. This method is based on the well-known approach of an illumination cone [Belhumeur and Kriegman 1996] and can handle illumination variation quite well. To handle variations due to rotation, it needs to completely resolve the GBR (generalized-bas-relief) ambiguity when reconstructing the 3D shape.

2.5 Lighting Invariance in Face Recognition

Lighting variations can be broadly classified into two categories: global intensity changes and localized gradients. Global intensity changes are lighting variations which affect the entire face. Localized gradients, on the other hand are more difficult to remove. Such lighting effects are caused by shadows, directional and specular lighting and require non-linear operation for compensation.

We have observed a range of face image processing techniques as potential pre-processing steps, which attempt to improve the performance of the eigenface method of face recognition and various other face recognition techniques [Chennubhotla et al 2002, Dinggang and Horace 1997, Finlayson et al. 1998, Phillips et al. 2005]. Even when there are only the illumination changes, its effects override the unique characteristics of individual features and thus greatly degrade the performance of state-of-the-art face recognition systems.

If our system were to be used in real-world environments, under varying light conditions, then it must be able to overcome irregular lighting. Varying illumination is one of the most difficult problems and has received much attention [Sim et al. 2002, Epstein et al. 1995, Wang and Wang 2003] in recent years. As described in section 2.4 it is know that the variation due to lighting changes is larger than that due to different personal identity. Because lighting direction changes alter the relative gray scale distribution of faces, the traditional histogram equalization method used in image processing and face detection for image normalization only transfers the holistic image gray scale distribution from one to another [Jain 1989]. This processing ignores the face-specific information and can not normalize these gray level distribution variations. To deal with this problem, researchers have made many breakthroughs in recent years.

Adini has compared different face representations, such as edge map, image intensity derivatives, and images convolved with 2D Gabor-like filters, under lighting direction changes [Adnin et al. 1997]. Their results demonstrated that none of these algorithms were robust to variations due to light direction changes. The main drawback of this kind of approaches is that the most valuable information, gray value, is discarded and person's discriminative information in face image is weakened in perusing so called "illumination invariant features".

The illumination Cone method [Belhumeur and Kriegman 1996, Georghiades and Belhumeur 2001] theoretically explained the property of face image variations due to light direction changes. In this algorithm, both self-shadow and cast-shadow were considered and its experiment results outperformed most existing methods. The main drawbacks of illumination cone are the computational cost and the strict requirement of seven input images per person.

2.6 Face detection and Recognition in a Video Sequence

Main steps in face recognition in a video sequence are,

- Video segmentation
- Key- frame extraction
- Face detection
- Face recognition

Since the accuracy of video segmentation affects the face detection and identification, several improvements have been reported, which combines temporal segmentation or tracking with spatial segmentation or manual segmentation [Calic and Izquierdo 2001, Calic and Thomas 2004, Calic and Izquierdo 2002].

A video sequence consists of a set of temporally ordered frames that, when shown sequentially, the Human Vision System interprets as a moving image. Neighboring frames are often similar, especially when a high number of frames per second was captured, leading to computational and perceptual difficulties. As Human understanding corresponds better to smaller and more semantic units and themes, a four-level hierarchy illustrated in Figure 2.3.

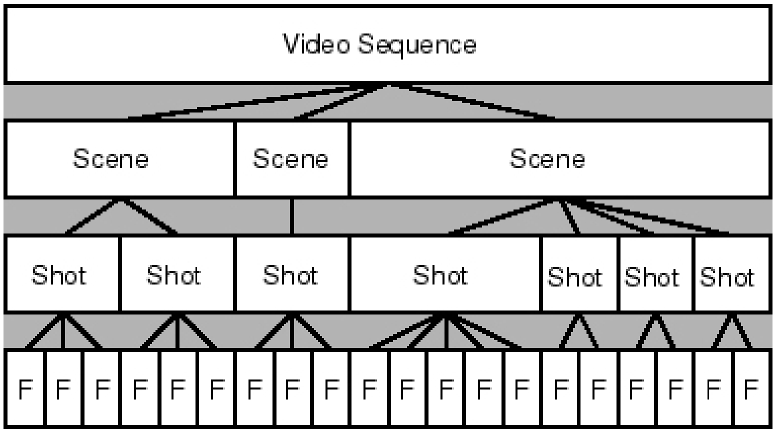


Figure 2.3: The four-level video structure.

- Scene: a sequence of concatenated-by-editing shots captured from the same location or at the same time.
- Shot: a clip that recorded continuously without breaks.
- Frame: an atomic unit in the temporal domain and cannot be further divided.

At the lowest level, the set of frames, a physical sequence is implemented. A *frame* is an atomic unit in the temporal domain and cannot be further divided. A *shot* is a group of frames that are captured continuously from the same camera without interruption. Shots are prevalent in highly structured video domains, such as newscasts, adverts, drama, entertainment, but less so in other domains such as sport and surveillance. However, for semantic-sensitive applications, shots still present a too low-level unit for Human understanding. Shots are therefore grouped into *scenes*. A scene is a set of shots that exhibit a common semantic, thread or story-line structure [Christel et al. 2000]. As shots and scenes have the same physical structure, they both consist of a group of neighboring frames; the generic term *segment* is used.

Though many research efforts have been devoted to video segmentation algorithms, most of them focused on shot or scene boundary detection [Bimbo 2000, Boreczky and Rowe1996, Gunsel et al.1997]. Some literatures addressing semantic video segmentation with different visual features, but these methods are more like shot grouping.

MPEG standard provides users to transmit, retrieve, download, store, and reuse arbitrarily shaped semantic video objects efficiently and also interact with media sources [Calic and Izquierdo 2001, Graves and Lalmas 2002]. However, MPEG doesn't provide concrete techniques for semantic video object extraction. But it's an indispensable process for many digital video applications. Most existing automatic semantic video object extraction schemes use motion information in video sequences as an important cue to produce semantic objects. Based on how the motion information is used, we can divide most current methods into three categories:

- Temporal segmentation,
- Spatial segmentation and temporal tracking, and
- Spatio-temporal segmentation.

Temporal segmentation only uses motion information deduced from consecutive frames and doesn't consider spatial information. For instance, Wang and Adelson [Wang. and Adelson 1994] employed the motion estimation, motion segmentation, and temporal integration to obtain video objects. To improve accuracy, spatial segmentation based on color and texture can be applied. One way is to perform a spatial segmentation for the initial frame and temporal tracking for the successive frames. Another way to improve accuracy is to impose spatial segmentation on each frame to modify the temporal segmentation result. In addition to fully automatic methods, researchers have also studied semi-automatic techniques with user interaction. Fuhui Long presented accurate and user-interactive semantic video object extraction system [Long et al. 2001]. The system adaptively performs spatial and temporal segmentation when necessary. To achieve this, their system detects the variations between successive frames, in addition, the system provides a flexible switch between the user-interactive and fully automatic extraction modes. User interactions can be imposed, removed, or changed in the automatic extraction process at any time.

These methods are successful to some extent. To benefit users, a good extraction method should be accurate, user interactive, and simple. Accuracy is an essential requirement. An inaccurate semantic video objects containing parts of the background or losing its own parts can hardly be reused in content-based applications. Nonetheless, semantic video objects that most methods produce aren't accurate enough at boundaries, especially for video sequences containing complex background and motion.

In the context of video structuring, indexing and visual surveillance, faces are important, because it is a unique feature of human beings. Faces can be used to index and search the video databases and classify video scenes [Lorente and Torres 1998]. Therefore, research on face detection and recognition is critical in video database applications. However, in general video databases, there is little or no constraint on the number, location, size, and orientation of human faces in the scenes. Because of these issues, successful face detection and recognition becomes important and challenging before the indexing, search, and recognition of the faces could be done. Face recognition in video sequences often involves four important steps:

- Face detection.
- Feature extraction.
- Recognition.

It is clear that the large amount of data involved in video sequences represent a challenge for real-time implementation of these three steps. Most Approaches to face detection and recognition in a Video Sequence are the same techniques mentioned above in Chapter 2.2 and 2.3. Therefore in the following paragraphs we will explore how those techniques are extended into a sequence of video frames. .

The development of standards for video encoding such as the MPEG family coupled with the increased power of computing has resulted that content-based manipulation of digital video information is now possible. Hualu Wang and Shih-Fu Chang have proposed a fast algorithm that automatically detects human face regions in MPEG video sequences [Wang and Chang 1996]. The processing unit of MPEG standards is the macroblock (16x16 pixels), so that the bounding rectangles of the detected face regions have a resolution limited by the boundaries of the macroblocks. Their algorithm takes the Discrete Cosine Transformation (DCT) coefficients of the macroblocks of MPEG frames as input and generates positions of the bounding rectangles of the detected face regions [Kobla et al. 1997]. In order to detect faces using the DCT coefficients, only minimal decoding of the compressed video sequence is required. The DCT coefficients can be obtained easily from I-frames of MPEG videos. This algorithm consists of three stages, where chrominance, shape, and DCT frequency information are used respectively. Bayes decision rule is applied to MPEG video streams, and classify each MPEG macroblock as a candidate face macroblock or a non-face one. They used rectangles to approximate face regions, and use locations of rectangles as the boundaries of faces.

The considered shape constraints are:

- Faces are contiguous regions that fit well in their bounding rectangles, whether the face is front view or side view, or whether the head is upright or a little tilted.
- The size of the bounding rectangles is bounded by the lower limit of face detection and the size of the video frames.
- The aspect ratios of the bounding rectangles should be in a certain range.

At this stage, face detection becomes the task to search for face-bounding rectangles that satisfy the above constraints. To limit the search area for matching, they detect non-overlapping rectangular regions that cover contiguous face macroblocks.

They have tested the algorithm on 100 I-frames from a MPEG-compressed CNN news video which included news stories, interviews, and commercials. The algorithm success rate is 92%, including faces of different sizes, frontal and side-view faces, etc. The run time of the algorithm ranges from 1 to 14 milliseconds per frame on a SGI ONYX workstation, depending on the complexity of the scenes in the video frames.

Hence this algorithm can be performed in real time with several restricted aspects. It can only be applied to color images and videos. False dismissals can not be totally avoided. There were still false alarms even after applying the shape and energy constraints.

Some effort is being conducted in face recognition and video segmentation within the activities of the new standard MPEG-7 (Multimedia Content Description Interface) [Lorente and Torres 1998]. Their key objective was to develop a tool to be used in the MPEG-7 standardization effort to help video indexing activities. They propose a Principal Component Analysis (PCA) for face recognition [Turk and Pentland 1991]. Lorente and Torres have extended the eigenface concept to certain parts of the face: eyes (left and right eigeneyes), the nose (eigennoses) and the mouth (eigenmouth). They have also introduced the new concept of eigenside (left and right), which are eigenfaces generated from the left and right sides of the face.

In this method, it is difficult to avoid certain limitations when some parts of the faces are occluded and when some conditions such as lateral lighting or facial expression change along the face. Tests using the four point model have been conducted with the MPEG-7 test sequences. Although the results are still in a preliminary stage, they show that the taken approach will be helpful for the video indexing application.

Michael C. Lincoln and Adrian F. Clark of the University of Essex have proposed a scheme for pose independent face identification in a video sequences [Lincoln and Clark 2000]. They propose an “unwrapped” texture map, constructed from a video sequence using a texture-from-motion approach. They consider an image that is a projection of the head shape onto a notional cylinder rather than onto a plane. They term it as an “unwrapped texture map”. Their scheme involves taking each image (planar projection) in a video sequence, tracking the head from frame to frame and determining the head orientation in each frame, then merging the appropriate region of the image into the unwrapped texture map. If the head exhibits a reasonable amount of motion, a fairly complete texture map can be accumulated.

The position and orientation of the head in the first frame of a sequence is currently specified manually, where it can be automated. In the Next Frame, an estimate for the head's new position and orientation is made. The head model is transformed to this new position and the image texture back-projected onto it. A match with the reference head texture is then performed. The six positions and orientation parameters of the head model are adjusted using a simplex optimization scheme until the best (smallest) match value is obtained.

Strong directional and varying light sources can adversely affect tracking. This is avoided by making the assumption that illumination varies slowly compared to the frame rate of the video. The approach to building texture maps appears to be reasonably effective, face-feature normalization and a more sophisticated classifier have not yet been included in this scheme.

Jeffrey S. Norris has developed a vision-based door security system at the MIT Artificial Intelligence Laboratory [Norris 1999]. Faces are detected in a real-time video stream using an algorithmic approach.

The basic steps that of the algorithmic approach is as follows:

- At startup, record several frames of the camera's input and average them.
  Store this average image as the background image for further steps.
- Capture an image and determine if a person is likely to be present by estimating how different the image is from the background.
- Subtract the current image taken by the camera from the background image and apply a threshold to produce a binary difference image where white corresponds to areas that differ greatly from the background.
- Apply an image morphological "erode" operation to remove artifacts in the difference image due to camera noise. Also remove highly unlikely regions from consideration.

- Locate the top of the largest white region and trace the contour of the head portion of the region by performing a series of line-based morphological "close" operations and then finding the extents of these lines.
- Grab the region of the original image that we now believe to be a face.

One great benefit of a reliable algorithmic approach such as this is that it can be used to bootstrap many learning methods. For instance, a face database generation system can be set up in a novel environment, and begin to find faces of pedestrians by relying entirely on this algorithmic approach. A step following this algorithm can reject all images except those that are certainly faces, and attempt to cluster the acquired data into likely face classes. Faces are recognized by using principal component analysis with class specific linear projection. This system is called as the "Gatekeeper" and the software for the Gatekeeper was written using a set of tools created by the author to facilitate the development of real-time machine vision applications in Matlab, C, and Java.

A modified Karhunen-Loève transform is defined with the aid of an automatic feature selection procedure which is used for feature extraction and face recognition from video sequences [Campos et al. 2000]. Face detection is performed by using a statistical skin-color model to segment the candidates face as well as a simple correlation procedure to verify the presence or absence of a face. Faces are tracked in a video sequence using Gabor Wavelet Networks (GWN) [Krüger 2000]. Basically, the idea of GWN is to represent a face image as a linear combination of 2D Gabor wavelets, whose parameters (position, scale and orientation) are stored in the network nodes, while the linear coefficients are represented as the syntactical weights. This approach considers the overall geometry of the face, thus being robust to deformations such as eye blinking and smile, which is usually a critical situation to most local-based traditional methods.

GVF-Snake (Gradient Vector Flow-Snake), where Gradient Vector Flow of the optical flow field is used as a component of the energy function to be minimized, to segment out the face from video sequence [Biswas and Pandit 2002]. GVF-Snake has a particular property that it can capture concave boundaries [Xu and Prince 1998]. The optical flow was calculated for all the frames of the video sequence. The basic assumption is that even if there is background movement, the optical flow of foreground pixels is appreciably different from the optical flow of the background pixels. This method gives satisfactory result even when background is not stationary or the background contains other compact objects. The snake inflation technique enables face segmentation in subsequent frames. The initial contour which uses an edge map of the video frame has to be selected very carefully. If the initial contour unable to detect the entire region of interest then the segmentation will be unsuccessful. Further development is needed to overcome of this problem.

Shin'ichi Satoh, Yuichi Nakamura and Takeo Kanade proposed a system which associates faces and names to news videos, by integrating face-sequence extraction and similarity evaluation, name extraction, and video-caption [Satoh et al. 1996]. The primary goal is to associate faces and names of persons of interest in news video topics. The system employs face detection and tracking to extract face sequences and natural-language processing techniques using a dictionary, thesaurus, and parser to locate names in transcripts. Since transcripts don't necessarily give explanations of videos, no straightforward method exists for associating faces in videos and names in transcripts. So they assume that a corresponding face and name are likely to coincide and may be an associated face-name pair. But some difficulties exist in associating faces and names: the lack of necessary faces or names and possible multiple correspondences of faces and names.

The system employs video-caption recognition to obtain face-name association. Video captions are superimposed text on video frames, therefore representing literal information. Because video captions don't necessarily appear for all faces of persons of interest, they use the video captions as supplements to the transcripts. Finally, results obtained by these techniques should be integrated to provide face-name association.

The first step is to employ face detection and tracking to detect face sequences in videos. Face tracking consists of three components face detection, skin-color model extraction, and skin-color region tracking [Kuchi et al. 2002]. To enhance the face similarity evaluation, the most frontal view of a detected face sequence is needed. To choose the most frontal face from all detected faces, the system first applies a face-skin region clustering method. For each detected face, cheek region which presumed to have skin color is located by using the eye locations of the detected face. To evaluate face similarity, the eigenface-based method is used [Turk. and Pentland 1991]. Finally, given videos as input, the system outputs a two-tuple list: timing information (start-end frame) and face identification information.

They implemented the system on an workstation and processed 10 “CNN Headline News” videos (30 minutes each) for a total of five hours of video. The system extracted 556 face sequences from the videos. Although the correct answers acquire higher ranking, the results might be recognized as imperfect due to many incorrect candidates within the top four results. However, when recalling, the system extracts face and name information and combines these unreliable sets of information to obtain face-name association, inevitably the results contain unnecessary candidates.

However, they fail to infer which word actually coincides with the face sequence. The main reason for this is the fact that transcripts don’t explain videos directly. To overcome this problem, the system may need in-depth transcript recognition, as well as in-depth scene understanding, and a proper way to integrate these analysis results. This system achieves an accuracy of 33 percent in face-to-name retrieval and 46 percent in name-to-face retrieval.

Each method includes several image processing techniques: face tracking, face identification, intelligent name extraction using dictionary, thesaurus, and parser, text region detection, image enhancement, character recognition, and the integration of these techniques. One main drawback of the system is that they use a skin color model to extract face in the video frames to identify people. Since the skin color model is very sensitive to color changes in faces, the face extraction will not be done successfully.

A performance comparison of the EGM approach with a system comprised of the PCA based feature extractor and a nearest neighbor classifier [Zhang et al. 1997]. Results on a combined database of 100 people showed that the PCA based system was more robust to scale and rotation variations, while the EGM approach was more robust to position variations. This report contributed the robustness to illumination changes to the use of Gabor features, while the robustness to position and expression variations was contributed to the deformable matching stage. However, one drawback is that the performance is very dependent on the high number of input parameters that have to be set. The main drawback of this approach is that this information is encoded in a kind of black box that does not allow easily to analyze how it works. Other approaches based on probabilistic structures such as PCA have the ability to express in a format directly comprehensible for researchers how the knowledge is represented.

Though tracking and recognizing face objects is a routine task for humans, building such a system is still an active research. Appearance-based approaches to recognition have made a comeback from the early days of computer vision research and the eigenface approach to face recognition may have helped this come about. Among the best possible known approaches for face recognition, Principal Component Analysis (PCA) has been an area of much effort and it appears that eigenfaces is a fast, simple, and practical algorithm. In addition, eigenface recognition method has several advantages:

- Raw intensity data are used directly for learning and recognition without any significant low-level or mid-level processing.
- No knowledge of geometry and reflectance of faces are required.
- Data compression is achieved by the low-dimensional subspace representation.
- Recognition is simple and efficient compared to other matching approaches.

Nevertheless, as far as recognition in video sequences is concerned, much work still remains to be done.

In summary, it is been considered that eigenfaces is a fast, simple, and practical algorithm. However, it is of limited use because its performance depends on a high degree of correlation between the pixel intensities of the training and test images. This limitation can be addressed by using extensive preprocessing to normalize the images. In our research we have explained a possible way to overcome this limitation by introducing a normalizing algorithm.

Chapter 3

3 System Design

3.1 System Architecture

In this chapter the overall architecture of the proposed system, the techniques and to develop the individual components and one of the major problem encountered which resulted into this research is explained. The overall architecture of the system is shown in Figure 3.1. The main components of the system are the keyword extractor, keyword organizer, feature extractor, profile creator and the query processor.

Various types of course materials such as course notes, PowerPoint presentations, quizzes, past examination papers and video clips are the main inputs to this system. The system stores these educational materials in a multimedia server. The keyword extractor extracts keywords from the main course materials. The keyword organizer assists the construction of an ontology in a database out of the keywords generated by the keyword extractor. The feature extractor extracts audio and video features from the video clips and the profile creator creates profiles of presenters from the information generated by the feature extractor. These profiles are then used to create indices on the video clips. Finally the query processor process enables the end users to browse and retrieve educational material stored in the object server by using the ontology and the indices.

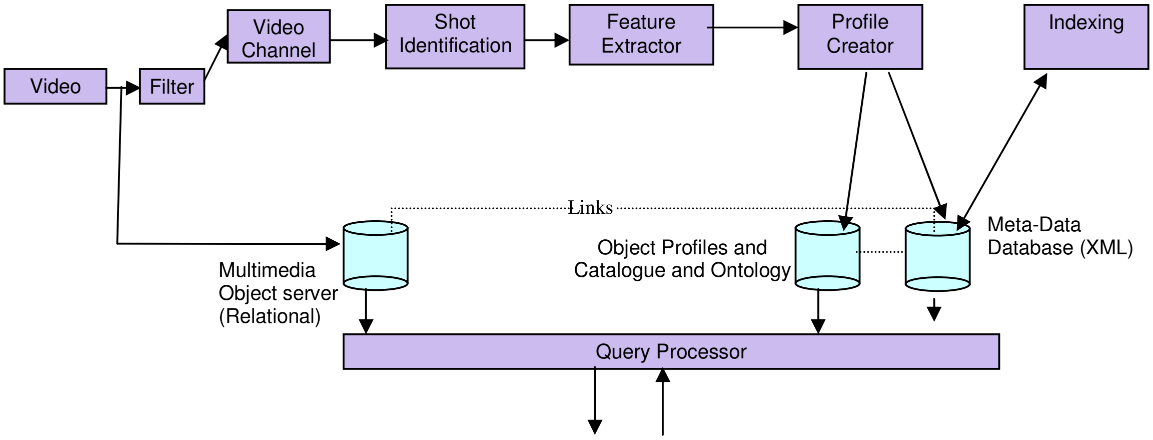


Figure 3.1: System architecture

In the system, a video is analyzed by segmenting it into shots, selecting key-frames from each shot, and extracting audio-visual features from the key-frames (Figure 3.2). This allows the video to be searched at the shot-level using content-based retrieval approaches. The scope of the research is to improve presenter recognition in video key frames. The following sections are focused on the aspect relevant to this research.

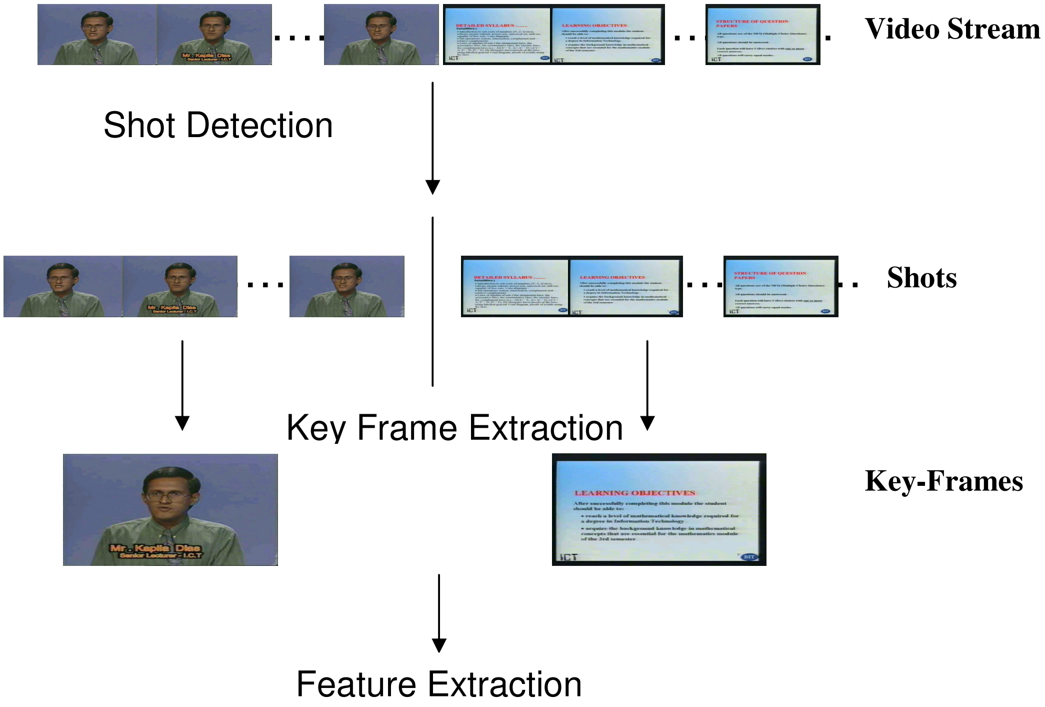


Figure 3.2: Segmentation of video clips

3.1.1 Video Segmentation

The goal of semantic segmentation is to partition the raw video into shots [Kobla et al.1997]. Video segmentation can be done either manually or automatically. Manual segmentation is usually time-consuming but more accurate. Many approaches to automate segmentation of video sequences have been proposed in the past [Yeo and Liu 1995, Zabih et al. 1995, Zhang et al. 1993]. Most of these approaches exploited the motion information in order to extract moving objects from a scene [Yeo and Liu 1995]. Few of the contemporary techniques have merged motion information with information obtained from edge extraction and/or texture analysis to increase the accuracy [Zabih et al. 1995, Zhang et al. 1993].

The color histogram-based shot boundary detection algorithm is one of the most reliable variants of histogram-based detection algorithms. The color histogram is computed of each frame of the video. Each pixel has Red, Green, and Blue components. First the pixel values in the RGB space are converted into YCbCr color space [Pei and Chou 1999]. A color histogram of an image displays the combined frequency of Y, Cb, and Cr channels. The color histograms for the entire video frames are computed [Zabih et al. 1995, Zhang et al. 1993]. Then the difference between the histograms of consecutive video frames is computed. A video shot boundary is detected if color histograms of neighbored video frames (i-1) and i differ to an extent greater than a pre-defined threshold T and no video shot detected if histogram difference in video frames (i-1) and i is less than threshold T [Dongge and Sethi 1999] (Figure 3.3). Whenever the difference between the histogram values crosses T, that point is identified as a boundary between two shots. The technique is based on the assumption that the color content does not change rapidly within but across shots. Thus, hard cuts and other short-lasting transactions can be detected as single peaks in the time series of the difference between color histograms of contiguous frames. However, this method is ineffective to fade and dissolve transactions.

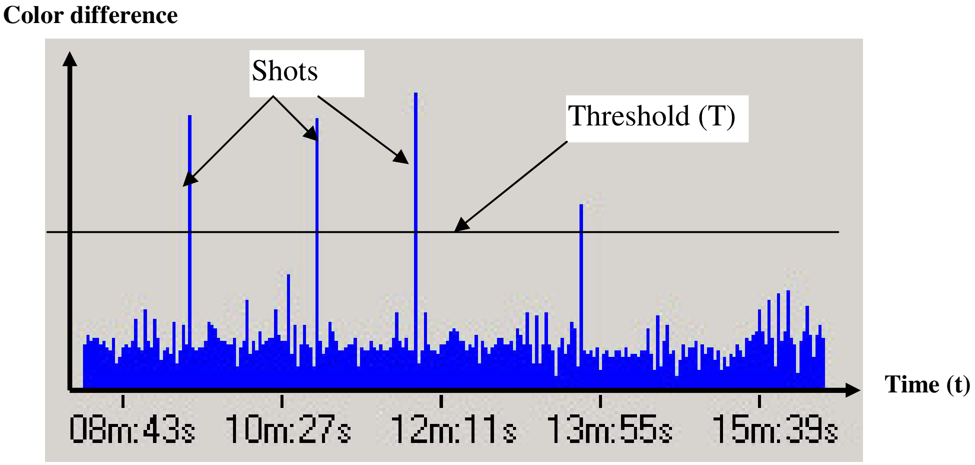


Figure 3.3: Color histogram-based shot detection

The principle behind the edge detection approach is that it can counter problems caused by fades, dissolves and other transactions which are invariant to gradual color changes. Like the color-based method, this also requires minimum difference between adjacent frames to detect a shot cut.

We have primarily investigated models that apply broadly to video content within our scope such as presenter vs. slide show, change of presenter and change of lecture etc. Our segmentation process segments the video by applying a hybrid approach based on color histogram and edge detection techniques. Consequently, the process identifies the shot boundary points more accurately.

Analyzing each video segment frame by frame is an exhausting process. Therefore for each shot a few representative frames are selected. These representative frames are referred to as key-frames. Each key-frame represents a part of the shot. Key-frames contain most of the static information present in a shot, so that face recognition process can focus on key-frames only. If the shot is less than 250 frames (10 seconds of PAL video), the center of every shot is picked as a “key frame” [Pei and Chou 1999]. If the shot is longer than 250 frames, the shot is divided into 250 frame segments and frames that are on the boundaries are picked as key frames [Dongge and Sethi 1999]. All key frames are converted into BMP files.

3.1.2 Multimedia Metadata Database

Since we are using MPEG-7 multimedia contents, media descriptions are XML documents which conform to schema definitions expressed with the XML Schemas. As more and more tools and applications producing and processing MPEG-7 compliant media descriptions are emerging we decided to employ an XML database as our multimedia metadata database [Kosch 2002].

In the system, the entire multimedia metadata database is on a XML database. We are using Apache Xindice 1.0 as our metadata database. The MPEG-7 Description Schemes (DS) provide a standardized way of describing in XML the important concepts related to audio-visual content description and content management in order to facilitate searching, indexing, filtering, and access. The advantage of using DS is that, our application does not need to stick to the pre-defined media description schemes. It can flexibly create new description schemes with MPEG-7 DDL, either from scratch or by extending or combining existing schemes. A logically structured multimedia objects are mapped into a hierarchical structure of metadata, such as presenters, text, shots, scenes and key frames in video. This logical structure determines how metadata content are related to multimedia contents. A relational database is used to store the multimedia objects. The idea is to provide facilities for the user to query and easily navigate through the structure of the Educational content. The main inputs to the profile identification and construction process (Figure 3.4) are these key-frames stored in the multimedia database.

3.2 Profile Identification and Construction Architecture

The profile detection and recognition process detects the faces in the key frame and try to match the detected faces with the presenter profiles available in the profile database (Figure 3.4). If the presenter in the key-frame matches with a profile then the system annotates the video shot with the profile identification and maps it with the metadata database. On the other hand, if the current presenter's key-frame does not match with the available profiles then the profile creator will create a new presenter profile and insert it in to the profile database.

Initially the system starts with no profiles in its profile database. Throughout the recognition process, the presenter profile will be created for unknown presenters. During the Video segmentation phase, the system will decode the MPEG video file into video frames and passes the key frames into the face detection and recognition process. The Face recognition process will retrieve the current profiles from the profile database and compare the input face by projecting the input face onto the eigenspace which is constructed from the profiles known to the system [Turk and Pentland 1991]. If it is identified as a known profile then the metadata database is updated appropriately. If the face is discovered as unknown, then the profile creation process allows the user to create a new profile for the new face. Upon construction of this profile it is added to the profile database and the current eigenspace is updated to reflect this new addition. In our research all faces appearing on key frames are considered important and hence are required to be indexed.

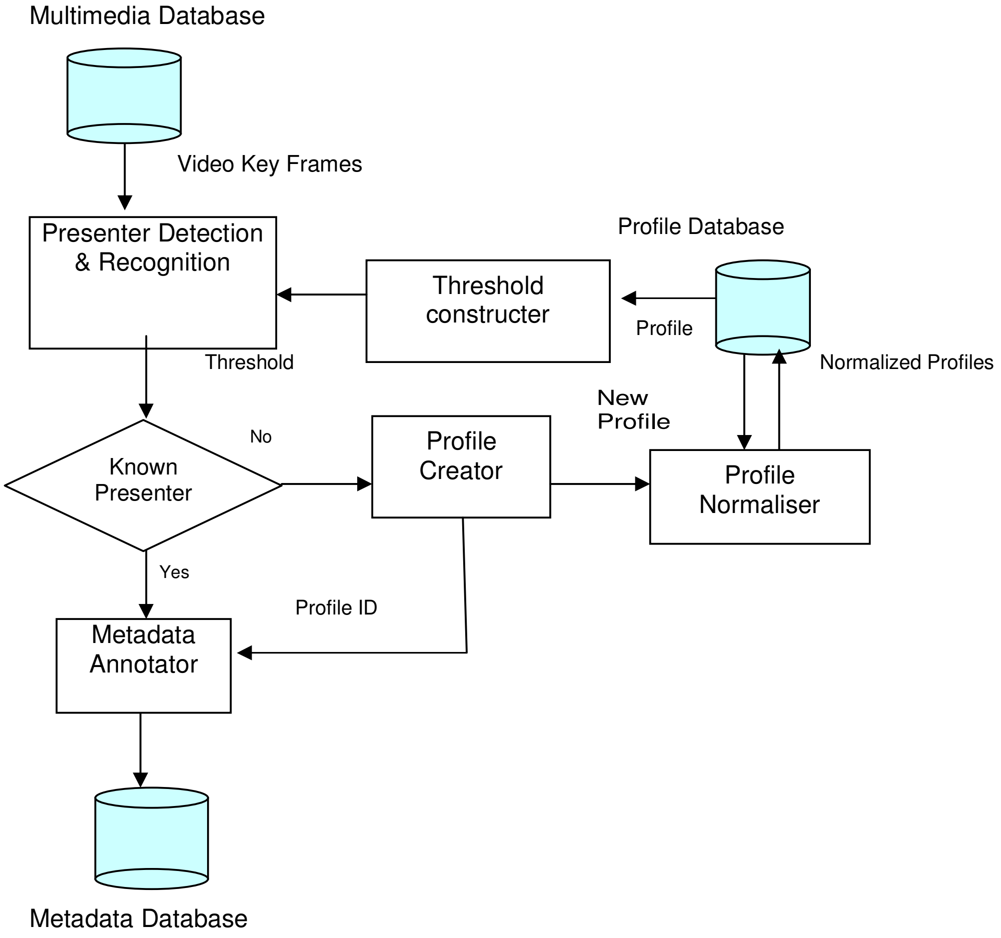
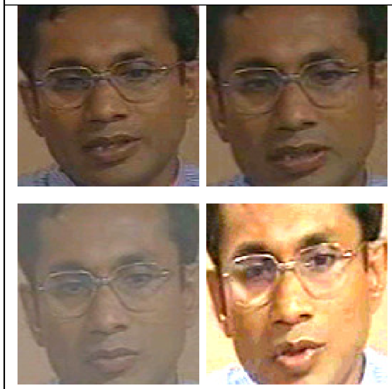
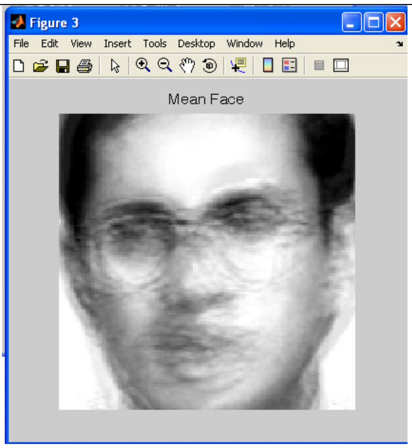
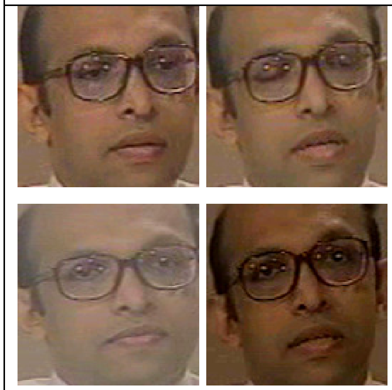
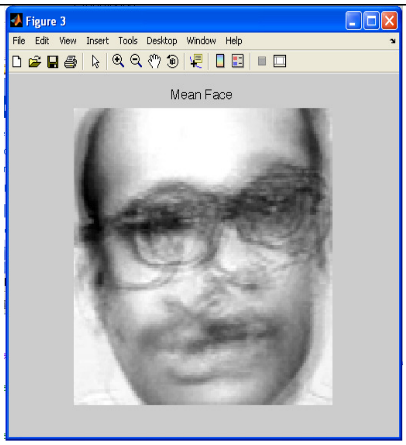


Figure 3.4: Profile construction & recognition architecture

The profile construction is based on PCA [Pentland et al. 1994, Turk and Pentland 1991, Zhang et al. 1997]. The basic idea is to represent presenter's facial features in a transformed feature space. Since the PCA does not correlate individual features at the time of their generation, individual features represented uncorrelated in the eigenspace as well. The feature space comprises of eigenvectors of the covariance matrix of the key-frame features.

In our research we have experimented with two different profile construction approaches. First approach is computing the mean values for each presenter from the available set of the relevant key-frame to determine eigenvectors which are considered as a profile. In this approach, it was realized that the presenters were distorted when the presenters are projected into the facespace. Sample faces are shown in figure 3.5.

Input Faces	Average Face
	 Figure 3 File Edit View Insert Tools Desktop Window Help Mean Face
	 Figure 3 File Edit View Insert Tools Desktop Window Help Mean Face

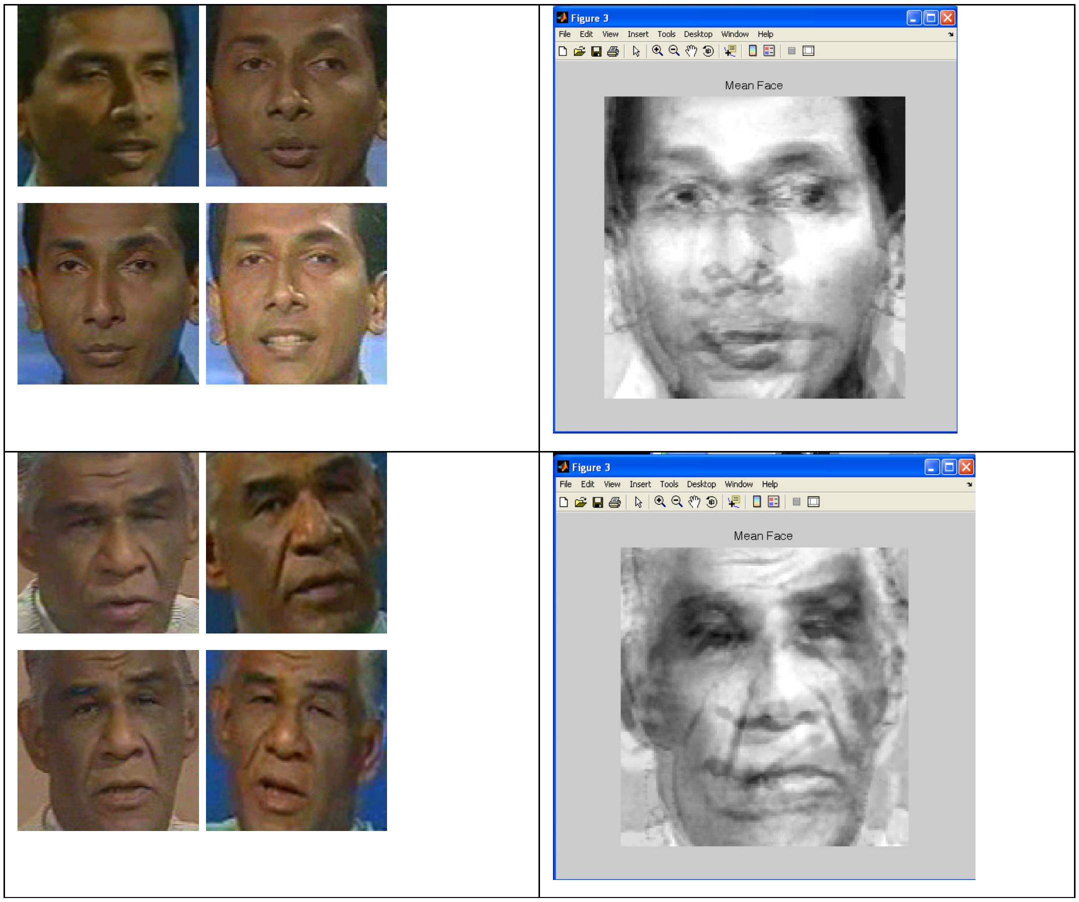


Figure 3
File Edit View Insert Tools Desktop Window Help
Mean Face
Figure 3
File Edit View Insert Tools Desktop Window Help
Mean Face

Figure 3.5: Presenters derived via mean face

3.2.1 Presenter Identification

To execute a complete presenter detection and recognition procedure, the system requires a presenter identification method. Therefore we have studied a few prespective face detection methods which mentioned in chapter 2. After analizing these techniques in detail, we have chosen the method of Paul Viola and Michael Jones to implement the presenter detection process. Motivated by the work of Paul Viola and Michael Jones we use a new representation called an integral image that allows for very fast feature detection [Viola and Jones 2001 b]. We use a set of features which are reminiscent of Haar Basis functions which have been used by Papageorgiou et al. [Papageorgiou et al. 1998]. In order to compute these features rapidly at many scales we used the integral image representation for key frames. The integral image can be computed from an image using a few operations per pixel. Once computed, any one of these Haar-like features can be computed at any scale or location very fast [Kawato and Ohya 2000].

We use AdaBoost to construct a classifier by selecting a small number of important features [Viola and Jones 2001 a]. Feature selection is achieved through a simple modification of the AdaBoost procedure: the weak learner is constrained so that each weak classifier returned can depend on only a single feature. As a result, each stage of the boosting process, which selects a new weak classifier, can be viewed as a feature selection process. The presenter detection process has been implemented without any modification to the above mentioned technique.

The complete face detection cascade has 32 classifiers, which total over 80,000 operations. Nevertheless the cascade structure results in rapid average detection times and the efficiency of the system is good. Figure 3.6 shows some screen shots of presenter detection applied to the key-frames where only one presenter appears from different video shots.

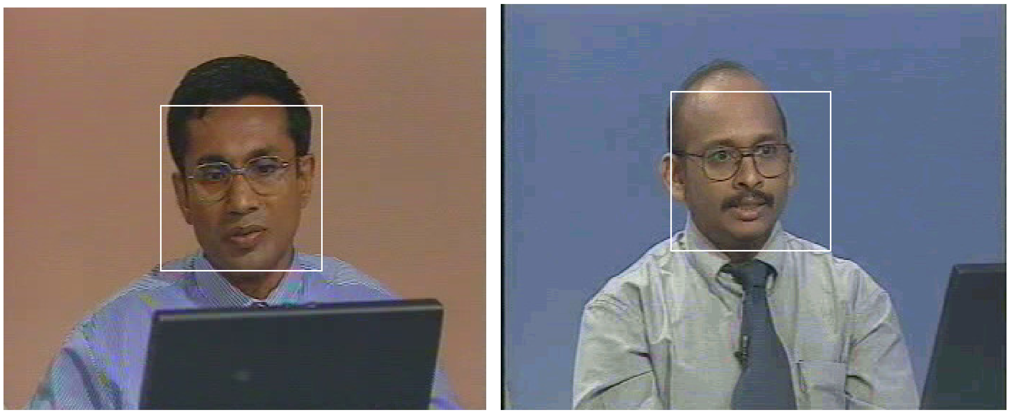


Figure 3.6: Presenter detection in single presenter key-frames

Figure 3.7 shows the results of the application of the chosen technique for presenter detection works on the key-frames where multiple presenters appear in a single video shot. After detecting the faces, the face segments are passed in to the face recognition system based on PCA.

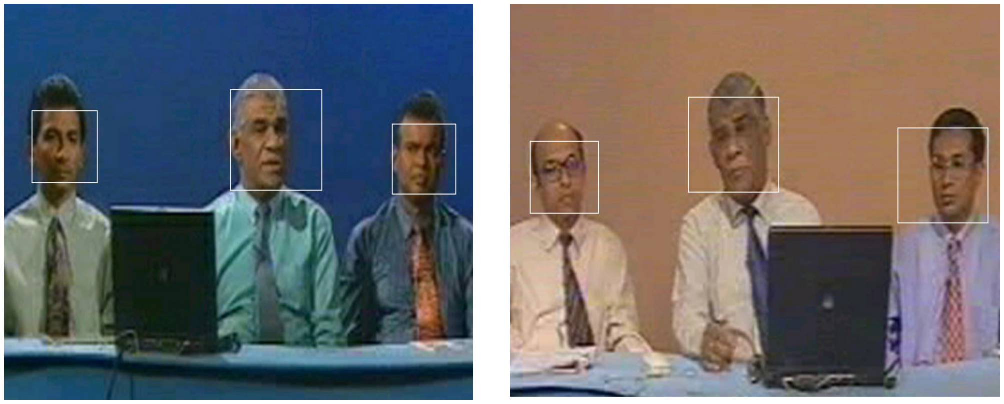


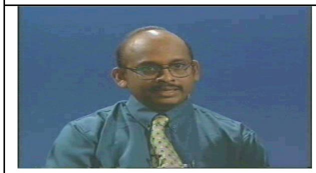
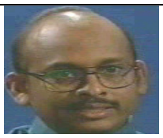
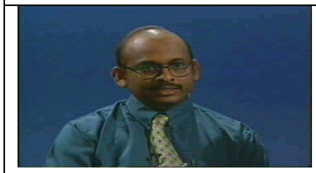
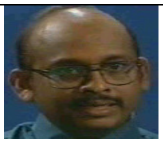
Figure 3.7: Presenter detection in multiple presenters key-frames

3.2.2 Profile Creation

Our profile creation process used in this research is based on Principle Component Analysis (PCA). The system uses a semi-autometic approach which learn on a given set of initial key frames. The algorithm initially uses a set of video shots from the media server to compute the eigenvectors of presenters [Lorente, L., and Torres, L. 1998, Turk and Pentland 1991]. An eigenvector computed for a presenter in this way can be thought as a point in the eigenspace. Due to various reasons the eigenvectors computed for the same presenter by using different video key-frames may result in multiple non equal eigenvectors. These eigenvectors can be thought of as a set of features that together characterize the variation between faces. In such cases, a single eigenvector is created by correlating the individual eigenvectors created for that presenter by considering the fact that faces possess similar structure (eye, nose and mouth position, etc). We store all the eigenfaces computed in this way on the profile database together with additional meta-data about the presenter like name, subject, email etc. The main reasons for using eigenfaces for our research are, its accuracy and its robustness when the faces are described in lower dimensional space.

3.2.3 Profile Normalizer

The main objective of profile normalization is to eliminate key-frame variations such as noise and illumination which effects badly on profile construction. Table 3.1 shows some key-frame extracted from the same presenter in different lighting conditions, their mean intensity and standard deviation. After analyzing the key-frames of the presenter we have observed the values of mean intensities and the standard deviations. Although the there is a considerable variation in the mean intensity values, the standard deviation values have less variation. Through this observation the intention was to buildup a relationship by including these two parameters. Profile normalizer acquires available profiles from the profile database and executes the normalization algorithm and returns the profiles to the database. Since we get key-frames from different lighting conditions we have to have a proper dynamic profile normalization algorithm to maintain the efficiency of the profile matching algorithm. Therefore we concentrate on two descriptors; mean intensity and the standard deviation of the data set that we use to construct presenter profiles. After investigating the variation of the light and the deviation of the mean intensity and standard deviation, we propose an algorithm to normalize the profiles which provide facilities to maintain the accuracy of the system when adding new profiles to the database.

Key-frame	Extracted Face	Mean Intensity	Standard Deviation
		69.4358	26.5136
		105.2734	26.5233

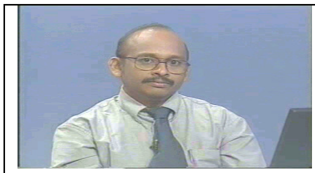
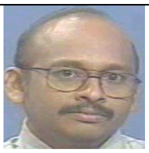
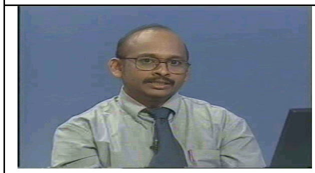
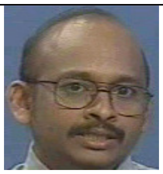
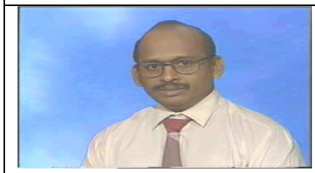
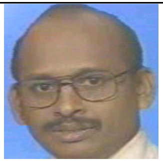
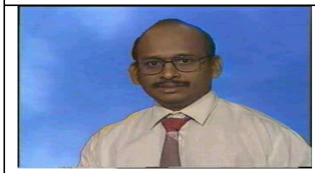
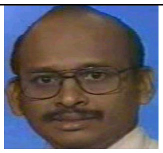
		155.1697	27.0680
		105.2723	24.9650
		136.0883	25.9162
		88.7830	27.2594

Table 3.1: Key-frames with different lighting conditions

3.2.4 Threshold Constructor

For recognition, we employ the Euclidian distance algorithm to compute the similarity of each exciting profile with the input face. As the minimum distance classifier, it works well when the key-frames have relatively small lighting and moderate expression variations. The weakness of this technique is that its performance deteriorates when lighting variations in the key-frames cannot be characterized as small (Table 3.1). This will be further explained in chapter 4. A value to determine known and unknown faces which associated with the minimum distance is called as the global threshold. Initially we put a global threshold manually after analyzing different key-frame sets of faces. Since the lighting variation factor is high, it’s difficult to find such a global threshold for profile recognition. Therefore the threshold levels for the detection and recognition has to be adaptive according to the lighting variations. In our system the threshold constructor will calculate a global threshold according to the profile normalization algorithm.

Chapter 4

4 Profile Construction Algorithm

In the profile construction we have tried out two approaches. These two approaches and the experience we had with these approaches are elaborated in this chapter.

4.1 Initial Approach

Our first approach is to, use the conventional PCA approach to recognize presenters in the key frames mentioned in chapter 2. For each presenter a set of key-frames either from the same video or from separate videos was chosen and it is called as our face database. The corresponding eigenvectors (eigenfaces) are calculated for each key-frame in the database. The faces extracted from the key-frames are projected onto the space. For each presenter we have created a profile as the weighted sum of this eigenfaces generated from the corresponding key-frames. This space is called eigenspace. These weights are used to identify the faces. The low-dimensional representation of faces in the eigenface approach is derived by applying PCA to a representative dataset of key-frames of faces. It should be noted that these features do not necessarily correspond to the facial features such as eyes, nose and ears. They merely capture the features that cause meaningful variations between the faces in the database that allow them to be differentiated. The faces are resized to 128x128 pixels gray scale matrix for recognition.

Using the equation 2.2, for each presenter, the average intensity matrixes are calculated to compute eigenvectors. This is considered as the profile of the presenter. In our initial approach we have constructed profiles by getting the average values of the training set of faces. From the results shown in the chapter 3 (Figure 3.2) it is clear the average faces get distorted and our system performance decreases when we increase the number of video key-frames and when they are captured in different lighting conditions.

4.2 Novel Approach

In the previous section we have explained the problems occurred during the initial experiments. To overcome the problem we have grouped each individual's extracted faces separately. It's considered as a profile (Figure 4.1). The features are stored in the face classes which are considered as presenter profiles. Since individual profiles are stored as separate classes it is easy to classify and insert face features and that is an advantage of using this approach.

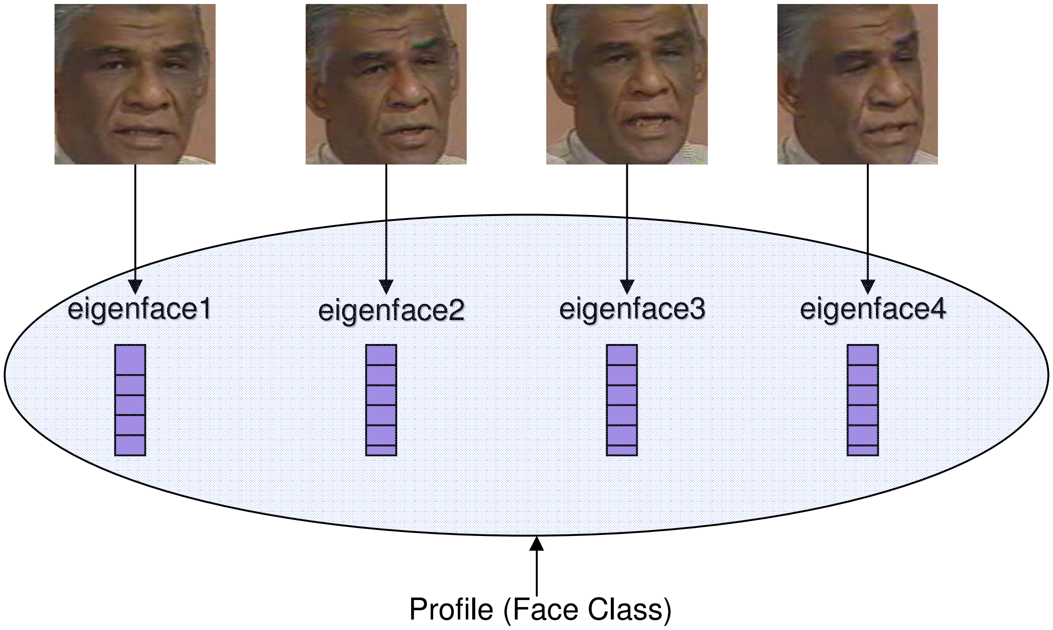


Figure 4.1: A presenter profile

A sample set of 64 key frames are shown below in figure 4.2

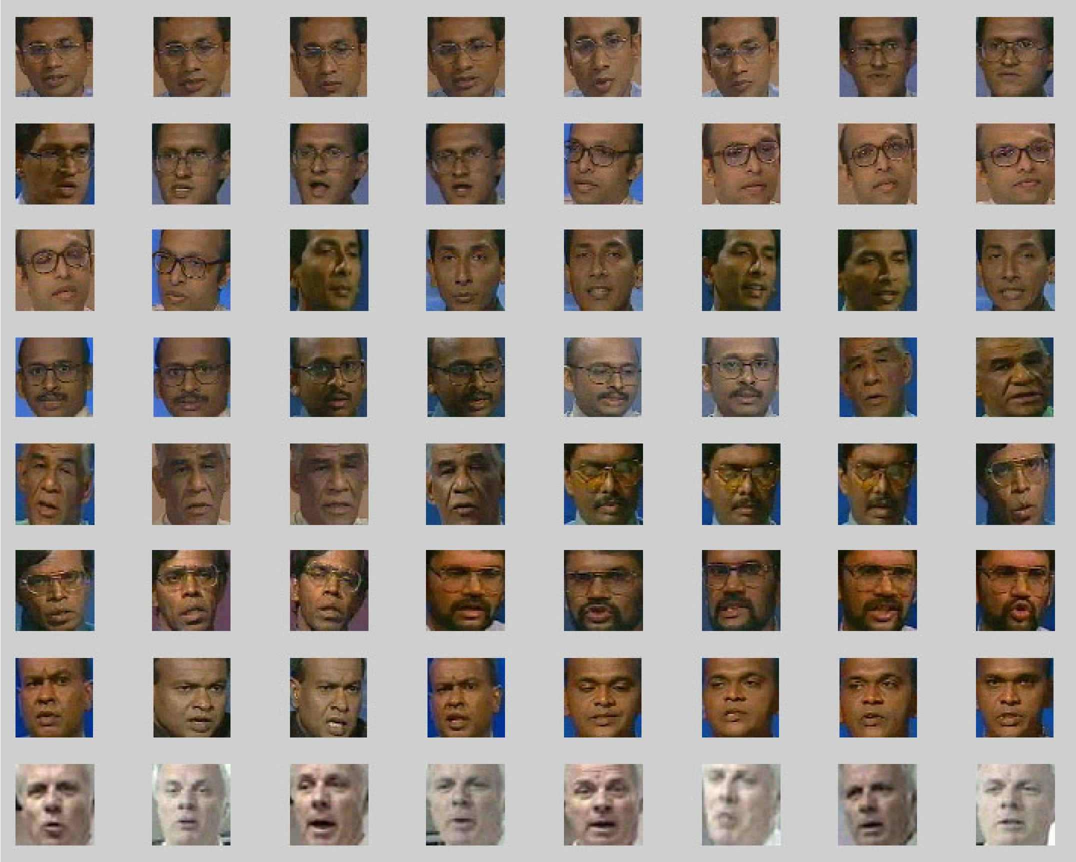


Figure 4.2: Presenters in the face database

The following figure contains a randomly selected 12 eigenfaces calculated from 60 frontal view face frames from 12 presenters (figure 4.3).

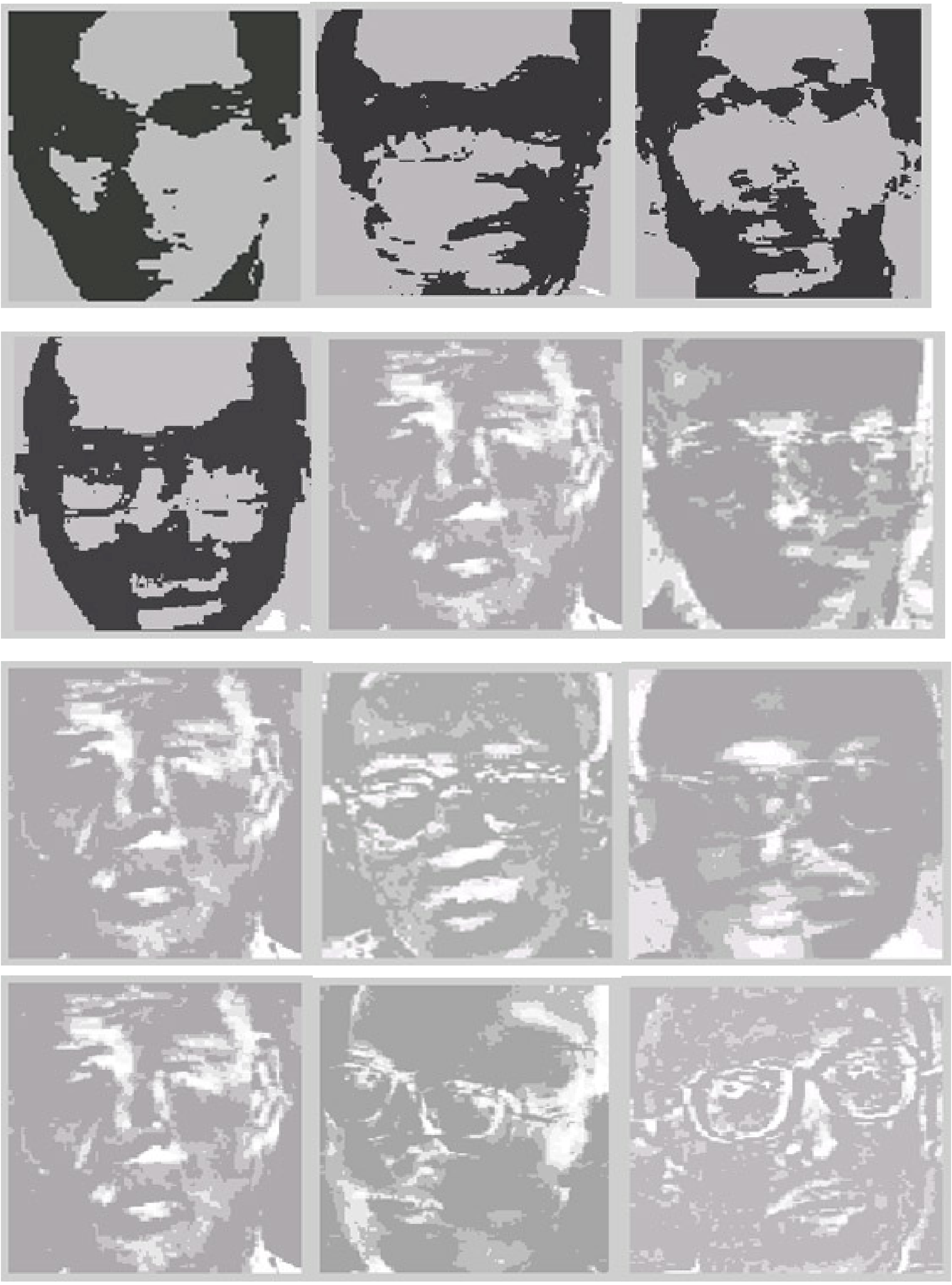


Figure 4.3: Eigenfaces generated from video key-frames

4.3 Profile Overlapping

The effects of illumination changes in key-frames are due to one of the two factors. The inherent amount of light reflected off the skin of the presenter, or the non-linear adjustment in internal camera control. Both of these conditions can have a major effect on facial features recognition [Lorente and Torres 1998]. In our initial profile construction approach lighting variations result in producing similar profiles for different presenters and hence overlap of profiles in the eigenspace (Figure 4.4).

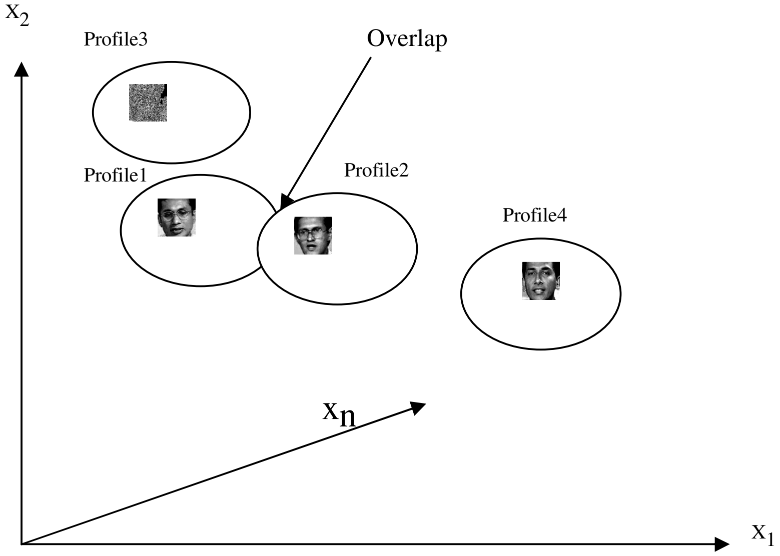


Figure 4.4: Profile overlapping

Figure 4.4 shows an example of profile overlapping, in the figure axis X_1, X_2… X_n represents the n-dimensional eigenspace and the projection of some presenter profile on the eigenspace.

The profile overlapping has resulted in a decrease in the accuracy of the recognition process of the system. This problem can be further illustrated using a sample set of presenters (figure 4.5) and the corresponding Euclidian distance associated with the profiles (4.7). The input face used for recognition is shown in figure (figure 4.6).

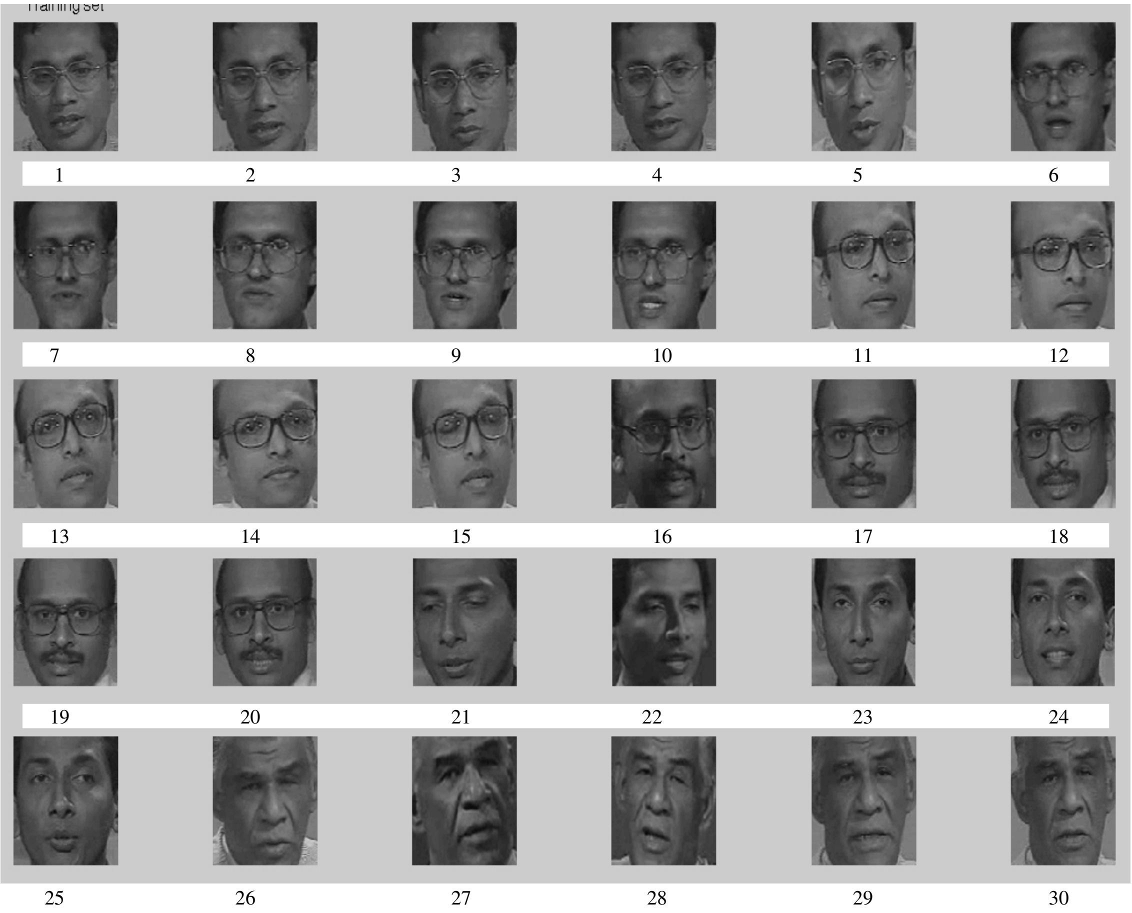


Figure 4.5: Sample presenters

Faces 1 -5 belongs to presenter 1
Faces 6-10 belongs to presenter 2
Faces 11 -15 belongs to presenter 3
Faces 16 -20 belongs to presenter 4
Faces 21 -25 belongs to presenter 5
Faces 26 -30 belongs to presenter 6

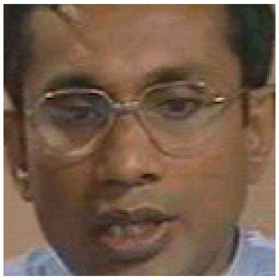


Figure 4.6: Input face 1

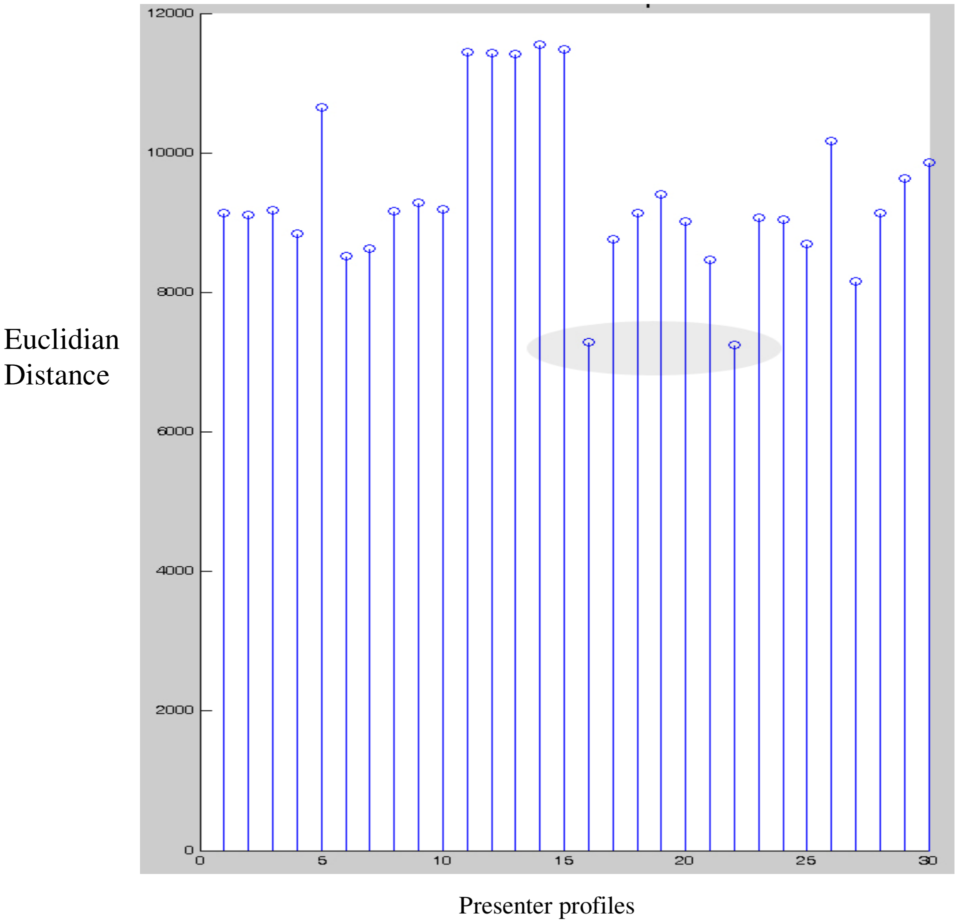


Figure 4.7: Euclidian distance calculation for face 1

1 -5 contains the features for presenter 1
6-10 contains the features for presenter 2
11 -15 contains the features for presenter 3
16 -20 contains the features for presenter 4
21 -25 contains the features for presenter 5
26 -30 contains the features for presenter 6

The Euclidean distance provides a measure of similarity between the presenters and the input face. When the Euclidian distance is calculated, presenters which contain similar features produces minimum distance to the input face as shown in figure 4.7. The input face belongs to the presenter 1 but according to the algorithm, it belongs to the presenter 5. This will create a false recognition as the input face doesn't belong to the correct presenter. Another sample input face and the corresponding Euclidian distance graph is shown in figures 4.8 and 4.9.

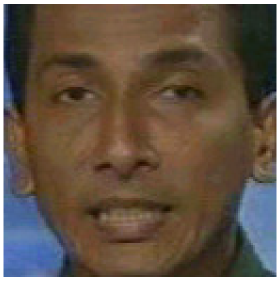


Figure 4.8: Input face 2

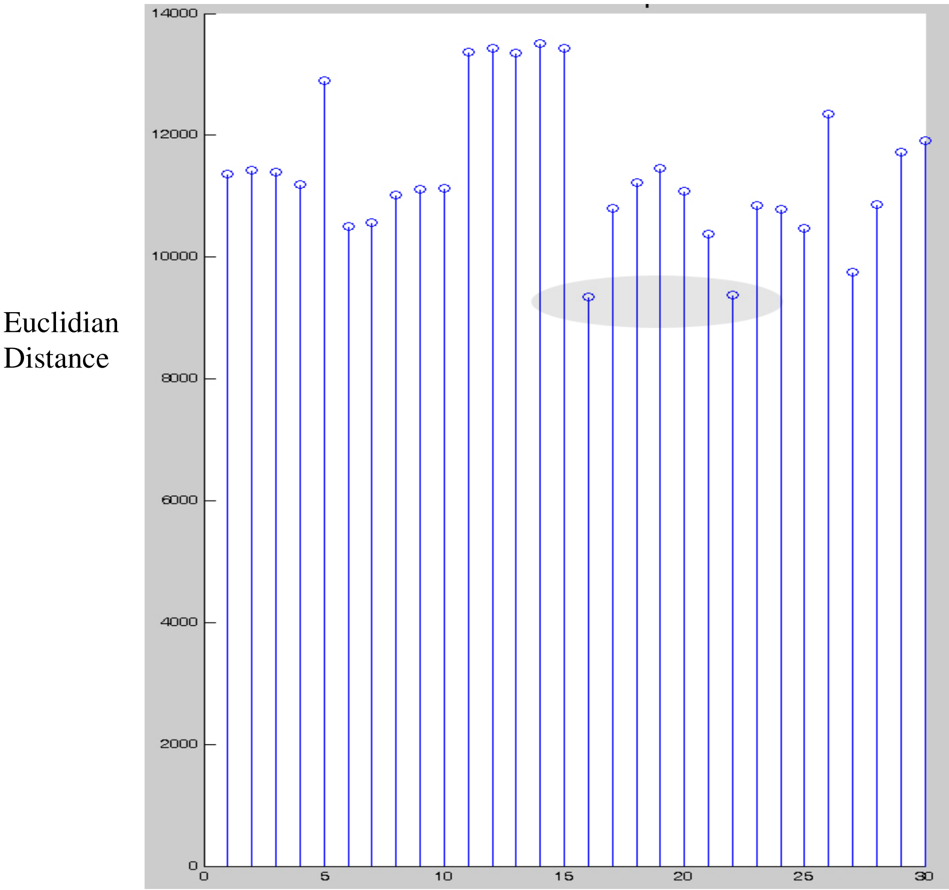
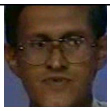
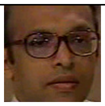
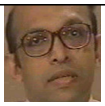
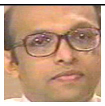
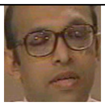
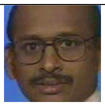
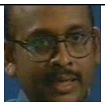
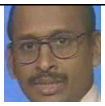
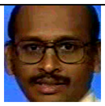
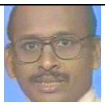
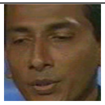
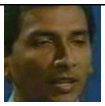
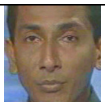
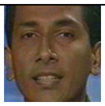
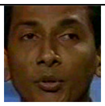


Figure 4.9: Euclidian distance calculation for face 2

Input face 2 in the figure 4.8 belongs to the presenter 5. But according to the graph on figure 4.9, it shows the input face belongs to presenter 4. Therefore the lighting variation affects the recognition of presenters in a negative way. The mean intensities and standard deviations of a sample set of 30 key-frames from 6 presenters are shown below (table 4.1). Hence it has to be normalized to improve the recognition of the system.

Presenter number	Face number	Key-frame	Mean Intensity	Standard Diviation
1	1		81.6898	24.5247
	2		61.8152	32.1090
	3		95.6965	22.2964
	4		50.5310	21.2696
	5		93.6653	26.6544
2	6		90.2643	23.0048
	7		57.9567	23.3135
	8		108.7274	19.3238
	9		36.9202	25.0835
	10		79.8223	24.6516
3	11		126.3576	24.2091

	12		67.1813	22.6358
	13		97.6046	22.9690
	14		153.8524	31.6940
	15		98.1050	23.3332
4	16		64.7756	21.5584
	17		76.7429	22.0465
	18		109.7634	23.9939
	19		64.2910	38.6175
	20		146.1952	23.0773
5	21		75.1146	19.7646
	22		63.9561	24.8471
	23		112.0309	23.4060
	24		79.0717	23.5115
	25		54.6031	24.1956

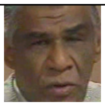
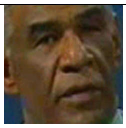
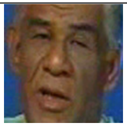
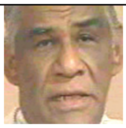
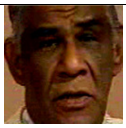
6	26		88.2534	27.138
	27		68.6036	22.6711
	28		78.5454	20.0087
	29		150.0811	33.5550
	30		55.0584	40.4808

Table 4.1: Mean intensities and standard deviations of a sample set of key-frames

4.4 Solution for Profile Overlapping by revising the Initial

Many researchers have tried to buildup a relationship between mean, median and standard deviation of image intensity values to construct a normalizing algorithm [Chennubhotla et al. 2002, Dinggang and Horace 1997, Phillips et al. 2005]. By analyzing the results and evaluations of Chennubhotla et al our research gave much attention to finding out a suitable relationship between mean intensity and standard deviation to improve recognition rate by normalizing the intensity levels of key frames[Chennubhotla et al. 2002].

After experimenting with different parameters we have explored a strategy to overcome this problem by using standard deviation and the mean intensity and we have developed an algorithm to implements this strategy as in equations (Equation 4.1 and 4.2).

```
for i = 1 to N
    for j= 1 to n
        $\text{key-frame}_{j,i}(x,y) = (\text{key-frame}_{j,i}(x,y) - \overline{X}_{i,j}) * \Gamma$   – (4. 1)
    end of for loop
end of for loop
```

Where,

key-frame$_{j,i}$ (x,y) = (x, y) pixel intensity value of the ith key-frame of the jth presenter profile

$\overline{X}_{i,j}$ = Mean intensity of jth key-frame of ith presenter

Equation 2.1 describes how our method transforms the key-frames of a presenter to the eigenspace. After experimenting with different parameters we have observed that the overlapping problem of eigenfaces can be overcome by introducing a parameter Γ. Γ is based on the adjusted values of the standard deviation and the mean of intensity values of key-frames known to the system which is computed as given in equation (4.2).

$$\Gamma = S + E_1 / S_{i,j} + \overline{X} + E_2 \qquad – (4.2)$$

Where,

$\overline{X}$ = Mean intensity of all key-frames

S = Standard deviation of all key-frames

$S_{i,j}$ = Standard deviation of jth key-frame of ith presenter

The parameters E_1 and E_2 in the equation are constants.

To obtain values for E_1 and $E_{2,}$ we carried out experiments and analyzed results on the basis of known and unknown presenters. In the following chapter we would like to present experiments that we have done to determine E_1 and E_1.

After integrating the normalizing algorithm to the system the Euclidean distance graphs for the presenter input face 1 (figure 4.6) and for the presenter input face 2 (figure 4.8) are shown in figure 4.10 and figure 4.11 respectively.

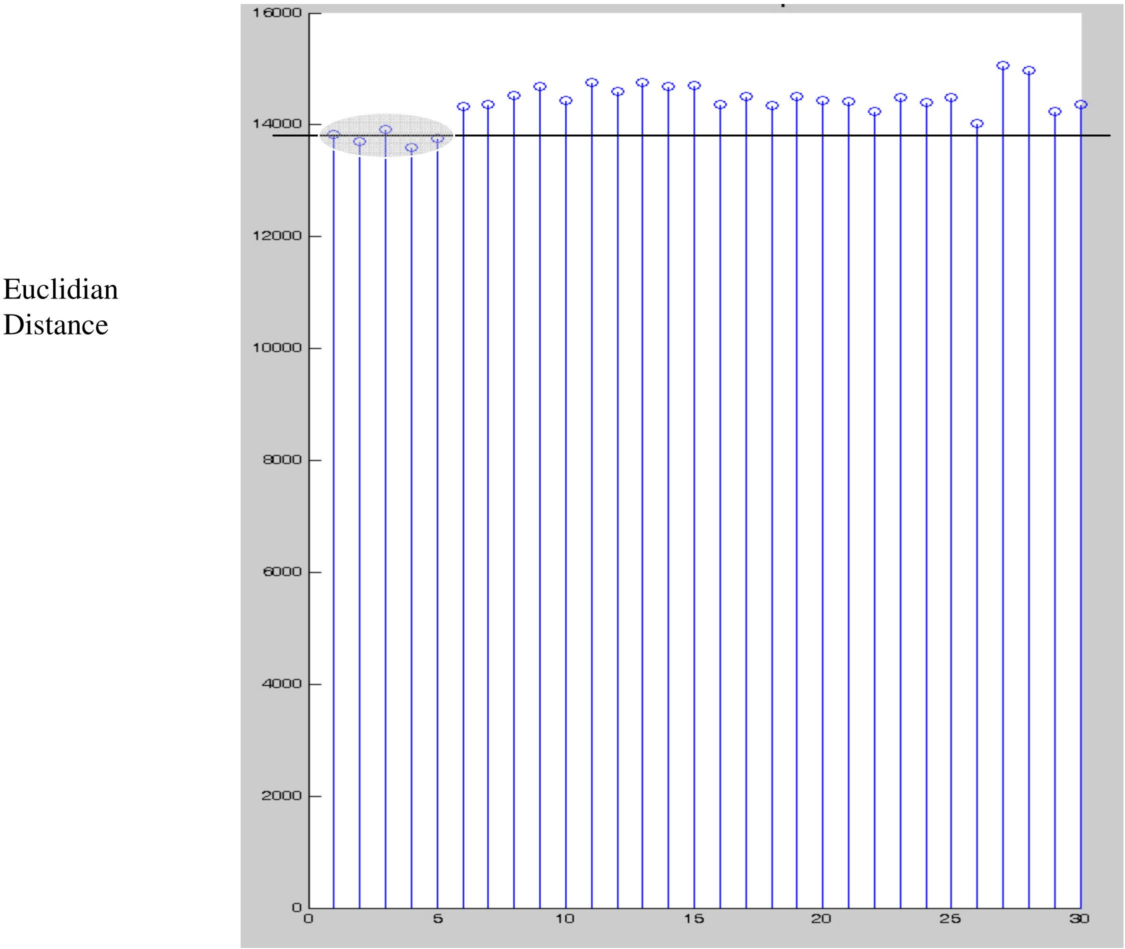


Figure 4.10: Euclidian distance calculation for face 1 after applying the normalization

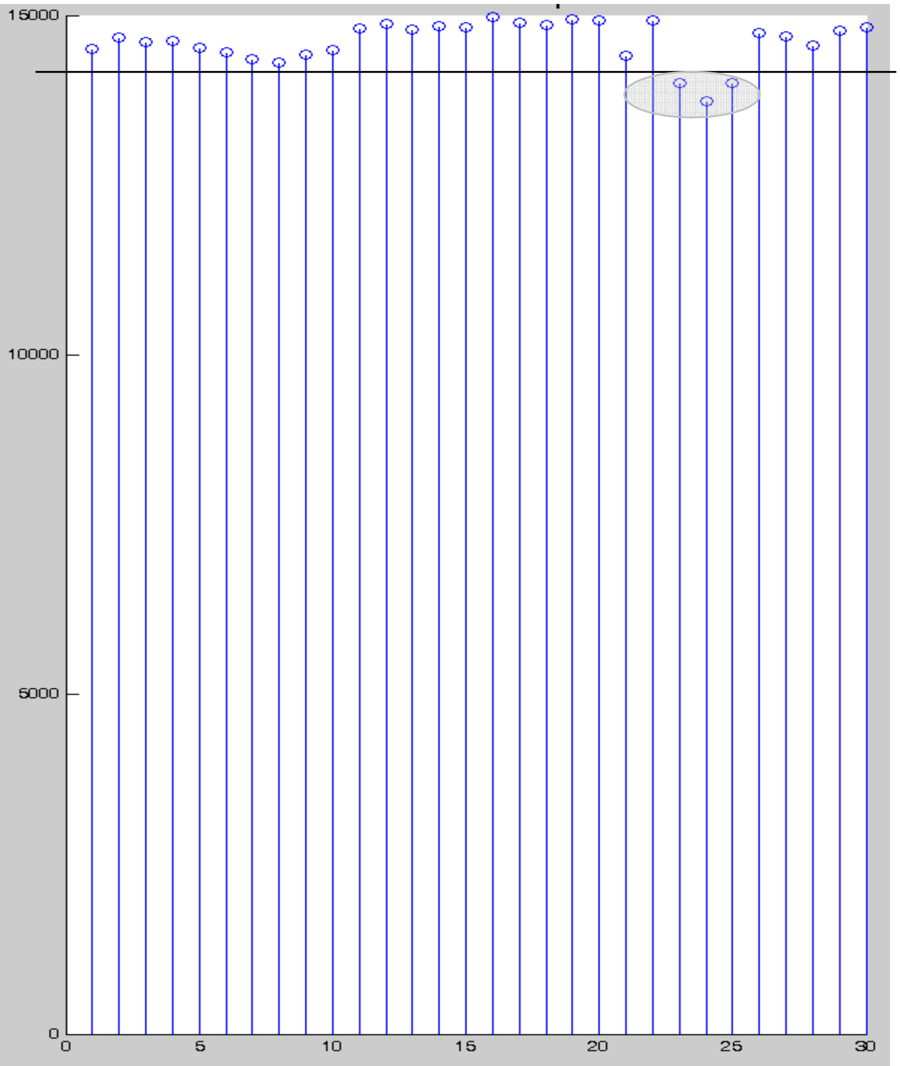
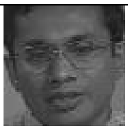
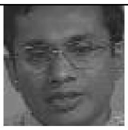
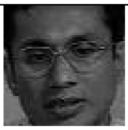
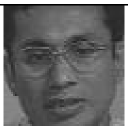
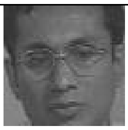
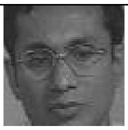
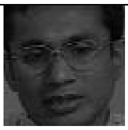
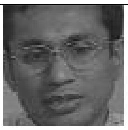
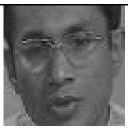
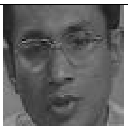
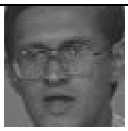
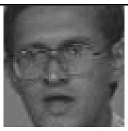
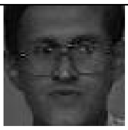
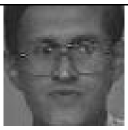
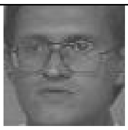
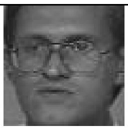
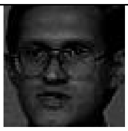
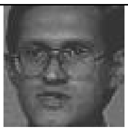


Figure 4.11: Euclidian distance calculation for face 2 after applying the normalization

From the results shown in figure 4.10 and figure 4.11 verifies the efficiency that can be provided to the system by integrating the normalizing algorithm. The table 4.2 shows a sample values for standard deviation and mean intensity. By observing that we can clearly see the key-frames gained similar values for standard deviation and mean intensity which makes the recognition process uncomplicated.

Presenter number	Face number	Key-frame Before Normalization (Gray scale)	Key-frame After Normalization (Gray scale)	Mean Intensity	Standard Diviation
1	1			89.9819	25.0176
	2			89.9990	25.0120
	3			89.9912	25.0280
	4			90.0125	24.9899
	5			90.0234	25.0012
2	6			90.0112	24.9944
	7			89.9988	25.0018
	8			89.9644	24.9879
	9			89.7888	25.0029

	10			89.9158	24.9126
3	11			89.9905	24.9730
	12			90.0059	25.0004
	13			89.9988	25.0039
	14			90.0054	25.0022
	15			90.0112	24.9984
4	16			90.0286	24.9962
	17			90.0090	24.9890
	18			89.9885	24.9632
	19			89.9885	25.0177
	20			90.0090	24.9959

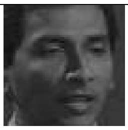
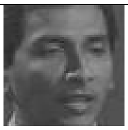
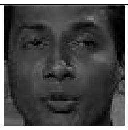
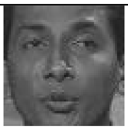
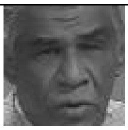
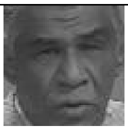
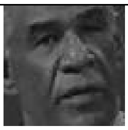
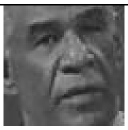
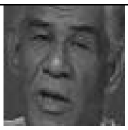
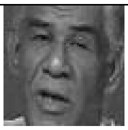
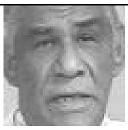
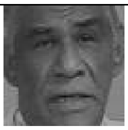
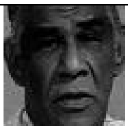
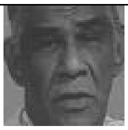
5	21			89.9739	24.9940
	22			90.0413	24.8098
	23			89.9890	24.9995
	24			90.0181	25.0278
	25			90.0022	24.9719
6	26			90.0061	25.0058
	27			90.0076	24.9833
	28			89.9829	25.0196
	29			89.9993	25.0257
	30			90.0244	24.9860

Table 4.2: Key-frames after applying the normalization algorithm

4.5 Selecting Eigenvectors

Since a face captured from a video key-frame is 128*128 pixels in dimension, the eigenvector will be a 16384 X 1 matrix. Since we use 10 faces from each presenter, therefore the dimensions of the covariance matrix would be 16384 * 10. Calculating this matrix would be very time consuming for the processor. This is one of the problems using PCA in pattern recognition since high dimensional vectors are used. Through the experience we gained from our initial approach we have realized that the efficiency of our system can be improved substantially by limiting the analysis to the dominant eigenvectors of related key-frames instead of all eigenvectors of all related key-frames.

To overcome this problem, a computationally feasible method must be used to calculate eigenfaces. All the eigenvectors which are calculated need not be used for recognition and the dimensionality reduction can be done by sorting the eigenvector according to their corresponding eigenvalues. The traditional motivation for selecting the eigenvectors with the largest eigenvalues is that the eigenvectors with the largest eigenvalues represent the amount of variance along a particular eigenvector [Turk and Pentland 1991]. By selecting the eigenvectors with the largest eigenvalues, we select the dimensions along which the presenters vary the most.

For a given square matrix of n x n dimention, it is possible to obtain n eigenvalues.
If we define e_i as the energy of the *i*th eigenvector when the eigen vectors are ordered in descending order, it is the ratio of the sum of all eigenvalues up to and including *i* over the sum of all the eigenvalues (from 1 to k).

$$e_i = \frac{\sum_{j=1}^{i} \lambda_j}{\sum_{j=1}^{k} \lambda_j} \qquad – (4.3)$$

Kirby defines e_i as the energy dimension [Broomhead and Kirby 2000]. The variation depends upon the stretching dimension, also defined by Kirby [Broomhead and Kirby 2000]. The stretch s_i for the ith eigenvector is the ratio of that eigenvalue over the largest eigenvalue (λ_1):

$$s_i = \frac{\lambda_i}{\lambda_1} \qquad – (4.4)$$

Experiments were carried by selecting 120 key frames of 12 distinct presenters such that 10 frames from each presenter. The calculated 120 eigenvector are shown below in descending order.

(1.8787+ 1.494+ 0.952+ 0.778+ 0.446+ 0.367+ 0.307+ 0.287+ 0.223+ 0.215+ 0.200+ 0.179+ 0.167+ 0.155+ 0.133+ 0.121+ 0.116+ 0.098+ 0.095+ 0.087+ 0.076+ 0.074+ 0.074+ 0.070+ 0.066+ 0.061+ 0.060+ 0.057+ 0.054+ 0.051+ 0.049+ 0.048+ 0.047+ 0.046+ 0.045+ 0.043+ 0.042+ 0.041+ 0.039+ 0.038+ 0.036+ 0.035+ 0.035+ 0.034+ 0.033+ 0.032+ 0.031+ 0.030+ 0.030+ 0.029+ 0.029+ 0.028+ 0.027+ 0.026+ 0.026+ 0.025+ 0.025+ 0.024+ 0.023+ 0.023+ 0.022+ 0.022+ 0.022+ 0.022+ 0.021+ 0.021+ 0.020+ 0.019+ 0.019+ 0.019+ 0.018+ 0.018+ 0.018+ 0.017+ 0.016+ 0.016+ 0.016+ 0.016+ 0.016+ 0.015+ 0.015+ 0.015+ 0.014+ 0.014+ 0.014+ 0.013+ 0.013+ 0.013+ 0.012+ 0.012+ 0.012+ 0.012+ 0.012+ 0.011+ 0.011+ 0.011+ 0.011+ 0.011+ 0.010+ 0.010+ 0.009+ 0.009+ 0.009+ 0.009+ 0.009+ 0.008+ 0.008+ 0.008+ 0.007+ 0.007+ 0.005+ 0.005+ 0.003+ 0.003+ 0.0006+ 0.0003+ 0.00008+ 0.00006+ 0.00004+ 0.000001) X 10^9

For this sample,

k=120 and,

$$\sum_{j=1}^{k} \lambda_j = 10.679781 \text{ X } 10^9$$

Using the equations (4.3) and (4.4), *s* and *e* are calculated for the sample set of 120 eigenvectors. A set of 60 key frames which includes known presenters in the database were selected to test the recognition performance and for each selected eigenvector set, recognition rate is calculated (See Table 4.2).

Number of Eigenvectors	s	e	Correctly Classified frames	Recognition rate
5	0.2374	51.96%	11	21.67%
10	0.1144	65.05%	19	31.67%
15	0.07079	72.86%	23	38.33%
20	0.04631	77.70%	28	46.67%
25	0.03513	81.08%	34	56.67%
30	0.02715	83.73%	37	61.67%
35	0.02395	85.93%	40	66.67%
40	0.02023	87.83%	43	71.67%
45	0.01757	89.45%	45	75.00%
50	0.01544	90.72%	47	78.33%
55	0.01384	92.03%	49	81.67%
60	0.01224	93.18%	51	85.00%
65	0.01118	94.23%	53	88.33%
70	0.01011	95.18%	55	91.67%
75	0.00852	96.02%	55	91.67%
80	0.00798	96.78%	55	91.67%
85	0.00745	97.47%	55	91.67%
90	0.00639	97.97%	55	91.67%
95	0.00586	98.53%	55	91.67%
100	0.00532	99.04%	54	90.00%
105	0.00479	99.47%	54	90.00%
110	0.00373	99.84%	53	88.33%
115	0.00032	99.995%	52	86.67%
120	0.00000	100.00%	53	88.33%

Table 4.3 : The Energy and Stretching dimensions.

Since the eigenvectors are ordered in high to low by the amount of variance found between key-frames along each eigenvector, the last eigenvectors are the smallest amounts of variance. The assumption can be made that noise is associated with the lower valued Eigenvalues where smaller amounts of variation are found among the key-frames [Broomhead and Kirby 2000]. The results given in the table 4.2 are plotted in figure 4.12 to visualize the effect of eliminating these Eigenvectors from the Eigenspace to improve the performance.

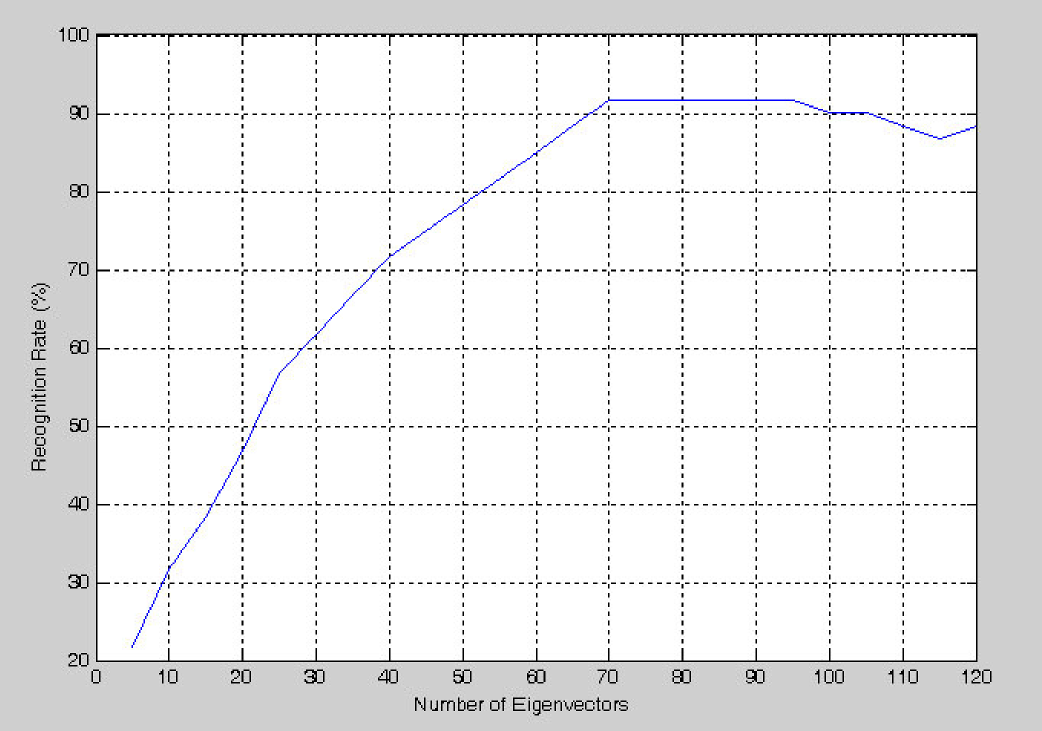


Figure 4.12: Performance when ordered by eigenvectors versus recognition rate.

By analyzing the graph on figure 4.12 it is evident that all Eigenvectors with s_i greater than a particular threshold decreasing the recognition rate. For the eigenvector selection process, threshold value (λ_t) is determined to select the eigenvectors which is most suitable to construct a presenter profile (equation 4.5). Algorithm has been developed to calculate λ_t by analyzing the behavior of each profile projection to the face space using different number of eigenvectors (table 4.3).

$$\lambda_t = (\lambda_{max}{}^{1/2}) * 3n \qquad - (4.5)$$

λ_{max} = Maximum Eigenvalue

λ_t = Threshold Value for Eigenvector Selection

n = Total Number of Eigenvectors in the Presenter Database

Using the above equation we eliminate the eigenvectors less than V_t when constructing a presenter profile. For the data set on table 4.2,

$\lambda_t = 1.5603830 \text{ x } 10^7$

Using our algorithm, the first 80 eigenvectors are selected and others are omitted. From the table 4.1 we observe that when s is in between the limits of 0.01 and 0.006, the system acquires the highest recognition rate. Therefore the eigenvectors should be chosen between 70 and 90 for the maximum performance.

This verifies that our algorithm helps to reduce the processing speed of the system and the efficiency of the recognition process. Figure 4.13 shows schematically what the algorithm does. It takes the training faces as input and yields the eigenfaces as output. Once this is done, the recognition process can begin.

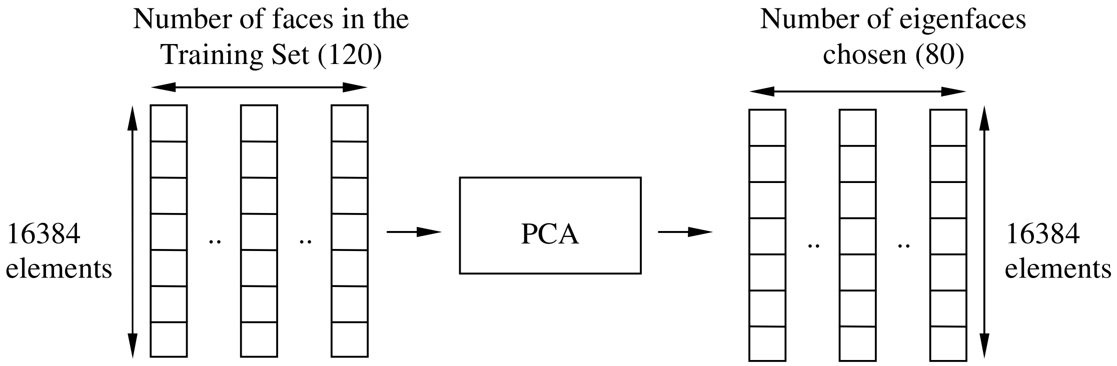


Figure 4.13: Selection of eigenvectors

Chapter 5

5 Experiment Results

In this chapter we describe how to determine error values E_1 and E_2 in the equation,

$$\Gamma = S + E_1 / S_{i,j} + \bar{X} + E_2$$

which was explained in detail in the chapter 4. The appropriate values of E_1 and E_2 for the key-frame normalization process are selected by considering how these values are affected on the standard deviation and mean intensity values of the key-frames in the database as given below.

Where, $\bar{X}_c = \bar{X}_a + E_2$ and $S_c = S_a + E_1$

Experiments are based on two different data sets.

- The Dataset obtained from the Bachelor of Information Technology (BIT) external degree program TV programs conducted by the University of Colombo School of Computing (UCSC). A sample of the dataset is shown in figure 5.1.
- The Dataset obtained from the ORL face database, which can be used freely for academic research. ORL face database contains 40 distinct persons, each person having ten different face images. A description of the ORL face database and a web-link to download the database can be found at http://www.uk.research.att.com/facedatabase.html.

The algorithm used to determine the values of E_1 and E_2 is given in the figure 5.2. From The test set 60 key-frames consists of known (presenters which are in the profile database) and unknown (presenters which are not the profile database) presenters and the distance between each key-frame from the test set and the profiles in the database are calculated according to the Euclidian distance measurement (equation 2.10) described in chapter 2. The purpose of this experiment is to discover the most suitable range of values for the threshold to verify presenters.

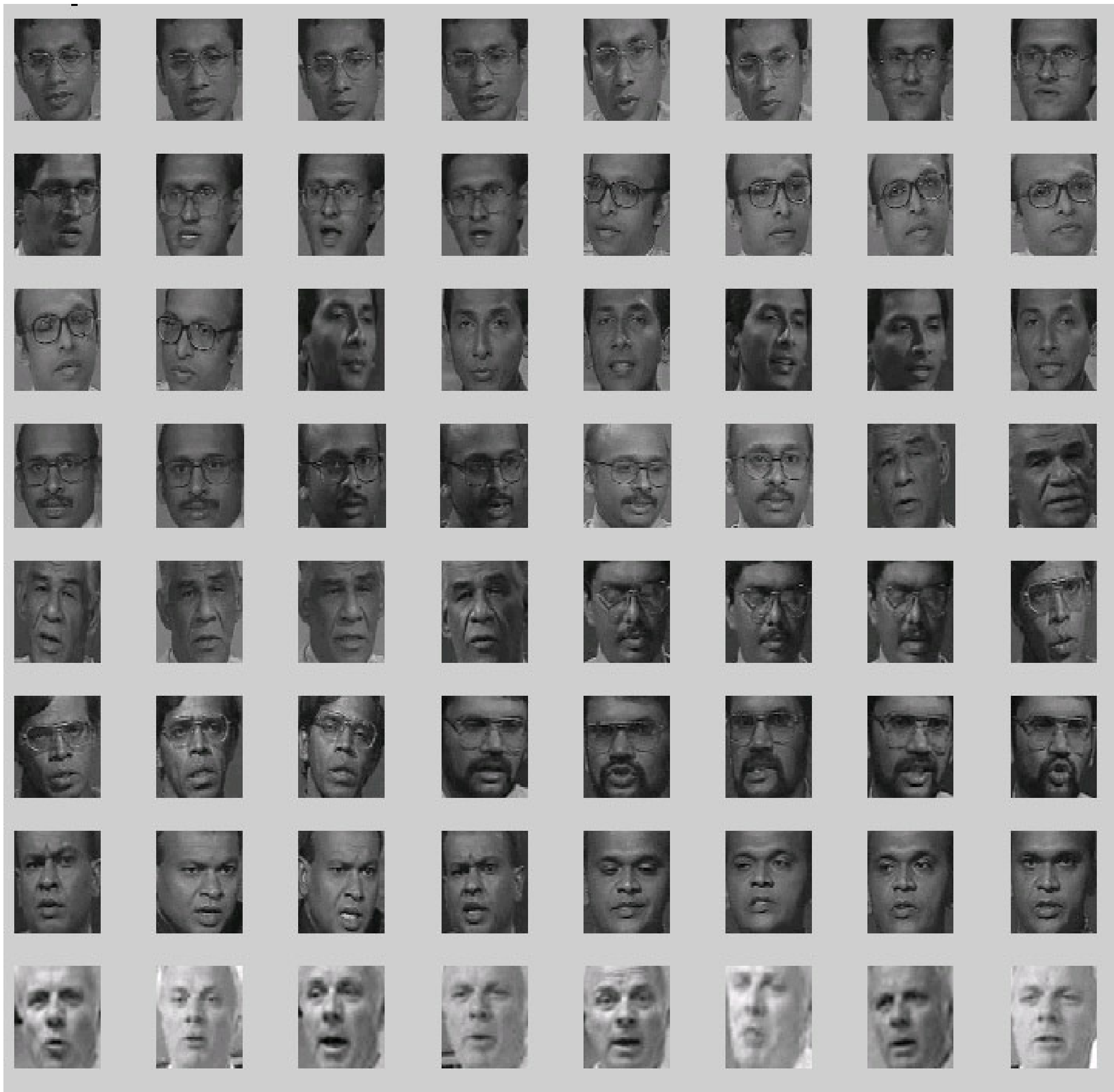


Figure 5.1: Sample dataset obtained from the Bachelor of Information Technology external degree program.

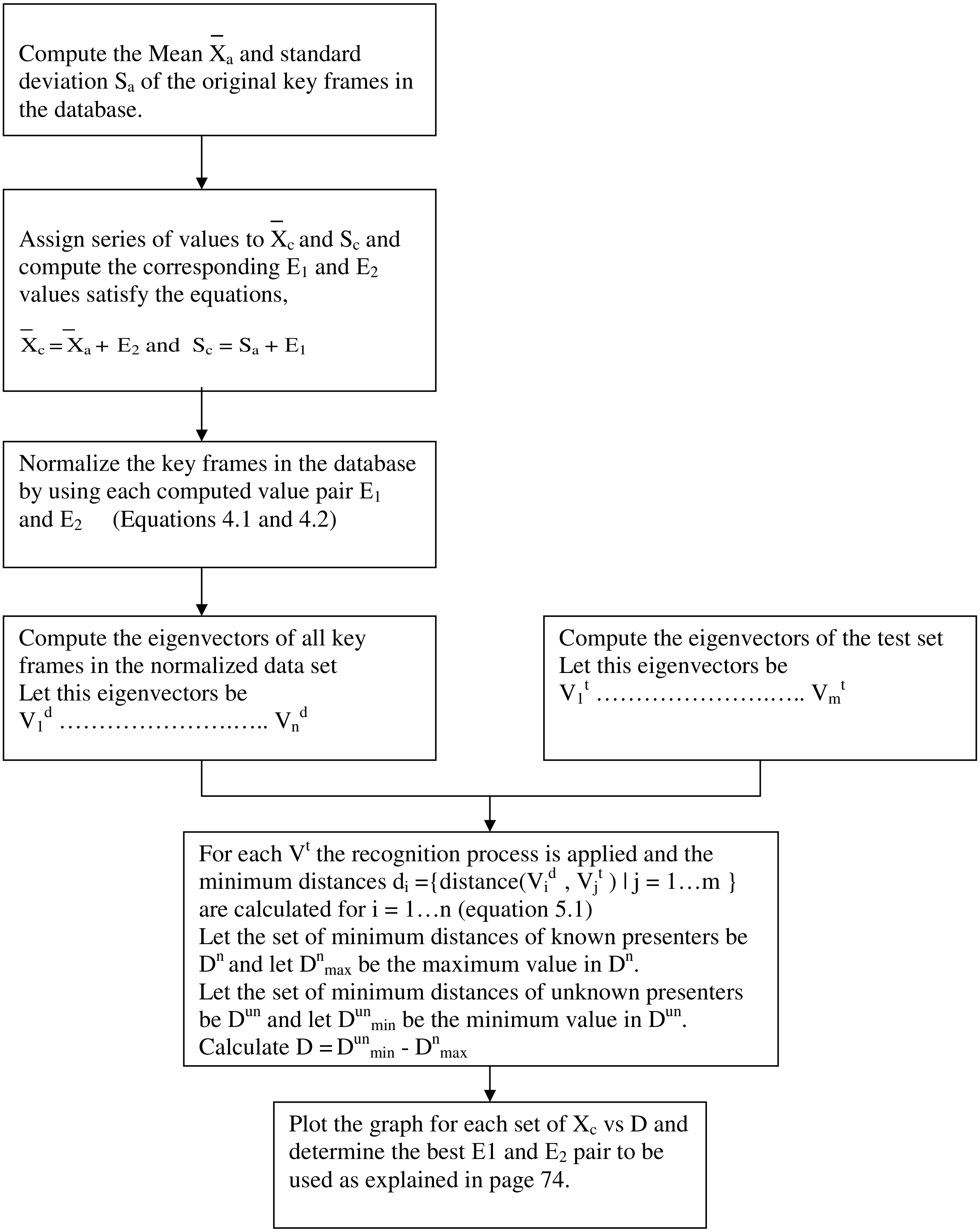

Compute the Mean $\overline{X}_a$ and standard deviation S_a of the original key frames in the database.
Assign series of values to $\overline{X}_c$ and S_c and compute the corresponding E_1 and E_2 values satisfy the equations, $\overline{X}_c = \overline{X}_a + E_2$ and $S_c = S_a + E_1$
Normalize the key frames in the database by using each computed value pair E_1 and E_2 (Equations 4.1 and 4.2)
Compute the eigenvectors of all key frames in the normalized data set Let this eigenvectors be V_1^d …………………….. V_n^d
Compute the eigenvectors of the test set Let this eigenvectors be V_1^t …………………….. V_m^t
For each V^t the recognition process is applied and the minimum distances $d_i = \{distance(V_i^d , V_j^t) \mid j = 1 \ldots m \}$ are calculated for $i = 1 \ldots n$ (equation 5.1) Let the set of minimum distances of known presenters be D^n and let D^n_{max} be the maximum value in D^n. Let the set of minimum distances of unknown presenters be D^{un} and let D^{un}_{min} be the minimum value in D^{un}. Calculate $D = D^{un}_{min} - D^n_{max}$
Plot the graph for each set of X_c vs D and determine the best E1 and E_2 pair to be used as explained in page 74.

Figure 5.2: The process to determine E_1 and E_2

5.1 Computation of E_1 and E_2 on the test data set

Each cell in the table 5.1 give the values which are D^{un}_{min} ,D^{n}_{max} **,** D where, D = (D^{un}_{min}) - (D^{n}_{max}) in that order. After conducting several experiments on different combinations of $\bar{X}_c$ and S_c and analyzing them we saw that after certain values, we cannot expect good result for E_1 and E_2 (See Figure 5.4). Hence the table 5.1 shows only the $\bar{X}_c$ and S_c which are useful for the experiments. The sample scenario is shown in figure 5.3. The threshold for the selection process must be chosen so that the known and unknown set of faces should have a clear difference in the distance when equation 2.10 is applied to the facespace.

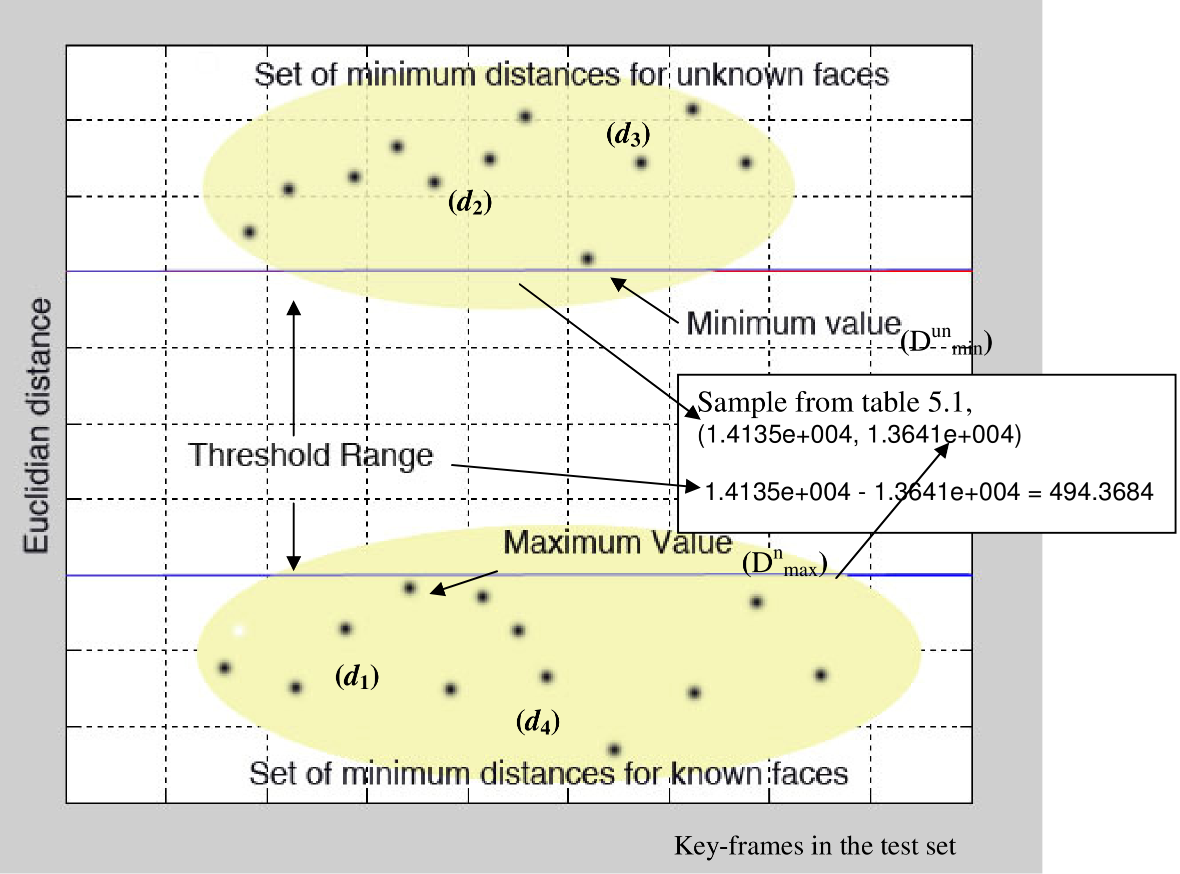

Set of minimum distances for unknown faces
(d_3)
(d_2)
Minimum value (D^{un}_{min})
Euclidian distance
Sample from table 5.1,
(1.4135e+004, 1.3641e+004)
Threshold Range
1.4135e+004 - 1.3641e+004 = 494.3684
Maximum Value (D^{n}_{max})
(d_1)
(d_4)
Set of minimum distances for known faces
Key-frames in the test set

Figure 5.3: Defining the threshold range

For the set of known faces there will be a maximum value from the set of minimum distance values and at the same time there will be a minimum value from the set of minimum distance values for unknown presenters (see figure 5.2). The values in the table 5.1 are calculated according to standard below.

In table 5.1,

$$(x,y) = \left(\begin{array}{l}\text{Minimum value from the set of}\\ \text{minimum distance values for}\\ \text{unknown presenters } (D^{un}_{min})\end{array}\right), \left(\begin{array}{l}\text{Maximum value from the set of}\\ \text{minimum distance values for}\\ \text{known presenters } (D^{n}_{max})\end{array}\right)$$

$\bar{X}_c$ S_c	40	60	80	100	120	140	160	180	200
20 (y1)	(7.1994e+003, 6.8728e+003) -26.6297	(1.0244e+004, 9.5550e+003) -88.6685	(1.3072e+004, 1.2193e+004) -78.9165	(1.5761e+004, 1.4653e+004) 88.0	(1.8295e+004, 1.7026e+004) 69.2	(2.0548e+004, 1.9367e+004) 181.7	(2.2600e+004, 2.1623e+004) 77.0411	(2.4416e+004, 2.3691e+004) 24.5688	(2.5984e+004, 2.5550e+004) 33.5920
30 (y2)	(8.8961e+003, 8.9011e+003) -4.9924	(1.1562e+004, 1.1273e+004) 289.2325	(1.4135e+004, 1.3641e+004) 494.3684	(1.6642e+004, 1.5942e+004) 699.4769	(1.8949e+004, 1.8163e+004) 785.9293	(2.1002e+004, 2.0356e+004) 646.2675	(2.2832e+004, 2.2388e+004) 444.6899	(2.4389e+004, 2.4209e+004) 180.1357	(2.5717e+004, 2.5678e+004) 39.4105
40 (y3)	(1.1281e+004, 1.1391e+004) -110.5575	(1.3389e+004, 1.3335e+004) 54.1764	(1.5705e+004, 1.5408e+004) 297.4701	(1.7866e+004, 1.7489e+004) 376.6582	(1.9833e+004, 1.9521e+004) 311.4084	(2.1583e+004, 2.1469e+004) 113.2760	(2.3083e+004, 2.3213e+004) -129.5541	(2.4354e+004, 2.4601e+004) -247.7341	(2.5420e+004, 2.5714e+004) -293.7693
60 (y4)	(1.4084e+004, 1.4197e+004) -113.0043	(1.5675e+004, 1.5722e+004) -46.9350	(1.7700e+004, 1.7462e+004) 238.6199	(1.9425e+004, 1.9255e+004) 170.3958	(2.0967e+004, 2.1051e+004) -84.3671	(2.2284e+004, 2.2667e+004) -383.3876	(2.3376e+004, 2.3935e+004) -558.0927	(2.4283e+004, 2.4879e+004) -596.4928	(2.5020e+004, 2.5739e+004) -719.1017
80 (y5)	(1.7126e+004, 1.7204e+004) -77.3148	(1.8276e+004, 1.8333e+004) -57.4355	(1.9800e+004, 1.9732e+004) 68.1864	(2.1226e+004, 2.1258e+004) -31.6178	(2.2253e+004, 2.2646e+004) -392.7196	(2.3019e+004, 2.3765e+004) -746.4282	(2.3594e+004, 2.4490e+004) -895.8738	(2.4092e+004, 2.5125e+004) -1032.8	(2.4551e+004, 2.5759e+004) -1208.7
100 (y6)	(2.0256e+004, 2.0300e+004) -44.1794	(2.1074e+004, 2.1126e+004) -51.3560	(2.2036e+004, 2.2180e+004) -143.6539	(2.2762e+004, 2.3293e+004) -531.2803	(2.3202e+004, 2.4161e+004) -958.9567	(2.3479e+004, 2.4613e+004) -1133.7	(2.3694e+004, 2.4948e+004) -1254.4	(2.3957e+004, 2.5324e+004) -1366.5	(2.4243e+004, 2.5748e+004) -1504.7
120 (y7)	(2.3419e+004, 2.3433e+004) -13.9557	(2.3959e+004, 2.4028e+004) -69.4990	(2.4248e+004, 2.4688e+004) -440.6733	(2.4252e+004, 2.5221e+004) -969.5989	(2.4121e+004, 2.5336e+004) -1215.5	(2.3967e+004, 2.5320e+004) -1352.9	(2.3881e+004, 2.5333e+004) -1452.3	(2.3890e+004, 2.5403e+004) -1513.6	(2.4003e+004, 2.5591e+004) -1587.8
140 (y8)	(2.6585e+004, 2.6603e+004) -17.9678	(2.6747e+004, 2.6967e+004) -220.6420	(2.6288e+004, 2.6959e+004) -671.2647	(2.5626e+004, 2.6682e+004) -1055.4	(2.5019e+004, 2.6278e+004) -1259.7	(2.4484e+004, 2.5882e+004) -1398.8	(2.4100e+004, 2.5530e+004) -1430.3	(2.3890e+004, 2.5337e+004) -1446.9	(2.3819e+004, 2.5331e+004) -1511.1
180 (y9)	(2.9667e+004, 2.9727e+004) -59.1125	(2.9221e+004, 2.9608e+004) -386.3919	(2.8074e+004, 2.8882e+004) -807.5172	(2.6859e+004, 2.7975e+004) -1116.0	(2.5825e+004, 2.7098e+004) -1273.2	(2.4990e+004, 2.6263e+004) -1272.5	(2.4318e+004, 2.5544e+004) -1226.8	(2.3887e+004, 2.5138e+004) -1251.1	(2.3661e+004, 2.4986e+004) -1324.4
200 (y10)	(3.2486e+004, 3.2671e+004) -184.3777	(3.1242e+004, 3.1664e+004) -421.3555	(2.9568e+004, 3.0237e+004) -668.8497	(2.7897e+004, 2.8805e+004) -908.9278	(2.6489e+004, 2.7569e+004) -1080.4	(2.5370e+004, 2.6399e+004) -1029.2	(2.4496e+004, 2.5423e+004) -927.5122	(2.3859e+004, 2.4824e+004) -964.7370	(2.3493e+004, 2.4538e+004) -1045.1

Table 5.1: Results obtained using the Bachelor of Information Technology external degree program database

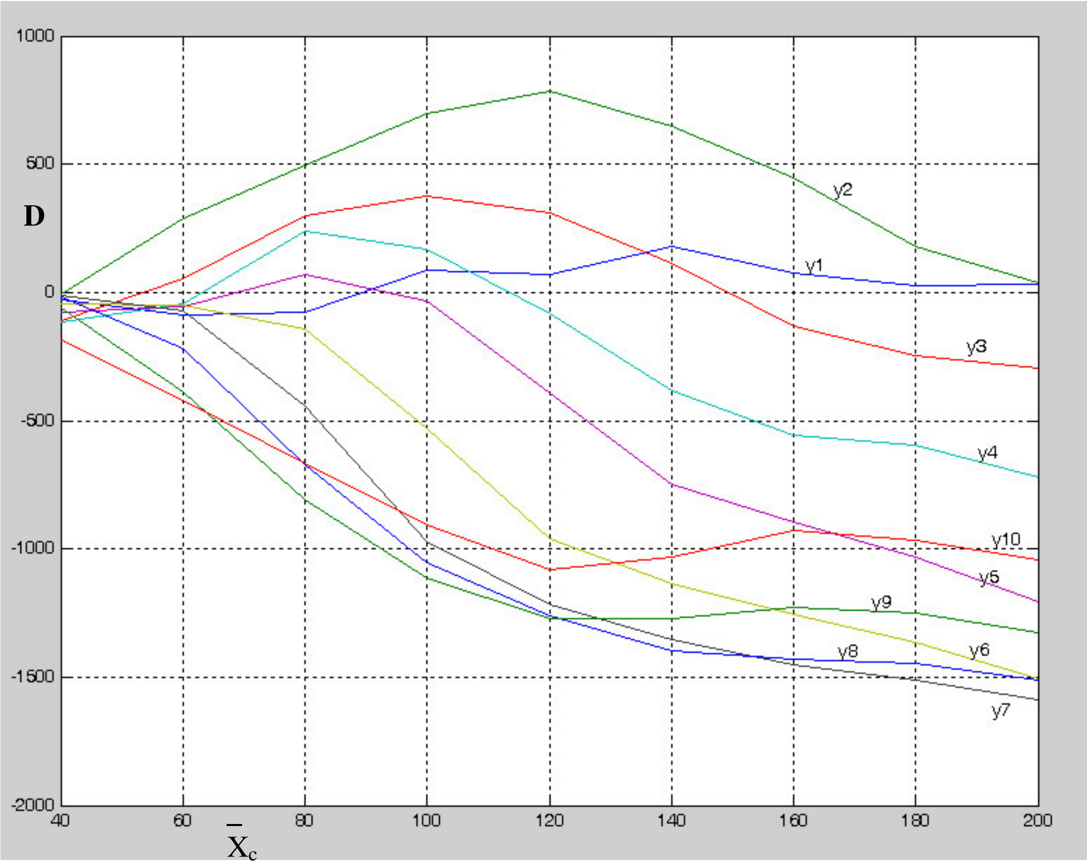


Figure 5.4: Results on D and $\bar{X}_c$ variations in the BIT database

X-axis = $\bar{X}_c$

y–axis = D

The graph is generated using the D versus the $\bar{X}_c$ to analyze the variation of the possible threshold values that can be obtained from the test set (figure 5.4). The idea behind these experiments is to get the maximum value for D which determines the range that we can set a threshold value for recognition. Comparing the results obtained from figure 5.4 we can see that y_2 graph gives the maximum range to set a threshold value. In figure 5.5 we can analyze clearly the best result can get when,

$$\overline{X}_c = 120 \qquad \text{and} \quad S_c = 30 \quad - (5.1)$$

The actual values for S and $\overline{X}$ are 23.9111 and 81.09 respectively.

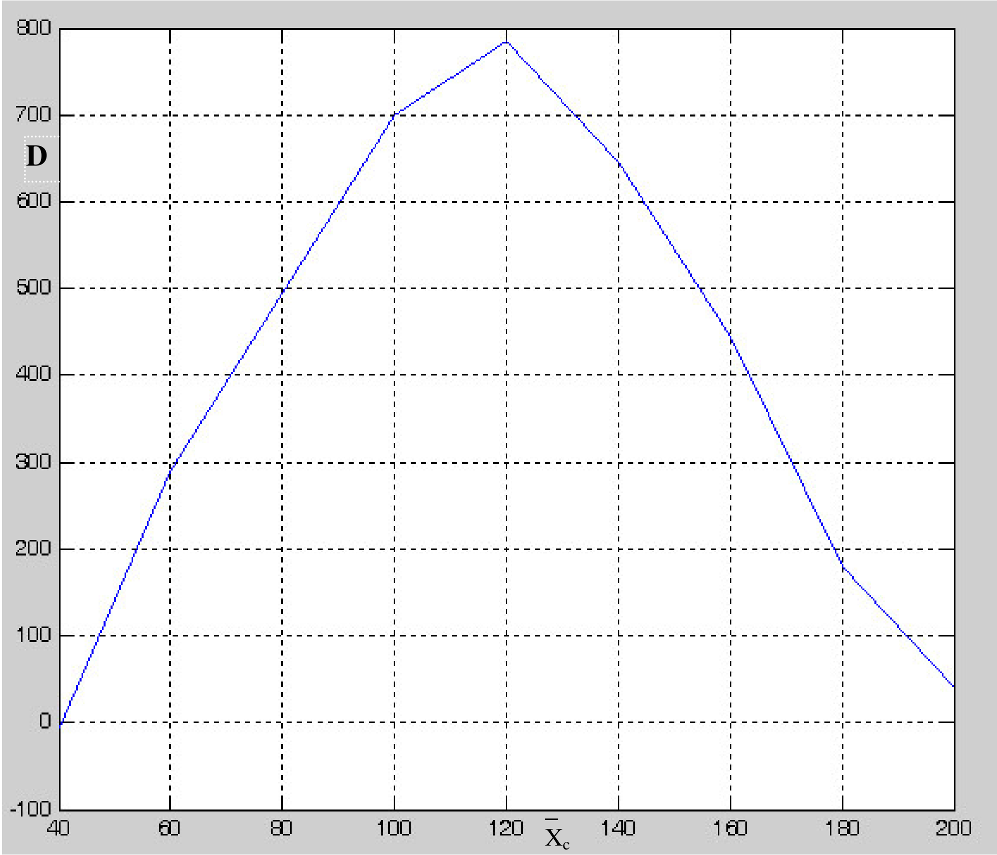


Figure 5.5: Best combination of D and $\overline{X}_c$

To verify the values for the E_1 and E_2 we experimented with ORL face database using 20 different people (Figure 5.6). This database is composed of ten different images of each of 40 distinct people. The images were taken at different times, varying the lighting, facial expressions (open / closed eyes, smiling / not smiling) and facial details (glasses / no glasses).

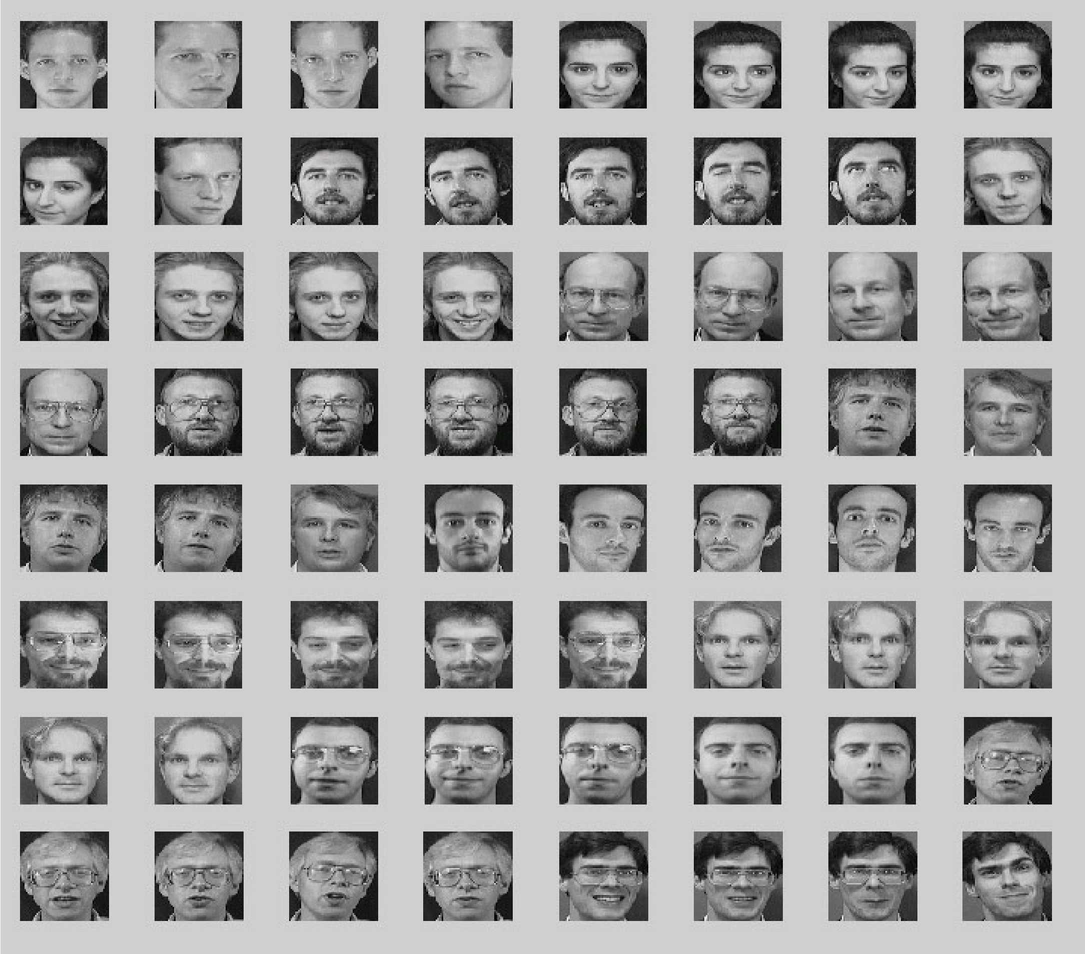


Figure 5.6: Dataset obtained from ORL face database

To facilitate the experiment process, selected faces from the database were selected manually. Several experiments were conducted in which, the training set consists of 20 people and the results are stored in Table 5.2. Figure 5.7 illustrates the performance of the normalization algorithm when tested by different sets of faces from the ORL face database.

$\bar{\mathbf{X}}_c$ $\mathbf{S}_c$	40	60	80	100	120	140	160	180
20(y1)	(6.9307e+003, 6.8568e+003) 73.9134	(8.6334e+003, 8.5525e+003) 80.9548	(1.0801e+004, 1.1142e+004) -341.3618	(1.3139e+004, 1.3991e+004) -852.5535	(1.5517e+004, 1.6928e+004) -1411	(1.7889e+004, 1.9872e+004) -1982.7	(1.9674e+004, 2.2470e+004) -2795.7	(2.0687e+004, 2.4652e+004) -3965.7
30(y2)	(1.0034e+004, 9.9822e+003) 51.7850	(1.1142e+004, 1.0994e+004) 147.2640	(1.3011e+004, 1.2892e+004) 119.0673	(1.5151e+004, 1.5408e+004) -256.8519	(1.7403e+004, 1.8153e+004) -749.6593	(1.9458e+004, 2.0800e+004) -1342.1	(2.0470e+004, 2.2949e+004) -2479.4	(2.1168e+004, 2.4783e+004) -3614.6
40(y3)	(1.3226e+004, 1.3186e+004) 39.4689	(1.4011e+004, 1.3866e+004) 145.0577	(1.5473e+004, 1.5263e+004) 210.5281	(1.7401e+004, 1.7235e+004) 166.2503	(1.9450e+004, 1.9633e+004) -182.4127	(2.0716e+004, 2.1714e+004) -998.4591	(2.1318e+004, 2.3401e+004) -2082.7	(2.1760e+004, 2.4863e+004) -3102.9
50(y4)	(1.6452e+004, 1.6420e+004) 31.9043	(1.7082e+004, 1.6963e+004) 119.2943	(1.8158e+004, 1.7932e+004) 226.1942	(1.9829e+004, 1.9563e+004) 265.9257	(2.1345e+004, 2.1373e+004) -27.6321	(2.1917e+004, 2.2661e+004) -744.0122	(2.2194e+004, 2.3993e+004) -1799.0	(2.2404e+004, 2.5026e+004) -2621.7
60(y5)	(1.9696e+004, 1.9670e+004) 26.8065	(2.0224e+004, 2.0123e+004) 101.3624	(2.1091e+004, 2.0874e+004) 217.3302	(2.2313e+004, 2.2067e+004) 245.4659	(2.2999e+004, 2.3296e+004) -297.3063	(2.3085e+004, 2.4412e+004) -1327.0	(2.3055e+004, 2.5237e+004) -2181.3	(2.3020e+004, 2.5798e+004) -2777.6
70(y6)	(2.2951e+004, 2.2928e+004) 23.1449	(2.3405e+004, 2.3318e+004) 87.5156	(2.4142e+004, 2.3953e+004) 189.2552	(2.4579e+004, 2.4661e+004) -82.5845	(2.4517e+004, 2.5416e+004) -899.3127	(2.4202e+004, 2.5914e+004) -1712	(2.3851e+004, 2.6140e+004) -2288.4	(2.3546e+004, 2.6254e+004) -2707.4
80(y7)	(2.6212e+004, 2.6191e+004) 20.3901	(2.6610e+004, 2.6533e+004) 76.9214	(2.7075e+004, 2.7005e+004) 70.1105	(2.6663e+004, 2.7235e+004) -571.8466	(2.5905e+004, 2.7246e+004) -1340.8	(2.5182e+004, 2.7009e+004) -1827.1	(2.4518e+004, 2.6701e+004) -2182.8	(2.3966e+004, 2.6414e+004) -2447.9
100(y8)	(2.9477e+004, 2.9458e+004) 18.2436	(2.9825e+004, 2.9757e+004) 67.8769	(2.9509e+004, 2.9868e+004) -359.1475	(2.8443e+004, 2.9441e+004) -997.8062	(2.7071e+004, 2.8565e+004) -1494.4	(2.5830e+004, 2.7675e+004) -1845.1	(2.4842e+004, 2.6904e+004) -2061.3	(2.4051e+004, 2.6281e+004) -2230.6
120(y9)	(3.2745e+004, 3.2728e+004) 16.5245	(3.2687e+004, 3.2845e+004) -158.0199	(3.1482e+004, 3.2005e+004) -523.3275	(2.9840e+004, 3.0692e+004) -851.9657	(2.8004e+004, 2.9354e+004) -1349.5	(2.6311e+004, 2.7917e+004) -1605.6	(2.4992e+004, 2.6761e+004) -1768.5	(2.3983e+004, 2.5883e+004) -1900.0
140(y10)	(3.5999e+004, 3.5992e+004) 6.8887	(3.4838e+004, 3.5159e+004) -320.8809	(3.2936e+004, 3.3341e+004) -404.7724	(3.0785e+004, 3.1552e+004) -766.9533	(2.8607e+004, 2.9607e+004) -1000.1	(2.6541e+004, 2.7782e+004) -1241.6	(2.4857e+004, 2.6334e+004) -1476.8	(2.3567e+004, 2.5253e+004) -1686.1
160(y11)	(3.8407e+004, 3.8509e+004) -101.5582	(3.6136e+004, 3.6360e+004) -224.3809	(3.3620e+004, 3.3992e+004) -371.8264	(3.1117e+004, 3.1708e+004) -590.8153	(2.8688e+004, 2.9414e+004) -725.9818	(2.6410e+004, 2.7364e+004) -953.7415	(2.4383e+004, 2.5699e+004) -1315.6	(2.2604e+004, 2.4458e+004) -1853.1
200(y12)	(3.8764e+004, 3.8851e+004) -86.7488	(3.6008e+004, 3.6196e+004) -188.1947	(3.3297e+004, 3.3543e+004) -246.0024	(3.0645e+004, 3.1103e+004) -457.5385	(2.8099e+004, 2.8862e+004) -762.8118	(2.5693e+004, 2.6698e+004) -1004.3	(2.3482e+004, 2.4886e+004) -1403.8	(2.1558e+004, 2.3530e+004) -1971.8

Table 5.2: Results obtained from ORL face database

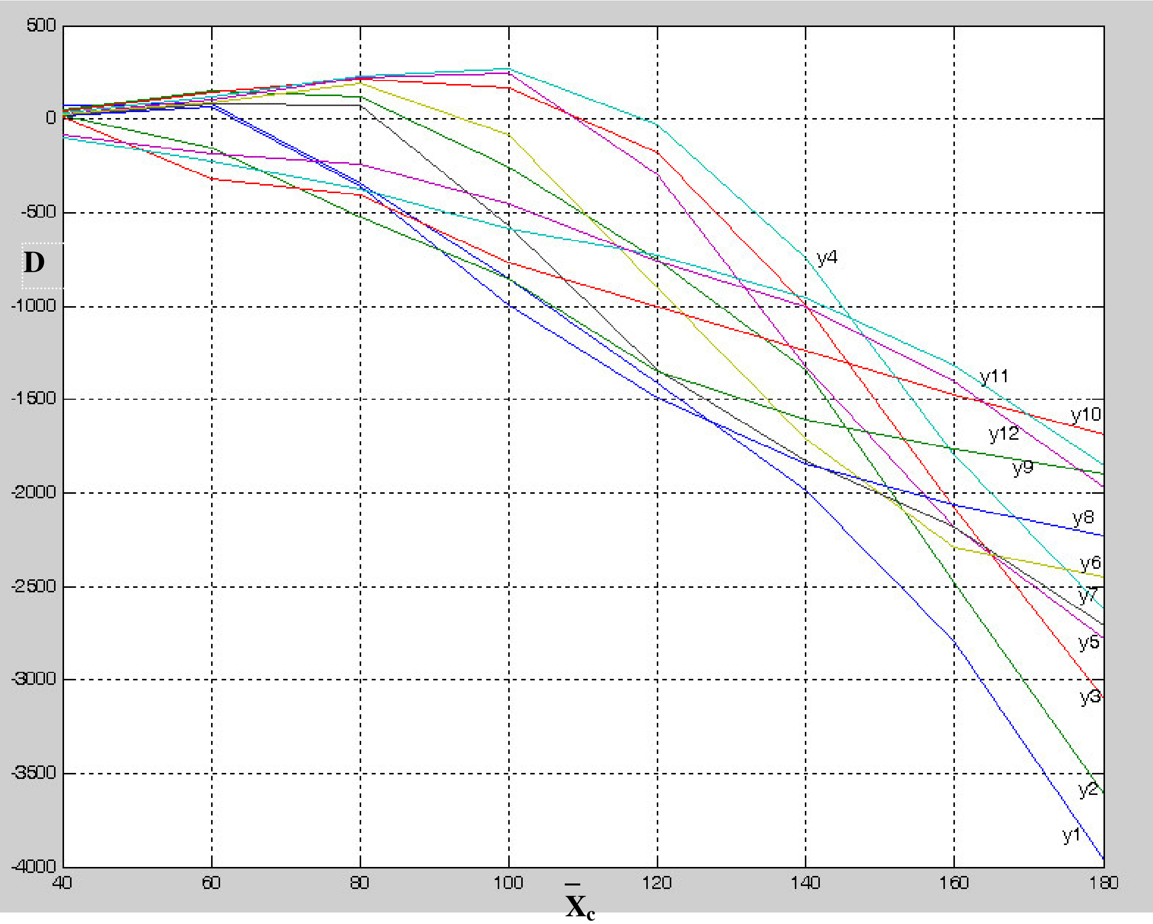


Figure 5.7: Results on D and $\bar{X}_c$ variations in the ORL face database

X-axis = $\bar{X}_c$

Y –axis = D

In this dataset, $\bar{X}$ = 60.36 and S = 40.9703

By analyzing results obtained from figure 5.7 we can see that y_4 graph gives the maximum range to set a threshold value. From figure 5.8 we can investigate clearly that the best result can be obtained when,

$\bar{X}_c$ =100 and S_c = 50 - (5.2)

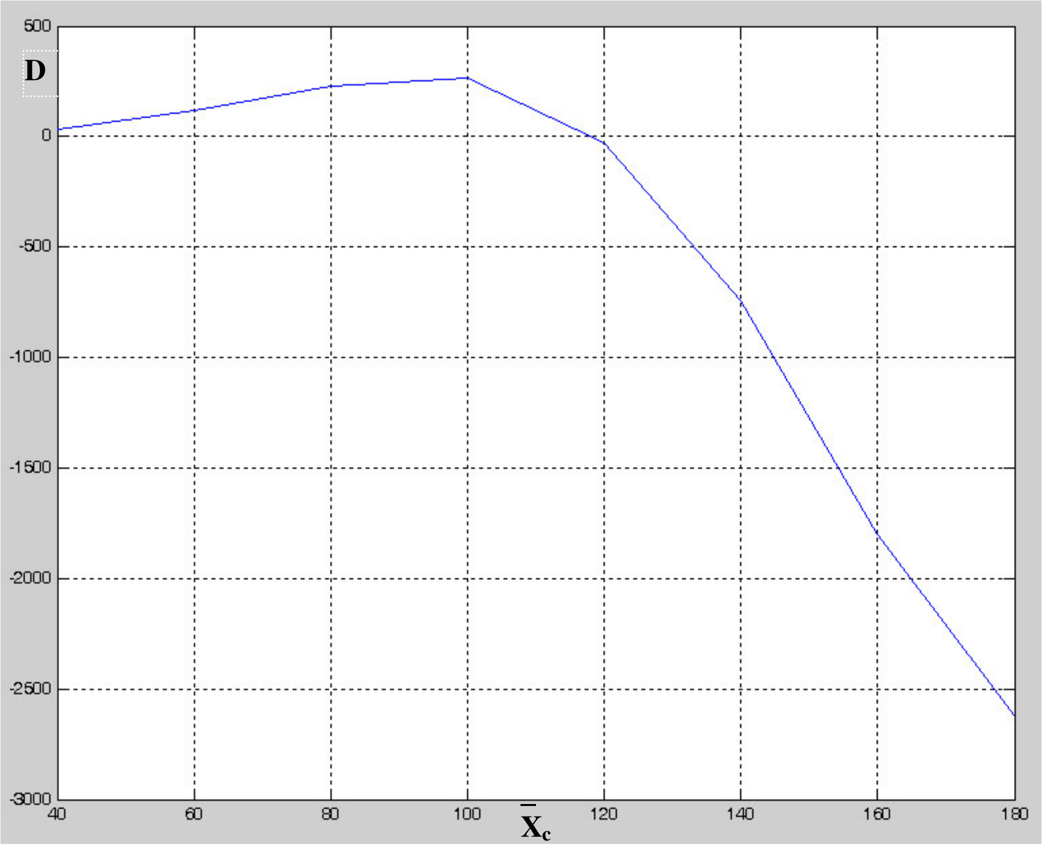


Figure 5.8: Best combination of D and $\bar{X}_c$ from ORL face database

By evaluating the result sets obtained from equations (5.1) and (5.2), we determine constant values as $E_1 = 40$ and $E_2 = 10$ approximately.

When the key-frames are normalized by the algorithm by using these computed values E_1 and E_2 (equation 1), the eigenvectors and eigenvalues are calculated for each set of key-frames corresponding to a particular presenter. These eigenvectors describe a set of axes within the facespace, along which there is the most variance in the faces and the corresponding eigenvalues represent the degree of variance along these axes. Eigenvectors for all the presenters in the database were calculated for the key frames needed to be recognized. Once the eigenfaces have been computed, each face can be viewed in the facespace (Figure 5.9). There are four possible results when a face is projected to the eigenspace.

The results are;

1. Projected face can be a known face and it is mapped to a point near to the cluster belong to the known presenter.
2. Projected face can be a known face and it is mapped to a point far away from the cluster belong to the known presenter (False Recognition).
3. Projected face can be an unknown face and it is mapped to a point near to the cluster belong to the known presenter (False Acceptance).
4. Projected face can be an unknown face and it is mapped to a point far away from the cluster belong to the known presenter

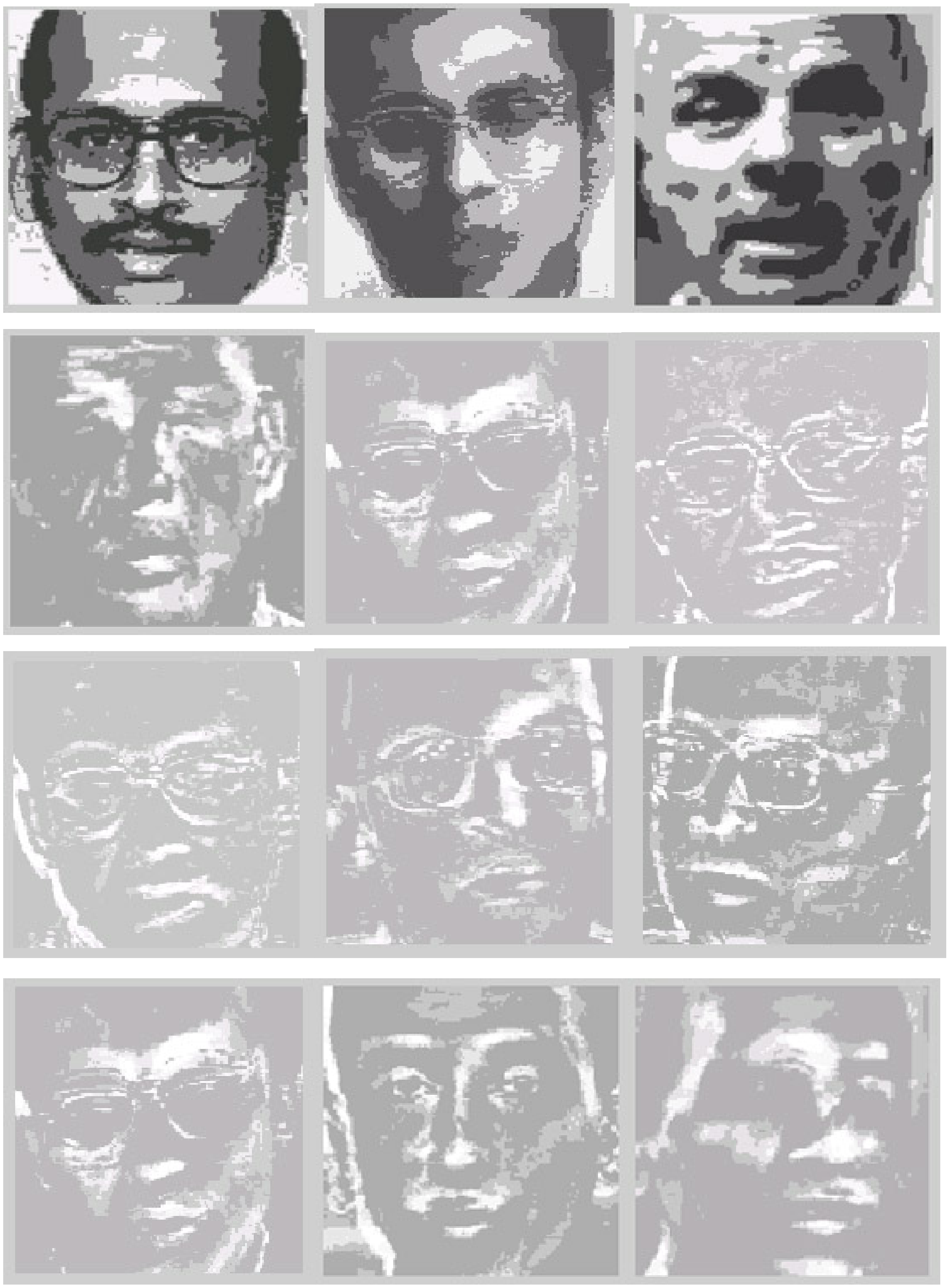


Figure 5.9:Eigenfaces generated after applying the normalization algorithm to video key-frames

5.2 Projection of Profiles

Each profile can be viewed as a set of features. When a presenter’s face is projected onto the facespace, its vector (made up of its weight values with respect to each eigenface) in the face space describes the importance of each of those features in the face. Figures 5.10 and 5.11 describe this process pictorially. In order to reconstruct the original face from the selected set of eigenfaces, we have to build a kind of weighted sum of eigenfaces. That is, the reconstructed original face is equal to a sum of the eigenfaces, with each eigenface having a certain weight. This weight specifies, to what degree the specific feature (eigenface) is present in the original face [Turk and Pentland 1991].

In Figures 5.10 and 5.11, a face is developed into the facespace. The face is described in the face space by its eigenface coefficients (or weights). In Figures 5.10, The Face is developed using the original presenter’s face and in figure 5.11 the face is developed after applying the normalizing algorithm. Since the face developed in the face space is indeed a face, the weight of the first eigenface should be very high, almost equal to unity [Cuevas H. and Rudomín I. 2000]. The value of the weights decreases as the number of the eigenface increases. This is in conformity with the definition of eigenfaces.

In fact, in figure 5.10 the weights are lower than the weights in figure 5.11. These experiments have proven that after normalizing, the effectiveness of the profile projection has significantly improved.

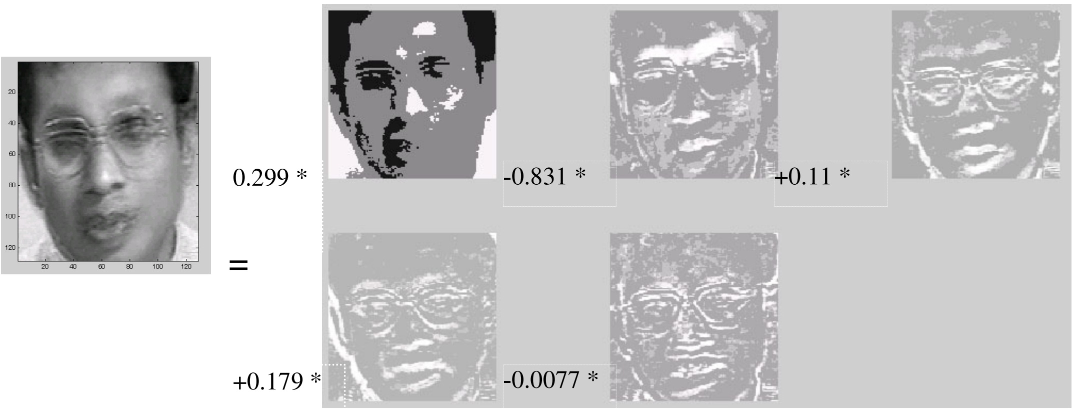


Figure 5.10: A profile developed without applying the normalizing algorithm

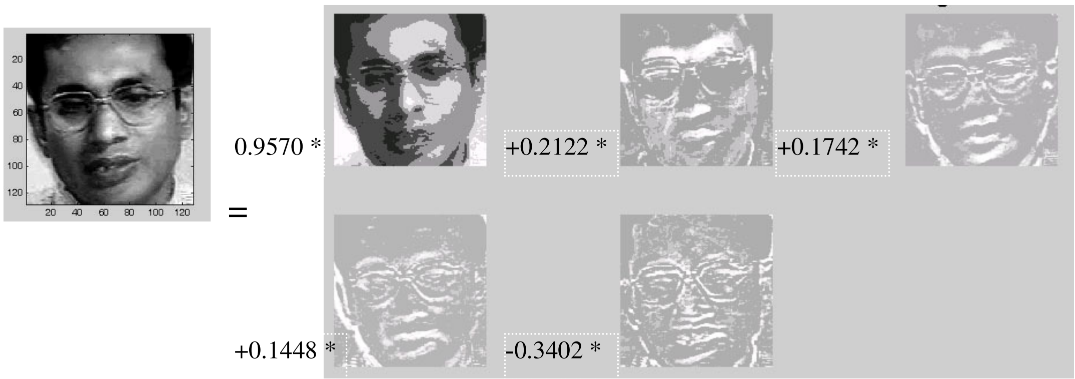


Figure 5.11: A profile developed after applying the normalizing algorithm

Chapter 6

6 Evaluation

The Techniques that are explained in this thesis have been evaluated by developing a prototype system and by using different data sets. The normalization algorithm is evaluated based on its ability to improve recognize faces under variations in lighting condition. Since there is no standard database that contains large 2-D faces suitable for video-based face recognition, a database of 200 video key frames of 20 individuals was used to evaluate presenter recognition. These educational video clips are acquired from the Bachelor of information technology (BIT) external degree program which is conducted by the University of Colombo School of computing. Frontal face key frames with lighting variations are selected from the database. Furthermore the training and test video key frames of the presenters have been taken from different shots from video clips although in a few cases lighting conditions are similar.

The effectiveness of our technique is evaluated by using two of the most widely used criterions, false acceptance rate (FAR) and false rejection rate (FRR) are defined below.

$$\text{False acceptance rate (FAR)} = \frac{\text{Number of Invalid faces who are incorrectly accepted as genuine face}}{\text{Total number of accepted faces}} \text{ X } 100 \quad (6.1)$$

$$\text{False rejection rate (FRR)} = \frac{\text{Number of Valid faces who are Incorrectly rejected as impostors}}{\text{Total number of rejected Faces}} \text{ X } 100 \quad (6.2)$$

Verification tests are carried out to gather FAR and FRR results from a data set comprised of key-frames that present typical difficulties when attempting recognition, such as strong variations in lighting direction and intensity. The Total Error Rate (TER) which is FAR + FRR is used as a single measure of the effectiveness of the system. Results obtained in our tests are tabulated in Table 6.1 and Table 6.2. To evaluate the effectiveness of our normalization algorithm, the same data set is being used with the conventional PCA approach. The results obtained through this approach is given in table 6.1 and the results is plotted in figure 6.1.

Number of presenters	Total frames (Training set)	Total frames (Test set)	False Acceptance Frames	False Acceptance Rate (FAR)	False recognized frames	False recognition Rate (FRR)	Total Error Rate (TER)	Total Recognition Rate (TRR)
2	10	10	0	0%	0	0%	0%	100%
4	20	20	0	0%	1	5%	5%	95%
6	30	30	1	3.33%	2	6.67%	10%	90%
8	40	40	2	5%	3	7.5%	12.5%	87.5%
10	50	50	4	8%	5	10%	18%	82%
12	60	60	6	10%	7	11.67%	23.33%	78.33%
14	70	70	8	11.43%	9	12.86%	24.29%	75.71%
16	80	80	10	12.5%	11	13.75%	26.25%	73.75%
18	90	90	11	12.22%	14	15.56%	27.78%	72.22%
20	100	100	15	15%	18	18%	38%	67%

Table 6.1: Recognition results obtained using the conventional PCA approach

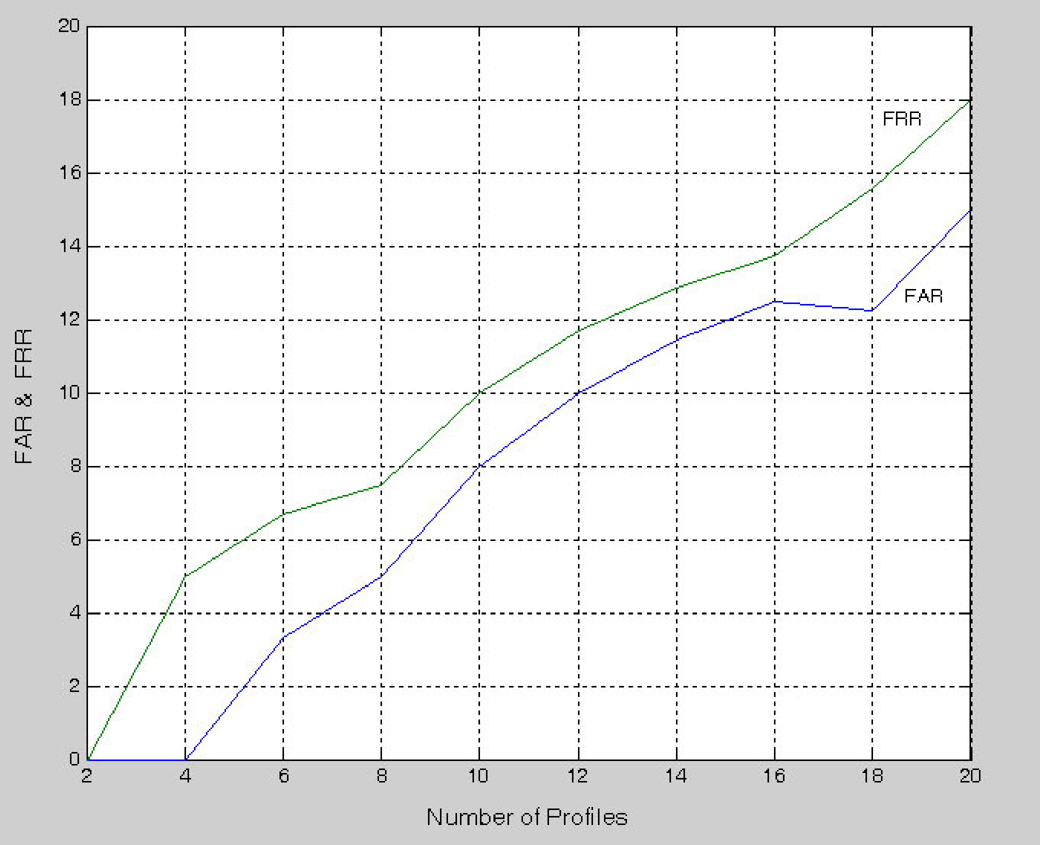


Figure 6.1: Recognition results obtained using the conventional PCA approach

Table 6.2 and in Figure 6.2 shows the results when the normalizing method is used on the same data. In the table 6.2 initial stages of testing, where 2 to 6 presenters were in the dataset the Total Recognition Rate (TRR) was 100% with the normalizing algorithm been applied. Then it decreased to 94% and 81% when the number of profile are increased to 10 and 20 respectively

Using our method of normalization the recognition system achieves a TER of 6% when we tested with 10 presenters (see Table 6.2 and Figure 6.2) and it increased to 19% when we added another 10 presenters to our database.

Number of presenters	Total frames (Training set)	Total frames (Test set)	False Acceptance Frames	False Acceptance Rate (FAR)	False recognized frames	False recognition Rate (FRR)	Total Error Rate (TER)	Total Recognition Rate (TRR)
2	10	10	0	0%	0	0%	0%	100%
4	20	20	0	0%	0	0%	0%	100%
6	30	30	0	0%	0	0%	0%	100%
8	40	40	1	2.5%	1	2.5%	5%	95%
10	50	50	2	4%	1	2%	6%	94%
12	60	60	2	3.33%	3	5%	8.33%	91.67%
14	70	70	4	5.71%	4	5.71%	11.42%	88.58%
16	80	80	5	6.75%	6	7.5%	14.25%	85.75%
18	90	90	8	8.89%	7	7.78%	16.67%	83.33%
20	100	100	9	9%	10	10%	19%	81%

Table 6.2: Recognition results obtained by applying the normalizer

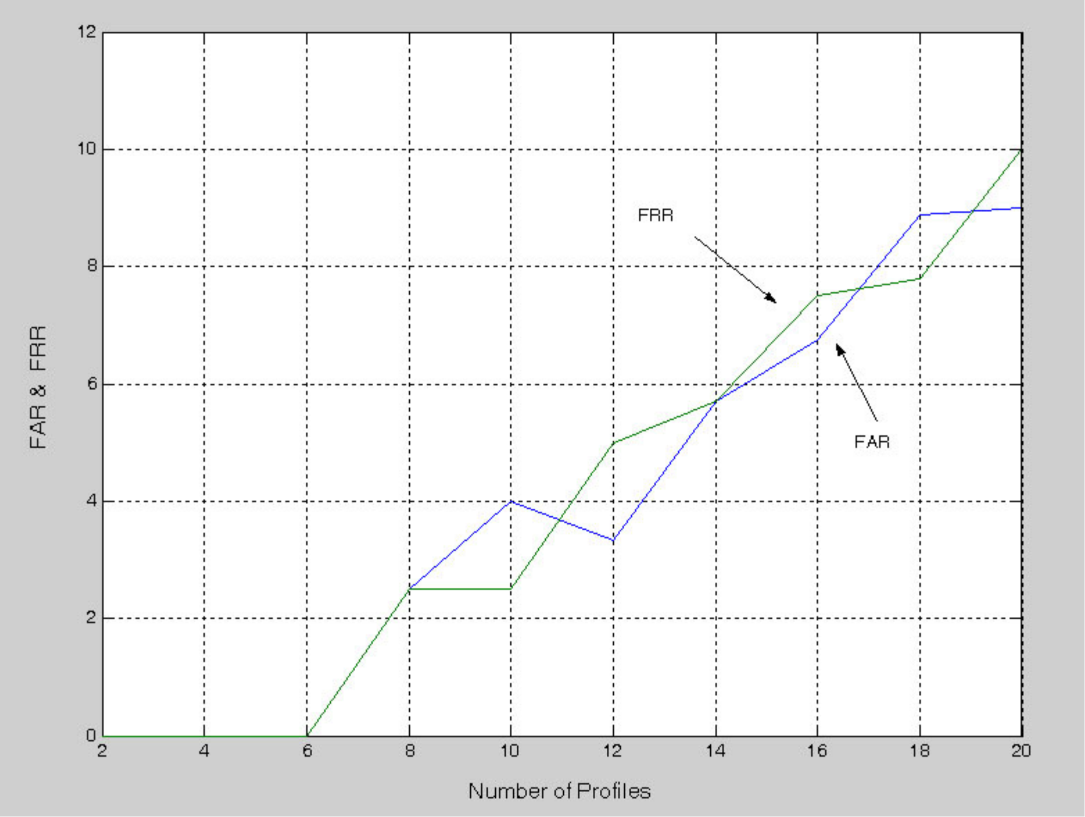


Figure 6.2: Recognition results obtained by applying the normalizer

Figure 6.3 shows the comparison graphically for TER. The results show that the insertion of profile normalizing method reduces TER by 38% to 19%.

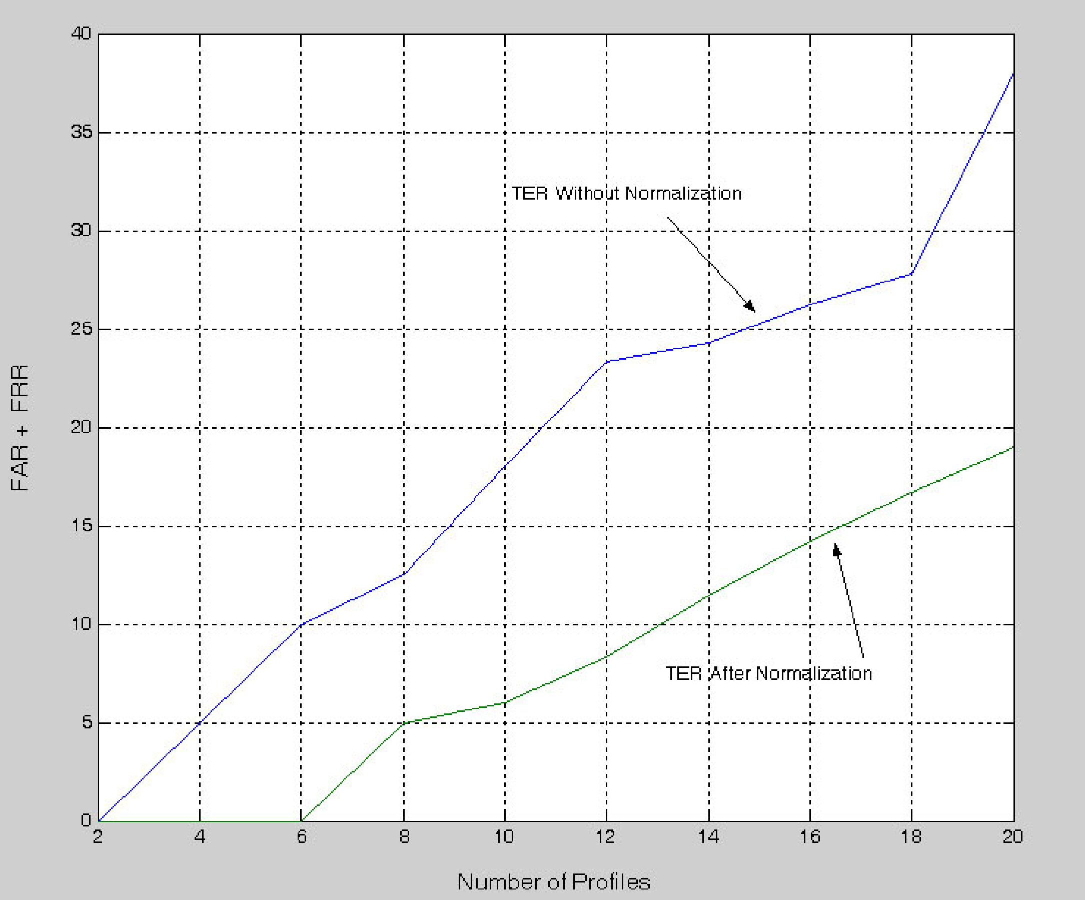


Figure 6.3: Total Error Rate (TER) comparison

Without any alterations to the eigenface technique itself, total error rate of 38% is observed (Figure 6.3). By using our normalizing algorithm the total error rate can be reduced to less than 20%. The algorithm was tested using two different counts of key frames of the same presenter to construct his profile. For the initial testing, 5 frames per presenter was used and for the second testing, the key frames per presenter was increased from 5 to 10. An 80% recognition rate was maintained even when the profile database expanded to 20 (Figure 6.4). The recognition rate with the conventional approach was less than 70%. Results indicate that our methodology is quite robust to both low resolution and luminance changes, which suggest that it can be used for face recognition even when with different lighting conditions.

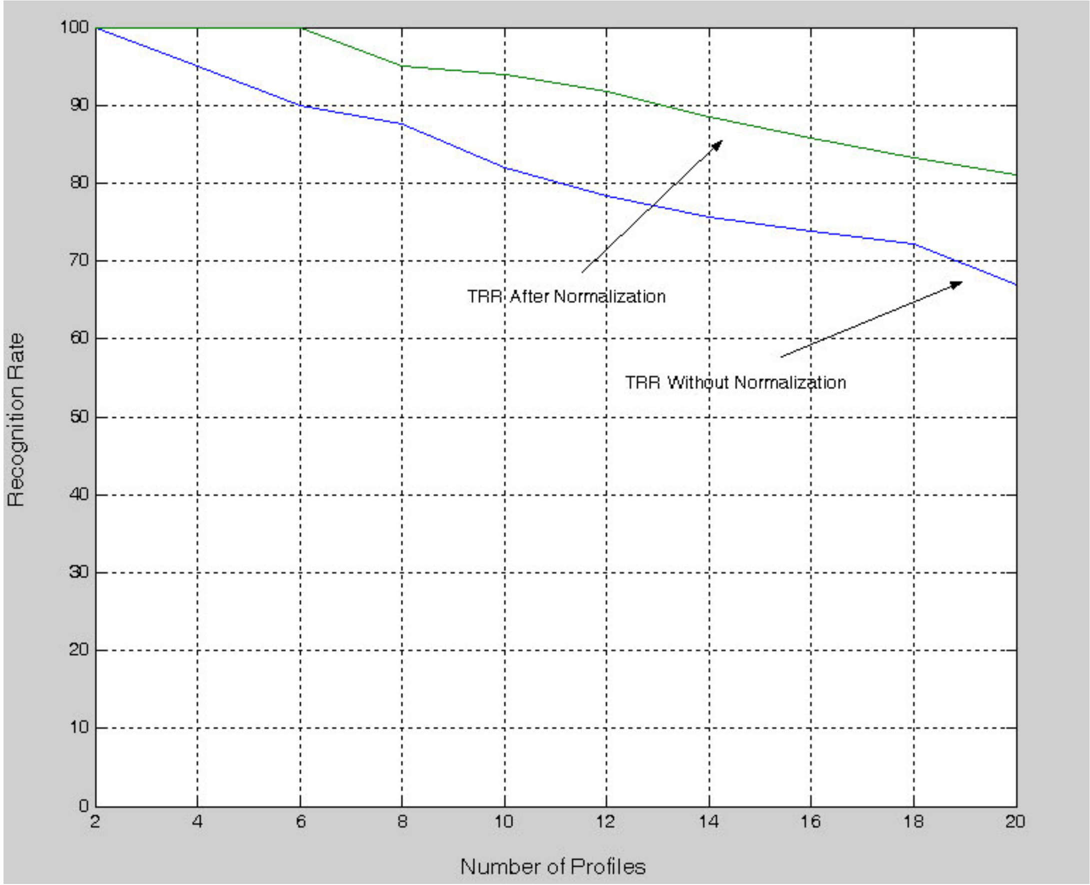


Figure 6.4: Total Recognition Rate (TRR) comparison

The experimental results show that the performance of the proposed method achieves a better success ratio (Figure 6.3 and 6.4). As shown in Figure 6.5, our algorithms can successfully rearrange profiles and overcome the profile overlapping.

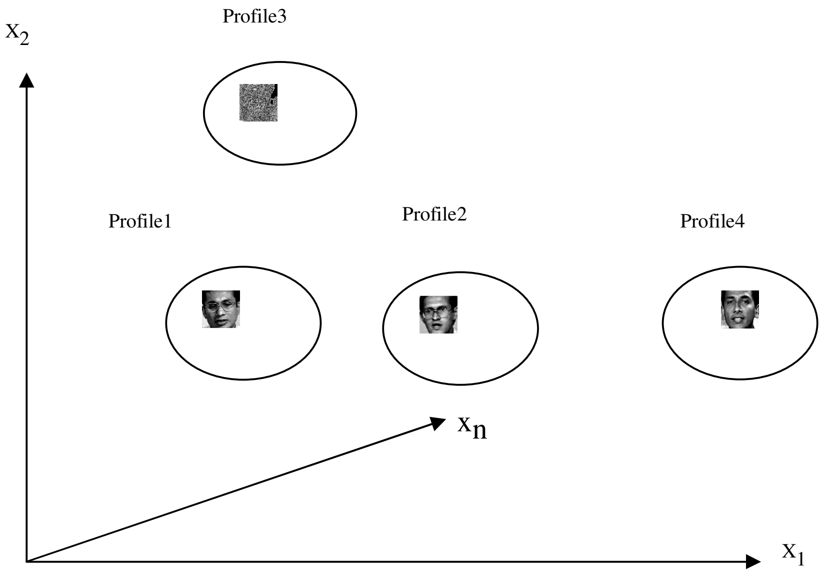


Figure 6.5: Normalized Profiles

The research and development activities was presented to the Japan International Cooperation Agency (JICA) mid-term evaluation team led by Prof. Tsuneo Nitta (Professor of Knowledge based Information Engineering, Toyohashi University of Technology, Japan) on 02/08/2004 and to the final evaluation team led by Prof. Homma Hiroomi (Toyohashi University of Technology, Japan) on 04/03/2005. Also during this experiment, our technique was refried and the approach we have used is validated by publishing 7 papers in international conferences and journals [Premaratne et al. 2004 a, Premaratne et al. 2004 b, Premaratne et al. 2005 a, Premaratne et al. 2005 b, Premaratne et al. 2005 c, Premaratne et al. 2006, Premaratne et al. 2007 a, Premaratne et al. 2007 b].

Chapter 7

7 Conclusion and Future work

7.1 Conclusion

Recognition of faces from a video sequence is still one of the most challenging problems in face recognition because video is of low quality and the resolutions of frames are relatively small. In this work we have looked into one challenging problem in human face recognition: the illumination problem. Our method is capable of dealing with illumination variations in the eigenspace recognition framework and we have shown that the eigenface-based method of face recognition can be significantly improved by means of pre-processing techniques.

The proposed method was extensively evaluated on a database of 20 presenter profiles with varying illumination. By using the techniques we have introduced, a TER of 19% percent can be achieved (a 19% reduction of error rate from the initial method) using a data set containing difficult lighting conditions. The approach described above shows encouraging initial results for a wholly automatic system operating on real world data (e-learning video). Such results are practical for certain simple video indexing problems, such as labeling educational learning materials.

There are some factors that may be the cause of the remaining 19% error, which were not compensated by our techniques. This could happen since variation in pose is associated with the key-frames. Pose discrimination is not difficult but accurate pose estimation is hard to accomplish. However, a number of simplifications have been made to the initial implementation of the system, and further research is needed to optimize and improve upon these methods.

From the final results obtained we can conclude that the new algorithm proposed in this thesis works well under controlled environments and the recognition algorithm took advantage of the environmental constraints to obtain high recognition accuracy.

7.2 Future work

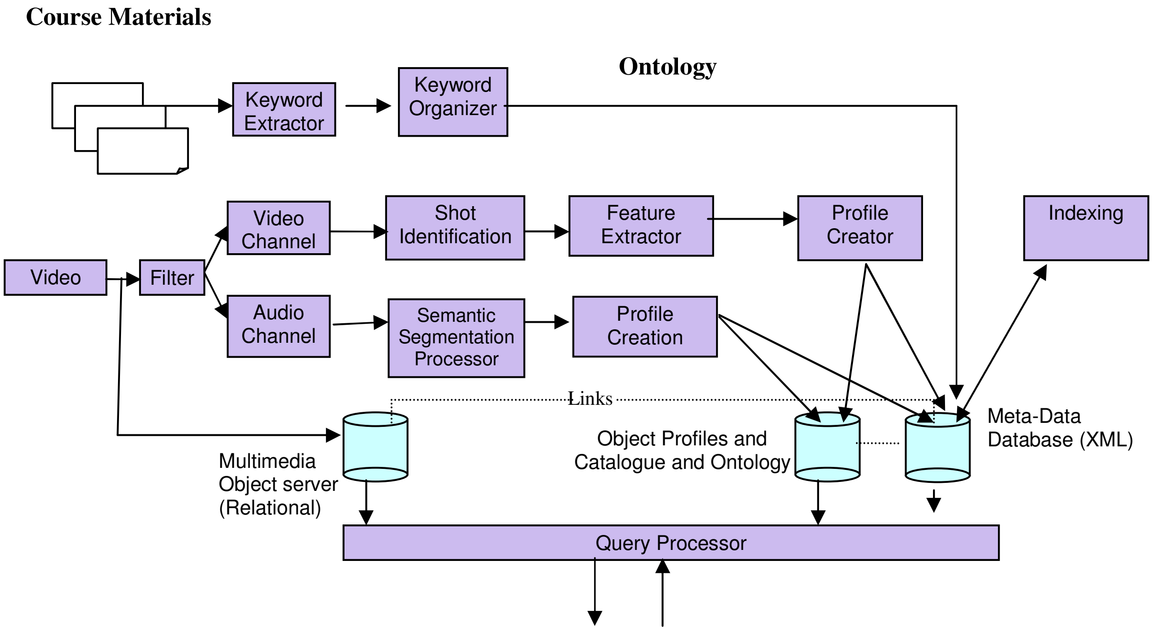


Figure 7.1: Overall System Architecture

This research will contribute to the overall system that has been designed by the research group (Figure 7.1). In addition to automatic analysis and modeling of the features of the video content, the system is designed to exploit the use of speech indexing to combine our approach for video retrieval. The speaker recognition application is to identify speakers in a given audio clip which is separated from the original audio-visual by using a filter. When the presenter identification system described in this paper is combined with a speaker identification system, it gives more accuracy on retrieval process of video lectures. This application developed for presenter identification does have limitations such as not belonging to the speaker and when there are multiple faces in a video segment. In these situations, the system can compare the results given by both speaker identification and presenter identification systems to verify that the video face is the current speaker and then it can be applied to create the video index (Figure 7.2). The video caption recognition process which is developed to extract key words from video key-frames will provide more metadata annotation facilities to the system. Ontology–based information systems have been developed to structure course materials like course notes, presentations and past exam papers to support information retrieval (Figure 7.1). Consequently this will add more value to the convenient, efficient and effective multimedia data retrieval process. By combining these techniques and tools a complete system can be implemented to store and retrieve multimedia educational materials which will be semi-automatically trained and tested on a data set of real video lectures.

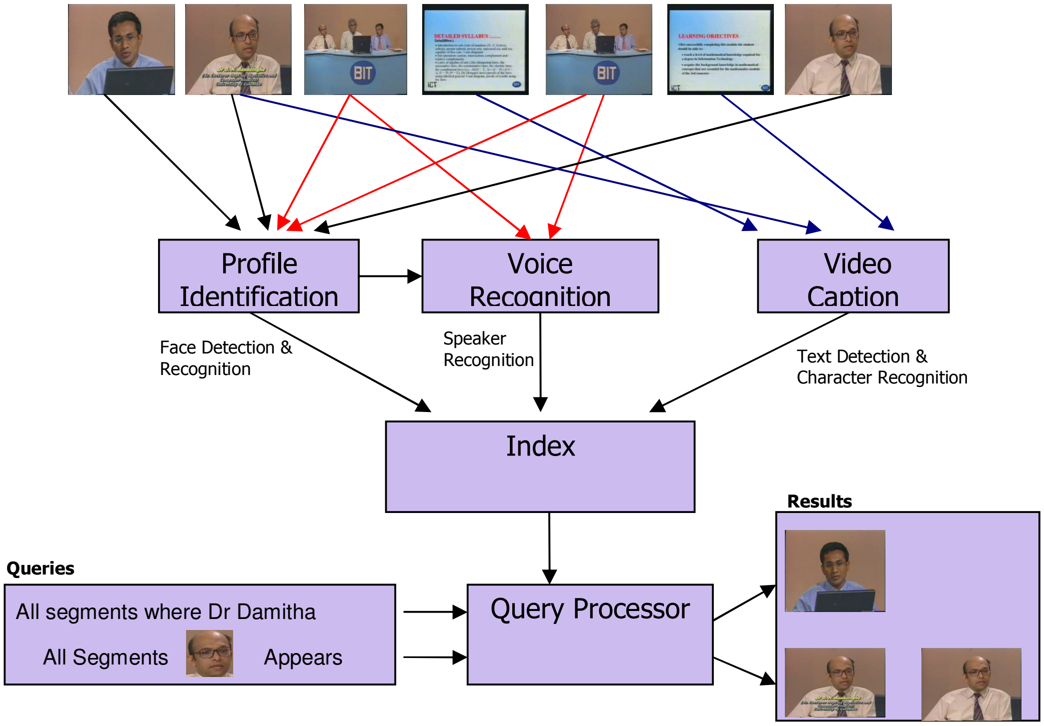


Figure 7.2: Implementation of the System

The work that had been done can be expanded in several directions. The algorithm can be improved in order to recognize video key-frames such as identify presenter in different poses and scale, although our system works well under small variations in orientation and scale. Development of multi-scale capabilities into our system would significantly add efficiency to the real-time application. Techniques which could be used to achieve multi-scale recognition include scale-based eigenspace and scale estimation based on frame analysis. These areas are proposed as the focus of future work on this research.

All current person 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 and it is worth pointing out here that the state-of-the-art person authentication systems are not meant for un-manned operation.

References

Abowd, G. D., Brotherton, J.A. and Bhalodai, J., 1998. Classroom 2000: A system for capturing and accessing multimedia classroom experiences.

Adnin, Y., Moses, Y. and Ullman, S., 1997. Face Recognition: The problem of compensating for changes in illumination direction. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 19, No. 7, p. 712-732.

Baek, K. B., Draper, **A**., Beveridge, J. R. and She, K., 2002. PCA vs ICA: A comparison on the FERET data set. Presented at Joint Conference on Information Sciences, Durham, N.C.

Bartlett, M.S., Movellan, J.R. and Sejnowski, T.J., 2002. Face Recognition by Independent Component Analysis, IEEE Trans. On Neural Networks, Vol. 13, No. 6, p. 1450-1464.

Bartlett, M. and Sejnowski, T., 1997. Independent components of face images: A representation for face recognition. Proceedings of the 4th Annual Joint Symposium on Neural Computation, Pasadena, California.

Bicego, M., Castellani, U. and Murino, V., 2003. Using Hidden Markov Models and Wavelets for face recognition. Proc. of IEEE Int. Conf. on Image Analysis and Processing (ICIAP03), p. 52-56.

Belhumeur, P. N., Hespanha, J. P. and Kriegman, D. J., 1997. Eigenfaces vs Fisherfaces Recognition Using Class Specific Linear Projection. IEEE Transactions on Pattern Analysis and Machine Intelligence, p. 45-58.

Belhumeur, P.N. and Kriegman, D.J., 1996. What is the Set of Images of an Object Under All Possible Lighting Conditions?. Computer Vision and Pattern Recognition. Proceedings CVPR '96, p. 270-277.

Beymer, D. J., 1997. Face Recognition Under Varying Pose. Pattern Analysis and Machine Intelligence, IEEE Transactions on.Volume: 19, p. 711-720.

Bimbo, A., 2000. Semantics based retrieval by content. In IEEE International Conference on Image Processing, volume 3, pages 516-519.

Biswas, P. K. and Pandit, M., 2002. OPTI-GVF Snake Model for Face Segmentation From Video Sequence. ICVGIP.

Boreczky, J.S. and Rowe, L.A., 1996. Comparison of video shot boundary detection techniques. In *s*torage and Retrieval for Still Image and Video Databases, p. 170-179.

Broomhead, D.S. and Kirby, M. A., 2000. New Approach to Dimensionality Reduction: Theory and Algorithms. SIAM Journal of Applied Mathematics, vol. 60, no. 6, p. 2114-2142.

Brunelli, R. and Poggio, T., 1993. Face Recognition: Features versus Templates. IEEE Trans. Pattern Analysis and Machine Intelligence, p. 1042 – 1052.

Calic, J. and Izquierdo, E., 2001. Towards Real-Time Shot Detection in the MPEG Compressed Domain. Proceedings of the Workshop on Image Analysis for Multimedia Interactive Services, WIAMIS'2001, Tampere, Finland.

Calic, J. and Izquierdo, E., 2002. Efficient Key-Frame Extraction and Video Analysis. ITCC, Las Vegas, Nevada, USA.

Calic, J. and Thomas, B. T., 2004. Spatial Analysis in Key-frame Extraction Using Video Segmentation. Proc. of 5th International Workshop on Image Analysis for Multimedia Interactive Services, WIAMIS'2004, Instituto Superior Técnico, Lisboa, Portugal.

Campos, T. E., Feris, R. S. and Cesar-Jr, R. M., 2000. A framework for face recognition from video sequences using gwn and eigenfeature selection. Proceedings of I-WAICV, p. 141-145.

Chennubhotla, C., Jepso, A. D. and Midgley, J., 2002. Robust contrast-invariant eigendetection. ICPR02, p. 745-748.

Christel, M., Olligschlaeger, A. and Hung, C., 2000. Interactive Maps for a Digital Video Library, IEEE Multimedia 7(1), p. 60-67.

Cuevas, H. and Rudomín, I., 2000. Generating a 3D Facial Model From a Single Image using Principal Components Analysis. Proceedings of Visual 2000.

Deshpande, S. G. and Hwang, J. N., 2001. A real-time interactive virtual classroom multimedia distance learning system. IEEE Trans. on Multimedia, p. 432 – 444.

Dinggang, S. and Horace, H.S., 1997. Generalized affine invariant image normalization. IEEE Transactions on PAMI, p. 431-440.

Dongge, L. and Sethi, I. K., 1999. MDC: A software tool for developing MPEG applications. Proceedings IEEE International Conference on Multimedia Computing and Systems, vol. 1, p. 445-450.

Dorai, C., Kermani, P. and Stewart A., 2001. Elm-n: e-learning media navigator. ACM Multimedia, p. 634–635.

Epstein, R., Hallanan, P. and Yuille A. L., 1995. 5±2 Eigenimages Suffice: An Empirical Investigation of Low-Dimensional Lighting Models. IEEE Conf. Workshop on Physics-Based Vision, p. 108-116.

Feraud, R., 1998. PCA, Neural Networks and Estimation for Face Detection. In Face Recognition: From Theory to Applications, Springer-Verlag.

Feris, R. S., Campos, T. E. and Cesar Junior, R. M., 2000. Detection and Tracking of Facial Features in Video Sequences. Lecture Notes in Artificial Intelligence, vol. 1793, p. 127-135.

Finlayson, G.D., Schiele, B. and Crowley, J.L, 1998. Comprehensive colour image normalization. ECCV.

Fr¨oba, B., Ernst, A. and K¨ublbeck, C., 2001. Real-Time Face Detection. Third International Conference on Audio- and Video-Based Biometric Person Authentication.

Georghiades, A. S., Belhumeur, P. N. and Kriegman, D. J., 1999. Illumination-Based Image Synthesis: Creating Novel Images of Human Faces Under Differing Pose and Lighting. IEEE Workshop on Multi-View Modeling and Analysis of Visual Scenes, p. 47-54.

Georghiades, A. S. and Belhumeur, P. N., 2001. From Few to many: Illumination cone models for face recognition under variable lighting and pose. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 23, No. 6, p. 643-660.

Graves, A. P. and Lalmas, M., 2002. Video retrieval using an MPEG-7 based inference network, ACM SIGIR International Conference on Research and Development in Information Retrieval, Tampere, Finland. p. 339-346.

Gunsel, B., Fu, Y. and Tekalp, A. M., 1997. Hierarchical temporal video segmentation and content characterization, Multimedia Storage and Archiving Systems II, SPIE, 3229, p. 46-55.

Guo, G., Li, S. Z. and Chan, K., 2000. Face Recognition by Support Vector Machines, School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798.

Hauptmann, A.G., 1999. Integrating and Using Large Databases of Text, Images, Video and Audio. IEEE Intelligent Systems, vol. 14, no. 5, p. 34 - 35.

Jain, A., 1989. Fundamentals of Digital Image Processing. Prentice Hall. p 240 - 245.

Kawato, S. and Ohya, J., 2000. Two-step approach for real-time eye tracking with a new filtering technique. In International Conference on Systems, Man and Cybernetics.

Kobla, V., Doermann, D., Lin, K. and Faloutsos, C., 1997. Compressed domain video indexing techniques using DCT and motion vector information in MPEG video. Storage and Retrieval for Image and Video Databases (SPIE), p. 200-211.

Kohonen, T., 1988. Kohonen's Self-Organizing Map. Published by Springer Verlag, New York, 3rd Edition.

Kosch, H., 2000. MPEG-7 and Multimedia Database Systems. SIGMOD Records, ACM Press, p. 34–39.

Krüger, V., Happe, A. and Sommer, G., 2000. Affine Real-Time Face Tracking using Gabor Wavelet Networks. ICPR. p, 1127-1130.

Kuchi, P., Gabbur, P., Bhat, P. S. and Davis, S., 2002. Human face detection and tracking using skin color modeling and connected component operators. IETE Journal of Research, Vol. 48, p. 289-93.

Lades, M., Vorbriiuggen, J. C., Buhmann, J., Lange, J., Malsburg, C., Wiiurtz, R. P. and Konen. W., 1993. Distortion Invariant Object Recognition in the Dynamic Link Architecture. IEEE Trans. Computers, vol. 42, p. 300-311.

Lawrence, S., Giles, C. L., Tsoi, A. C. and Back, A. D., 1997. Face Recognition: A Convolutional Neural Network Approach. IEEE Transactions on Neural Networks, Special Issue on Neural Networks and Pattern Recognition, Volume 8, Number 1, p. 98–113,.

Lincoln, M. C. and Clark, A. F., 2000. Pose-independent face identification from video sequences. In Proceedings of the Third International Conference on Audioand Video-Based Biometric Person Authentication, Halmstad , Sweden, Springer-Verlag, volume LNCS 2091, p. 14 – 19.

Liu, M., Chang, E. and Dai, B., 2002. Hierarchical Gaussian Mixture Model for Speaker Verification. Proceedings International Conference on Spoken Language Processing,.

Liu, C. and Wechsler, H., 1999. Comparative Assessment of Independent Component Analysis (ICA) for Face Recognition. International Conference on Audio and Video Based Biometric Person Authentication, Washington, D.C.

Long, F., Feng, D. and Peng, H., 2001. Extracting Semantic Video Objects. IEEE Computer Graphics and Applications Volume 21, Issue 1, p. 48 – 55.

Lorente, L. and Torres, L., 1998. Face Recognition of Video Sequences in a MPEG-7 Context Using a Global Eigan Approach. International Workshop on Very Low Bit-rate Video Coding, Urbana, Illinois.

Moghaddam, B., Jebara, T. and Pentland, A., 2000. Bayesian Face Recognition. Pattern Recognition, Vol 33, Issue 11, p. 1771-1782.

Moghaddam, B., 1999. Principal Manifolds and Bayesian Subspaces for Visual Recognition. International Conference on Computer Vision, Corfu, Greece, p. 1131-1136.

Norris, J. S., 1999. Face Detection and Recognition in Office Environments. Massachusetts Institute of Technology.

Palanivel, S., Venkatesh, B. S. and Yegnanarayana, B., 2003. Real time face authentication using autoassociative neural network models. IEEE International Conference On Multimedia and Expo, Baltimore, USA.

Papageorgiou, C. P., Oren, M. and Poggio, T., 1998. Proceedings of the Sixth International Conference on Computer Vision, p. 555-563.

Park, S., Park, J. and Aggarwal, J. K., 2003. Video Retrieval of Human Interaction Using Model-Based motion Tracking and Multi layer Finite State Automata. International Multimedia Conference, p. 65-76.

Pei, S. and Chou, Y., 1999. Efficient MPEG compressed video analysis using macroblock type information. IEEE Transactions on Multimedia. Vol.1 p. 321-333.

Pentland, A., Moghaddam, B. and Starner, T., 1994. View-based and modular eigenspaces for face recognition", MIT Media Laboratory Perceptual Computing Section, Technical Report 245.

Phillips, P. J., Flynn, P. J., Scruggs, T., Bowyer, K. W., Chang, J., Hoffman, K., Marques, J., Min, J. and Worek, W., 2005. Overview of the Face Recognition Grand Challenge, 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1, p. 947-954.

Pissarenko, D., 2002. Eigenface-based facial recognition. http://openbio.sourceforge.net/resources/eigenfaces/eigenfaces.pdf

Premaratne, S. C., Karunaratna, D. D., Wikramanayake, G. N., Hewagamage, K. P. and Dias, G. K. A., 2004a. An Architecture of a Media Based System to Support E-Learning. The Bulletin of the British Computer Society Sri Lanka Section, October, p. 32-33.

Premaratne, S. C., Karunaratna, D. D., Wikramanayake, G. N., Hewagamage, K. P. and Dias, G. K. A., 2004b. Profile Based Video Segmentation System to Support E-Learning. Proceedings of the 6th International Information Technology Conference. Colombo, Sri Lanka, p. 74-81.

Premaratne, S. C., Karunaratna, D. D., Wikramanayake, G. N., Hewagamage, K. P. and Dias, G. K. A., 2005a. Implementation of a Profile Based Video Segmentation System. Proceedings of the International Conference on Information Management in a Knowledge Society, Grand Hyatt Mumbai, Maharashtra, India, p. 89-100.

Premaratne, S. C., Karunaratna, D. D., Wikramanayake, G. N., Hewagamage, K. P. and Dias, G. K. A., 2005b. Efficient Profile Construction Algorithm for Video Indexing in E-Learning. Proceedings of the 11th International Conferenceon Virtual Systems and multimedia, Flanders Expo, Ghent, Belgium, p. 65-74

Premaratne, S. C., Karunaratna, D. D., Wikramanayake, G. N., Hewagamage, K. P. and Dias, G. K. A., 2005c. Improvised Profile Construction for Multimedia Databases in E-Learning. Proceedings of the MMU International Symposium on Information and Communication Technology. Kuala Lampur, Malaysia, TS12, p. 9-13

Premaratne, S. C., Karunaratna, D. D. and Hewagamage, K. P., 2006. Profile Based Video Browsing for E-Learning. Proceedings of the 10th IASTED International Conference on Software Engineering and Applications. Dallas, Texas USA, p. 489-494.

Premaratne, S. C., Karunaratna, D. D. and Hewagamage, K. P., 2007. Collaborating Educational Videos with Presenter Profiles for Effective Content-based Video Retrieval. Proceedings of the Digital Learning Asia 2007. http://www.digitallearning.in/dlasia/2007/agenda_day3_3.asp.

Romdhani, S., Blanz, V. and Vetter, T., 2002. Face Identification by Fitting a 3D Morphable Model using Linear Shape and Texture Error Functions. Proceedings of the 7th European Conference on Computer Vision-Part IV, p. 3-19.

Rowley, H. A., Baluja, S. and Kanade, T., 1998. Neural Network-Based Face Detection. PAMI.

Sandeep, K. and Rajagopalan , A. N., 2002. Human Face Detection in Cluttered Color Images Using Skin Color and Edge Information. Department of Electrical Engineering Indian Institute of Technology – Madras, Chennai – 600 036, India

Satoh, S., Sato, T., Smith M., Nakamura, Y. and Kanade, T., 1996. Name-It: Naming and Detecting Faces in News Video. Network-Centric Computing (NCC) Special Issue.

Spaniol, M., Klamma, R. and Jarke, M., 2002. Data Integration for Multimedia E-learning Environments with XML and MPEG-7. Springer LINK, p. 244-255.

Sim, T., Baker, S., and Bsat, M., 2002. The CMU Pose, Illumination, and Expression (PIE) Database**.** Proceedings of the IEEE International Conference on Automatic Face and Gesture Recognition.

Sun, F., Omachi, S., Kato, Nei, Aso, H., Kono, S. and Takagi, T., 2000. Two-Stage Computational Cost Reduction Algorithm Based on Mahalanobis Distance Approximations. 15th International Conference on Pattern Recognition (ICPR'00) - Volume 2 p. 2696.

Szlávik, Z. and Szirányi, T., 2003. Face identification using CNN-UM, Proc. of ECCTD'03, vol. 2, p. 81-85.

Tritschler, A. and Gopinath, R. A., 1999. Improved Speaker Segmentation and Segments Clustering Using the Bayesian Information Criterion. Sixth European Conference on Speech Communication and Technology.

Turk, M. and Pentland, A., 1991. Eigenfaces for Recognition. Journal of Cognitive Neuroscience, p. 71-86.

Vezhnevets, V., 1998. Method for Localization of Human Faces in Color Based Face Detectors and Trackers. Department of Computational Mathematics & Cybernetics. Moscow State University, Moscow, 119899, Russia.

Viola, P. and Jones, M. 2001a. Fast and Robust Classification using Asymmetric AdaBoost and a Detector Cascade. Neural Information Processing Systems.

Viola, P. and Jones, M., 2001b. Robust Real-time Object Detection", Second International Workshop on Statistical and computational theories of vision Canada.

Wang, H. and Chang, S., 1996. Automatic face region detection in MPEG video sequences. Proc. SPIE. Electronic Imaging and Multimedia Systems, p. 160-168.

Wang, H., Li, S. Z., Wang, Y., and Zhang, W., 2003. Illumination Modeling and Normalization for Face Recognition. Proceedings of IEEE International Workshop on Analysis and Modeling of Faces and Gestures, Nice, France.

Wiskott, L., Fellous, J. M., Kruger, N. and Malsburg, C. V., 1995. Face Recognition and Gender Determination. International Workshop on Automatic Face- and Gesture-Recognition, Zürich, p. 92-97.

Wiskott, L., Fellous, J. M., Krüger, N., and von der Malsburg, C. V., 1997. Face Recognition by Elastic Bunch Graph Matching**.** IEEE Transactions on Pattern Analysis and Machine Intelligence, p. 775-779.

Wang, H. and Wang, Y., 2003. RECOGNIZING FACE IMAGES UNDER DIFFERENT LIGHTING CONDITIONS. International Conference on Acoustic, Speech and Signal Processing (ICASSP).

Wang, J. Y. A. and Adelson, E. H., 1994. Representing Moving Images with Layers. IEEE Transactions on Image Processing, p. 625-638.

Xu, C. and Prince, J. L., 1998. Snakes, Shapes, and Gradient Vector Flow. IEEE Ttarnsactions on Image Processing. Vol. 7, No. 3

Yang, M. H., Kriegman, D. J. and Ahuja, N., 2002. Detecting Faces in Images: A Survey. IEEE Transactions on Pattern Analysis and Machine Intelligence, p. 34-58.

Yeo, B. L. and Liu, B., 1995. Rapid scene analysis on compressed video. IEEE Transactions on Circuits & Systems for Video Technology. p. 533-544.

Yilmaz, A. and Gokman, M., 2001. Eigenhill vs. Eigenface and Eigenedge. Pattern Recognition Journal, Vol. 34 , p. 181-184.

Yuen, P. C. and Lai, J. H., 2000. Independent Component Analysis of Face Images. IEEE Workshop on Biologically Motivated Computer Vision. Seoul: Springer-Verlag.

Zabih, R., Miller, J. and Mai, K., 1995. A feature-based algorithm for detecting and classifying scene breaks. Proc. ACM Multimedia, p. 189-200.

Zhang, H., Kankanhalli, A. and Smoliar, W., 1993. Automatic partitioning of full-motion video. Multimedia Systems, p. 10-28.

Zhang, J., Yan, Y. and Lades, M., 1997. Face Recognition: Eighenface elastic matching and neural nets. Proceedings of the IEEE, Vol. 85, No. 9, p. 1423-1435.

Zhu, Yi. and Cutu, F., 2002. Face Detection using Half-Face Templates. Technical Report TR572.