A
Dissertation
On

STUDY OF FACE RECOGNITION TECHNIQUES USING VARIOUS MOMENTS

Submitted in partial fulfillment of the requirement
For the award of degree of

Master of Technology
In
Signal Processing and Digital Design

Submitted by

PALLAVI MATHUR

Roll No. DTU/M.Tech/173

Under the esteemed guidance of

Dr. RAJIV KAPOOR

Department of Electronics and Communication Engineering

DELHI TECHNOLOGICAL UNIVERSITY

2010-2012

DELHI TECHNOLOGICAL UNIVERSITY

DELHI- 110042

CERTIFICATE

This is to certify that the work contained in this dissertation entitled "**Study of Face Recognition Techniques using various Moments**" submitted in the partial fulfillment, for the award for the degree of M.Tech in Signal Processing and Digital Design at **DELHI TECHNOLOGICAL UNIVERSITY** by **PALLAVI MATHUR, Roll No. DTU/M.Tech/173,** is carried out by her under my supervision. The matter embodied in this project work has not been submitted earlier for the award of any degree or diploma in any university/institution to the best of my knowledge and belief.

Date: ___________

(Dr. RAJIV KAPOOR)

HOD & Project Guide

Department of Electronics and Communication Engineering

Delhi Technological University

ACKNOWLEDGEMENT

It is a great pleasure to have the opportunity to extend my heartfelt gratitude to everybody who helped me throughout the course of this project.

First of all, I thank my parents who made me capable of taking up this project. They have always motivated and blessed me for all my endeavors.

It is a distinct pleasure to express my deep sense of gratitude and indebtedness to my learned supervisor Dr. Rajiv Kapoor, HOD, Department of Electronics Engineering, Delhi Technological University, for his invaluable guidance, encouragement and patient review. It was indeed a privilege and honor to work under his supervision.

I would like to extend my earnest and heartfelt thankfulness to my in laws who were a constant motivation for me not only during the project work but throughout the M. Tech program. Their support gave me the inner strength to sail smoothly through this journey.

Words fail me while I thank my husband Mr. Pranav Bhatnagar for his inspiration, motivation, endurance and active support during the entire tenure.

I take this opportunity to appreciate all my friends at DTU, in particular Indu Sirohi, Rashmi Gautam and Jaya Choudhary for their constant support, help and making this work duration jovial.

Pallavi Mathur

Roll no: DTU/M.Tech/173

M.Tech. (Signal Processing and Digital Design)

Department of Electronics and Communication Engineering

Delhi Technological University

Delhi-110042

ABSTRACT

The field of face recognition has been explored a lot and the work is still going on. In the presented work we have proposed a novel approach for face recognition using moments. Four methods have been used for feature extraction: Hu moments, Zernike moments, Legendre moments and Cumulants. Hu moments are a set of seven moments which have been derived from the conventional geometric moments. These are invariant against rotation, scaling and translation. Legendre moments and Zernike moments have an orthogonal basis set and can be used to represent an image with a minimum amount of information redundancy. They are based on the theory of orthogonal polynomials and can be used to recover an image from moment invariants. Cumulants are sensitive to the image details and therefore are suitable for representing the features of images. For feature extraction, moments of different orders are calculated which form the feature vectors. The obtained feature vectors are stored in the database and are classified using three classifiers: Minimum Distance Classifier, Support Vector Machine and K Nearest Neighbor. For testing the proposed approach, the ORL (Olivetty Research Laboratories) database is used. It consists of 40 subjects, each having 10 orientations.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Dedication

I dedicate this thesis

To my family, my teachers and my friends for

Supporting me all the way & doing all the

Wonderful things for me.

Chapter 1

INTRODUCTION

Recognition of faces is something that people can usually do without any effort and much conscious thought. But it has remained a difficult problem in the area of computer vision. A face recognition system in simple words can be described as a computer application for automatically identifying or verifying a person from a digital image or a video frame from a video source.

1.1 Motivation

Face recognition has become a necessity in today's world. It is required to fulfill the higher levels of security required these days. Today, face recognition technology is being used in many areas like to combating passport fraud, supporting law enforcement, identifying missing children, minimizing benefit/identity fraud etc. Although there are many other methods for recognizing persons like iris recognition, finger print recognition etc., but Face recognition is particularly attractive, because it concentrates on the same identifier that humans use mainly to distinguish one person from another which is their "faces".

Face recognition, since long has been an area of computer vision, but only in recent few years reliable and automated face recognition has become a realistic target. New algorithms and developments along with the falling costs of cameras and the increasing availability of processing power has led to practical face recognition systems. These systems are increasingly being deployed and used in many practical applications worldwide and further improvements promise to spread the use of face recognition in future.

In machine learning, pattern recognition can be defined as the assignment of a label to a given input value. In pattern recognition we attempt to assign each input value to one of the given set of classes. However, pattern recognition is a more general problem and face recognition can be termed as a subset of pattern recognition. The most convenient way to do this is by comparing selected facial features from the image and a facial database. The face recognition problem has three main stages: face detection, recognition (or identification) and matching (or authentication). The detection stage is the first stage which includes identifying and locating the face in an image. The recognition stage is the second stage. It includes feature extraction, which is the most important part because here important information for discrimination is saved. Last is matching, where the recognition result is given with the help of a face database.

Over the last ten years or so, face recognition has become a popular area of research in computer vision and one of the most successful applications of image analysis and understanding. Because of the nature of the problem, not only computer science researchers, but neuroscientists and psychologists are also interested in it. It is believed that advances in computer vision research will provide useful insights to neuroscientists and psychologists into how human brain works, and vice versa.

The most popular face recognition algorithms include Principal Component Analysis using eigenfaces, Linear Discriminate Analysis (LDA), Elastic Bunch Graph Matching using the Fisherface algorithm, the Hidden Markov Model (HMM), and the neuronal motivated dynamic link matching. Some facial recognition algorithms identify faces by extracting landmarks, or features, from an image of the subject's face.

1.2 Various approaches for face recognition

Face recognition techniques can be classified in many ways. One possible classification at high-level is:

1.2.1 Holistic approach

This approach involves matching of faces on the basis of entire face. This makes it sensitive to face alignment. Global face measures are taken into account; hence they treat face differences that are due to different facial traits and non neutral expressions.

1.2.2 Region based approach

These methods are more effective than holistic methods. They work on the principle of applying different processing methods to distinct face regions and hence filter out those regions that are mostly affected by expression changes or spurious elements. In other words they partition the face surface into regions and extract appropriate descriptors for each of them. But, these are also sensitive to face alignment which makes useful face regions hard to detect automatically. Also, local features and differences in resolution determine their performance.

1.2.3 Hybrid and multimodal approach

These methods involve greater architectural complexity and hence provide highest accuracy. In hybrid approaches, different approaches like holistic or region based are combined or are used to perform matching separately and then their results are fused. They are mainly used for verification.

1.3 Applications of face recognition

There are innumerable applications of face recognition of which some have been mentioned above. There is potential for many more as soon as the problems due to pose variations, age variations and lighting variations etc. are overcome.

1.3.1 Access Control

These days password protection is not enough and has increasingly become unreliable and unpredictable. An access control system based on **face recognition technology** eliminates the need of the primitive security methods of implying security guards or maintaining a heap of papers containing the access details. This electronic security system captures the facial details of the individual and saves the data electronically into the database of the computer. When the individual revisits the premise for the next time, the face recognition system again captures the facial details and performs a match with the already stored patterns. When it finds a match, it gives the permission for access right. This is an easy method of checking the access of the individuals to certain restricted areas or resources.

This access system has many unique features like it ensures high security and is much more effective than the primitive methods. Also, there is no need of human touch.

1.3.2 Identification Systems

There may be many identification tasks, like sometimes a new applicant being enrolled must be compared against the entire database of previously enrolled claimants, to ensure that the new person is not claiming under more than one identity. In this case face recognition is most optimum than any other method because it is more acceptable and less intrusive. For example, face recognition can be employed to ensure that that people do not obtain multiple driving licenses.

1.3.3 Surveillance

Surveillance is the application domain where most of the interest is being shown in face recognition. It can be applied without the person's active participation, and indeed without his knowledge. Automated face recognition system can be applied 'live' to search for persons in the watch list or after using surveillance footage of a crime to search through the database of suspects.

1.3.4 Pervasive Computing

Pervasive or ubiquitous computing is another domain where face recognition is expected to become very useful, although it is still not commercially feasible. Computing devices, these days are already equipped with sensors of various kinds. These are found in our cars, phones and in many appliances in our homes. Networking of these devices is also becoming possible these days. We can imagine a future where many everyday gadgets will have some computational power, allowing them to change their behavior with time, user, user control and many other factors. Standard protocols are being developed which permit such devices to communicate with one another *e.g.* Bluetooth, ZigBee etc.

Most devices today can be controlled only by commands by the user. Some devices can also sense the environment, but it will be increasingly interesting and useful for such a pervasive, networked computing device to know about the physical world and the people within its region of interest. One of the most important parts of human awareness is knowing the identity of the user close to the device, and for this purpose, face recognition is most appropriate because of its passive nature.

Many examples can be found of pervasive face recognition tasks. Some devices such as Personal Digital Assistants (PDAs) already contain cameras for other purposes and in good illumination conditions can be used to identify their users. Similarly, a domestic message centre may have user personalization that depends on identification by a built in camera.

These kinds of devices when developed fully will not only make our lives easier but also prevent theft and intrusion of any kind by unwanted persons.

1.4 Challenges in face recognition

There are many variations in the facial images that are considered for face recognition. These are due to changes in illumination conditions, viewing direction, facial expression and aging etc.

1.4.1 Illumination variations

The illumination problem is illustrated in **Figure 1.1**, 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.

Figure 1.1: Variation in facial images due to illumination

In the above figure, we see that due to different lighting conditions the same face appears different. This poses a problem for face recognition systems.

1.4.2 Pose variations

The pose problem can be simple when the angle of rotation is small or it can be more difficult when there are large pose variations along with illumination variations. The pose problem is illustrated in the following figure.

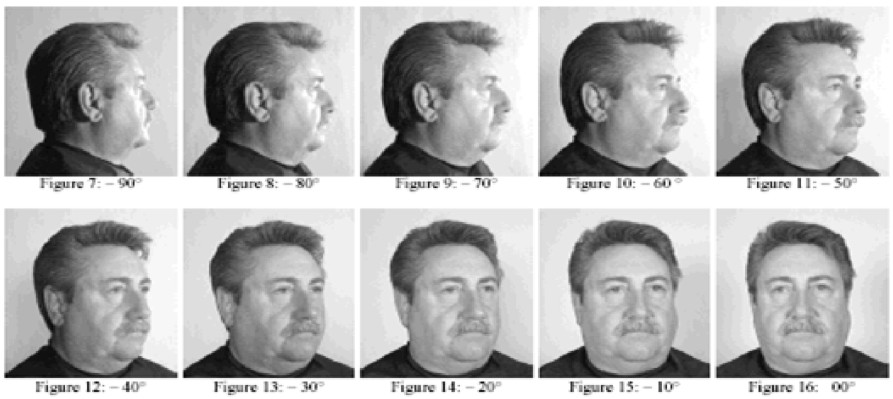


Figure 1.2: Variation in facial image due to different viewing angles

In the above figure, the same face appears differently due to changes in viewing condition. This makes face recognition difficult.

1.4.3 Age variations

Apart from illumination conditions and pose variations, age is another challenge in face recognition. There are many significant changes in human face as a person grows older. The features of the face vary for every person and are affected by many factors such as exposure to sunlight, inherent genetics, and nutrition. The color of the skin might also change with age. The performance of face recognition systems cannot cope with the dynamics of temporal metamorphosis over a period of time. Figure 1.3 shows the variations in face due to age.

Figure 1.3: Changes in face due to age

In the above figure, we see that the variations in face as the age progresses are large. This makes identification of the face difficult.

Apart from the above mentioned challenges, there are other factors also. The face images have similar geometrical features which make discriminating one face from the other in the database a challenging task. All these factors make it difficult to represent face images with distinct feature vectors that are invariant to transformation. The extracted feature vectors may possess overlapping characteristics, but the problem may be easily solved if there exists a feature extraction method which can generate distinct features for each class of images or a classification technique capable of discriminating the overlapping features of the images.

1.5 Scope of the work

In this work, we have focused on the problem of face recognition and have derived a novel way for recognizing faces using moments. For the purpose of feature extraction we have used four methods, Hu moments, Zernike moments, Legendre moments and cumulants. For the first three moments, their value for different orders is calculated. Feature vectors are formed using these values. For cumulants, first the bispectrum is calculated and then wavelets are applied on it for dimension reduction. After applying wavelets we get an envelope which can be classified.

For the purpose of classification, three approaches have been used: Minimum Distance Classifier, Support Vector Machine (SVM) and K Nearest Neighbor (KNN). Comparison has been done between all the three feature extraction methods using these classifiers.

For all the experiments conducted the ORL (Olivetty Research Laboratories) database has been used.

1.6 Organization of the thesis

The remainder part of this thesis is organized in the following chapters:

Chapter 2: Literature survey

In this section, the critical points of current knowledge on face recognition have been reviewed. Substantive findings as well as theoretical and methodological contributions in the field of face recognition have been included. This section summarizes the work done in the field of face recognition using various techniques in recent times.

Chapter 3: Proposed face recognition system

In this section, image preprocessing has been discussed. Also, detailed description about the methods of feature extraction used has been provided. These methods include Hu moments, Zernike moments, Legendre moments and cumulants. Along with this, the methods of classification also have been discussed. These include, minimum distance classifier, support vector machine, K nearest neighbor etc.

Chapter 4: Experiments and results

In this section, initially the dataset used for the work has been mentioned. Thereafter, all the experiments conducted along with the analysis of the results have been explained in detailed. The section is divided into two parts, feature extraction and classification. Graphs, figures and tables have been provided wherever necessary.

Chapter 5: Conclusion and Future Scope

In this section the conclusion of the thesis work and the future scope of the work are presented.

References: This section gives the reference details of the thesis.

Appendix A: Abbreviations

Appendix B: Introduction to Image Processing in MATLAB

Chapter 2

LITERATURE SURVEY

Face recognition has become a most sought after area of research in computer vision since the last ten years. Many successful applications of image analysis have been developed during this time. For this purpose different algorithms have been used.

A lot of work has been done in this field using Linear Discriminant Analysis. In [5], the author has proposed a novel Bayesian logistic discriminant (BLD) model which addresses normality and heteroscedasticity (problem in which the LDA algorithm assumes the sample vectors of each class which are generated from underlying multivariate normal distributions of common covariance matrix with different means). A subclass and multinomial versions of the BLD has been proposed. The posterior distribution of the BLD model is suitably approximated by a tractable Gaussian form using variational transformation and Jensen's inequality. This allows straightforward computation of the weights. But, BLD cannot solve the problem posed by nonlinearly separable data classes.

Chao-Kuei Hsieh, Shang-Hong Lai and Yung-Chang Chen [8], proposed face recognition using an optical flow based approach. A single 2-D face image with facial expression is used. The training database contains only neutral face images with one neutral face image per subject. Information from the computed intrapersonal optical flow and the synthesized face image are combined in a probabilistic framework for face recognition. However, the proposed integrated system is more computationally costly compared to the previous works, since the optical flow computation, image synthesis, and the probability calculations are needed for all candidates in the database.

Color Space Normalization to enhance the Discriminating Power of Color Spaces for Face Recognition was used in [9]. Here the authors explain the concept of color space normalization (CSN) and two CSN techniques for enhancing the discriminating power of color spaces for face recognition.

In [10], the author has combined Gabor features within the scope of diffusion-distance calculation. This strategy starts from the Gabor filtering that consists of three scales and six orientations. It is followed by the calculation of diffusion distance based on a Bayesian model. Bayesian model was used to determine the similarity degree between two histograms. The rationale of using this Bayesian decision model is to ensure the maximization of likely estimation across different histograms. There are various shortcomings of this method. For example, the recognition rate of the proposed algorithm reduces while handling the occlusions due to dramatical pose changes.

Another application which used Gabor filters was developed by Zhen Lei, Shengcai Liao, Matti Pietikäinen and Stan Z. Li [16], they proposed a face representation and recognition approach by exploring information jointly in image space, scale and orientation domains. First, the face image is first decomposed into different scale and orientation responses by convolving multi scale and multi orientation Gabor filters. Thereafter, local binary pattern analysis is used to describe the neighboring relationship in image space and in different scale and orientation responses. Then, discriminant classification is performed which is based upon weighted histogram intersection or conditional mutual information with linear discriminant analysis techniques.

Gabor features were again used by Shufu Xie, Shiguang Shan, Xilin Chen and Jie Chen in [22], for face recognition. Here, Local Gabor XOR patterns (LGXP) are proposed initially, which encode the Gabor phase by using the local XOR pattern (LXP) operator. Then, block-based Fisher's linear discriminants (BFLD) are introduced to reduce the dimensionality of the proposed descriptor and at the same time enhance its discriminative power. Finally, by using BFLD, local patterns of Gabor magnitude and phase are fused for face recognition.

Another application using Gabor representation on images was worked on by Christian Tenllado, Jose Ignacio Gomez, Javier Setoain, Dario Mora and Manuel Prieto [24]. They used Gabor representation of the images which is a holistic method for face recognition. A combination scheme by fusing the recognition scores obtained from natural face images and their Gabor representations has been proposed.

In [23], Zhiming Liu and Chengjun Liu present a novel face recognition method by means of fusing color, local spatial and global frequency information. Specifically, the proposed method fuses the multiple features derived from a hybrid color space, the Gabor image representation, the local binary patterns (LBP) and the discrete cosine transform (DCT) of the input image. The RCrQ color space is constructed by combining the R component image of the RGB color space and the chromatic component images, Cr and Q, of the YCbCr and YIQ color spaces, respectively. Three image encoding methods are then proposed for the component images in the RCrQ hybrid color space to extract features: (i) a patch- based Gabor image representation for the R component image, (ii) a multi-resolution LBP feature fusion scheme for the Cr component image, and (iii) a component based DCT multiple face encoding for the Q component image. Finally, at the decision level, the similarity matrices generated using the three component images in the RCrQ hybrid color space are fused using a weighted sum rule.

Yin Zhang And Zhi-Hua Zhou in [11], discussed cost sensitive face recognition. They proposed that in almost all application scenarios of face recognition the losses of all the misclassifications are not the same, i.e. different kinds of mistakes will lead to different losses. The face recognition problem is formulated as a multiclass cost sensitive learning task where the author tries to minimize the total cost. The costs of different kinds of misclassifications are given by the users according to their requirement. Two new cost-sensitive Methods have been proposed, mcKLR and mckNN, for face identification problems. mcKLR is an inductive learning method derived from Bayes decision theory, while mckNN is a cost-sensitive version of k-nearest neighbor classifier. mckNN is particularly suitable for situations where group members vary frequently, while mcKLR is more suitable for situations with stable group members.

Weiwen Zou and Pong C. Yuen [12], proposed multi-image face recognition, instead of using a single still-image-based approach in order to handle complex face image variations in face recognition. Two new measurements, namely discriminability index (DI) and reliability index (RI), have been proposed to evaluate the enrolled and query images, respectively. By considering the distribution of enrolled images from individuals, the discriminability index of each image is calculated and a weight is assigned. For testing images, a reliability index is calculated based on matching quality between the testing images and enrolled images. If the reliability index of a testing image is small, the testing image will be discarded as it may degrade the recognition performance. To evaluate and demonstrate the use of DI and RI, the combining classifier method with eigenface representations in input and kernel feature spaces has been adopted.

Work has also been done to overcome the challenges of face recognition like illumination variations in the images. Xiaoyang Tan and Bill Triggs [14] focused mainly on this issue i.e. robustness to lighting variations. For this purpose, the author has combined robust illumination normalization, local texture-based face representations, distance transform based matching, kernel based feature extraction and multiple feature fusion. Face recognition is based on local ternary patterns (LTP), a generalization of the local binary pattern (LBP). Kernel principal component analysis (PCA) has been used for feature extraction.

In [15], the authors have investigated methods that leverage multiple features extracted from the component images in a new color space, namely, the color image discriminant (CID) color space. The CID color space, thus, defines three component images, which are optimal with respect to a discriminant criterion. As the three color component images display different image characteristics and discriminating capabilities, three different methods are applied to extract the features from them, respectively. Further, the similarity scores due to the three color component images are fused for the final decision making.

Richard M. Jiang, Danny Crookes and Nie Luo [17], concentrated on feature selection process. In the proposed scheme, global harmonic features instead of disconnected pixels are used, which represent information about 2-D spatial structures. A nonlinear subspace approach, Laplacian Eigenmap, is then applied to perform subspace analysis based on global harmonic features.

Baochang Zhang, Yongsheng Gao, Sanqiang Zhao and Jianzhuang Liu in [18] worked on high-order local pattern descriptor, local derivative pattern (LDP), for face recognition. LDP is a general framework to encode directional pattern features based on local derivative variations. The n^{th} order LDP is proposed to encode the $(n-1)^{th}$ order local derivative direction variations. Different from LBP encoding the relationship between the central point and its neighbors, the LDP templates extract high-order local information by encoding various distinctive spatial relationships contained in a given local region.

Gui-Fu Lu, Zhonglin and Zhongjin [19], proposed a discriminant locality preserving projections based on maximum margin criterion (DLPP/MMC). DLPP/MMC seeks to maximize the difference, rather than the ratio, between the locality preserving between class scatter and locality preserving within class scatter. DLPP/MMC is theoretically elegant and can derive its discriminant vectors from both the range of the locality preserving between class scatter and the range space of locality preserving within class scatter. DLPP/MMC can also derive its discriminant vectors from the null space of locality preserving within class scatter when the parameter of DLPP/MMC approaches $+\infty$.

In the paper by Yong Wangn and Yiwu [20], the proposed method called intrinsic face model is based on the analysis of the image differences and the information conveyed by each face image. In this model, each face image is divided into three components corresponding to different properties of the face image. That is, facial commonness difference corresponding to the common nature of all the face images, individuality difference discriminates different individuals, and intrapersonal difference corresponding to the different facial expressions, illuminations and view changes of the same individual. Then, supervised dimensionality reduction algorithm called Intrinsic Discriminant Analysis (IDA) classifies different face images by maximizing the individuality difference, while minimizing the intrapersonal difference. This method is a little rough and can be improved.

Discrete cosine transform (DCT) is one of the most commonly used methods for face recognition. It has been used to extract proper features for face recognition. In [21], Saeed Dabbaghchian, Masoumeh P. Ghaemmaghami and Aliaghagolzadeh proposed a new category of coefficient selection approach in the Discrete Cosine Transform domain for face recognition. The approach called Discrimination power analysis (DPA) is a statistical analysis based on the DCT coefficients properties and discrimination concept. Coefficients which have more power to discriminate different classes better than others are searched. The proposed approach is data-dependent and finds DCs on each database.

In [25], Miao Cheng, Bin Fang, Yuan Yan Tang, Taiping Zhang and Jing Wen devise a supervised learning method, called local discriminant subspace embedding (LDSE), to extract discriminative features for face recognition. Thereafter, an incremental-mode algorithm, incremental LDSE (ILDSE), is proposed to learn the local discriminant subspace with the newly inserted data. It applies incremental learning extension to the batch LDSE algorithm by employing the idea of singular value decomposition (SVD) updating algorithm.

Imran Naseem, Roberto Togneri and Mohammed Bennamoun [27], in their research, proposed a linear Regression based classification (LRC) for the problem of face identification. Samples from a specific object class are known to lie on a linear subspace. This concept is used to develop class specific models of the registered users by simply using the downsampled gallery images, thereby defining the task of face recognition as a problem of linear regression. Least-squares estimation is used to estimate the vectors of parameters for a given probe against all class models. Finally, the decision rules in favor of the class with the most precise estimation. The proposed classifier is categorized as a Nearest Subspace (NS) approach.

Wen-Chung Kao, Ming-Chaihsu and Yueh-Yiing Yang [28] concentrate on recognizing human faces in various lighting conditions. An integrated human face recognition system is proposed that first compensates uneven illumination through local contrast enhancement. Then the enhanced images are fed into a face recognition system which adaptively selects the most important features among all candidate features and performs classification by support vector machines (SVMs). The dimension of feature space along with the selected types of features is customized for each hyperplane.

Shuicheng Yan, Huan Wang, Jianzhuang Liu, Xiaoou Tang and Thomas S. Huang [29], in their paper worked on providing solution to the face recognition problem under the scenarios with spatial misalignments and/or image occlusions. A comprehensive system was presented by Ritwik Kumar, Angelos Barmpoutis, Arunava Banerjee and Baba C. Vemuri [30], for capturing the reflectance properties and shape of the human faces using Tensor Splines. But the method requires at least nine input images with known illumination directions. Also, accurate recovery of Apparent Bidirectional Reflectance Distribution functions ABRDF field from a single image with cast shadows and specularities with no lighting information remains a challenge.

In [31], Tae-Kyun Kim, Josef Kittler and Roberto Cipolla have addressed the problem of face recognition by matching image sets. Each set of face images is represented by a subspace (or linear manifold) and recognition is carried out by subspace to subspace matching. Jiwen Lu and Yap-Peng Tan propose in their paper [33] a parametric regularized locality preserving projections (LPP) method for face recognition. The proposed method is designed to regulate LPP features corresponding to small and zero eigen values, i.e., the noise space and null space, and exploit the discriminant information from the whole space.

In [35], a semi-supervised dimensionality reduction method called sparsity preserving discriminant analysis (SPDA) was developed by Lishan Qiao, Songcan Chen and Xiaoyang Tan. This algorithm models the ‘‘locality” and improves the performance of typical LDA. It can be applied to face recognition problem with only a few training samples. Sang-Ki Kim, Youn Jung Park, Kar-Ann Toh and Sangyoun Lee [36], proposed a feature extraction algorithm called SVM-based discriminant analysis (SVM-DA). Through his approach, the author aims at overcoming the limitations of LDA. He suggests that by implanting an SVM margin to the framework of LDA, the feature extraction can be made applicable to heteroscedastic data while alleviating the small sample size (SSS) and the dimensionality problems.

Bidirectional principal component analysis (BDPCA) has been used for face recognition in [37] by Chuan-Xian Ren and Dao-Qing Dai. It attempts to solve the small sample size problem of PCA. The main idea behind BDPCA is to find the optimal projection subspaces in the row and column directions of images without the matrix to vector transformation. That is, it first constructs the image covariance matrices from rows and columns of images and then computes their eigen vectors as the projection vectors. However, BDPCA has to be performed in batch mode; it means that all the training data has to be ready before we calculate the projection matrices. If there are additional samples need to be incorporated into an existing system, it has to be retrained with the whole updated training set. Moreover, the scatter matrices of BDPCA are formulated as the sum of K (samples size) image covariance matrices, this leads to the incremental learning directly on the scatters impossible.

In [38], Yong Wang and YiWu used Complete Neighborhood preserving embedding (CNPE) for the purpose of face recognition. CNPE aims at removing the singularity problem of eigen matrix suffered by Neighborhood preserving embedding (NPE). CNPE transforms the singular generalized eigen system computation of NPE into two eigen value decomposition problems.

All the works mentioned above have used 2D face recognition methods. 3D face recognition methods have also emerged significantly in the past few years. Since then many authors have chosen to work with 3D images for face recognition. In [1], Stefano Berretti, Alberto Del Bimbo and Pietro Pala have used Isogeodesic strips for 3D face recognition. A graph is made using the relevant geometrical information from the 3D image. This information is encoded into a compact representation. There are arcs between pair of nodes which are labeled with descriptors. These descriptors are called 3D Weighted Walkthroughs (3DWWs). They capture the mutual relative spatial displacement between all the pairs of points of the corresponding stripes. Face partitioning into isogeodesic stripes and 3DWWs together provide an approximate representation of local morphology of faces. This exhibits smooth variations for changes induced by facial expressions.

In [2], Chaua C. Queirolo, Luciano Silva, Olga R.P. Bellon, and Maurı´Cio Pamplona Segundo worked on Simulated Annealing-based approach (SA) to perform 3D face recognition. They also used Surface Interpenetration Measure (SIM), as similarity measure, for matching two 3D face images. The acquired input image is segmented into four different face regions: circular and elliptical areas around the nose, forehead, and the entire face region. These regions of corresponding faces are matched and SIM values are combined to obtain an authentication score. A modified SA approach is proposed for better handling of facial expressions.

Ira Kemelmacher-Shlizerman and Ronen Basri in [3], used as input a single image and used a reference 3D model to reconstruct the desired 3D shape. The fact that there is a global similarity in all the faces and the main features like eyes, nose etc are located at more or less at the same places has been exploited. The method uses Lambertian reflectance and harmonic representations of lighting.

In the work done by R. Salalloncha, E. Kokiopouloub, I. Tosic and P. Frossard [4], face recognition is performed using simultaneous sparse approximations on the sphere. The 3D face point clouds are first registered using an algorithm based on an average face model (AFM) and on the iterative closest point (ICP) algorithm. After that, the faces are converted from point clouds to spherical signals. This makes dimensionality recognition easier which is done through simultaneous sparse approximations. This is done in order to preserve geometrical information. Finally, recognition is performed by effectively matching in the reduced space.

In [6], Antonio Rama, Francesc Tarres and Jurgen Rurainsky have used an approach where only part of the information (partial concept) is used in recognition. It is called Partial Principal Component Analysis (P2CA) where Partial concept is fused with the fundamentals of the well known PCA algorithm. Here 3D data is used in the training but either 2D or 3D information can be used in the recognition. An approach for automatic creation of 180° aligned cylindrical projected face images is presented which uses nine different views. These facial images are created using a cylindrical approximation for the actual object surface. The alignment is done by applying first a global 2D affine transformation of the image and thereafter a local transformation of the desired face features using a triangle mesh. Finally, these aligned face images are used for training a face recognition approach (P2CA) which is pose invariant.

In their work [32], Nes¸E Alyüz, Berk Gokberk and Lale Akarun, proposed an expression insensitive 3-D face recognition system. Region based registration scheme that uses common region models has been proposed which makes it possible to establish correspondence to all the gallery samples in a single registration pass. Curvature based 3-D shape descriptors are utilized and statistical feature extraction methods are applied.

Yueming Wang, Jianzhuang Liu and Xiaoou Tang in [34] proposed a 3D face recognition approach called Collective Shape Difference Classifier (CSDC). First a fast posture alignment method is presented which avoids the registration between an input face against every face in the gallery. Then, a Signed Shape Difference Map (SSDM) is made between two aligned 3D faces as a mediate representation for the shape comparison. Based on the SSDMs, three kinds of features are used to encode both local similarity and the change characteristics within facial shapes. The local features which are most discriminative are selected optimally by boosting and trained as weak classifiers for assembling three collective strong classifiers, namely, CSDCs with respect to the three kinds of features. Although this method works well on common faces of approximate mirror-symmetry with a nose, it can fail when too many data points of the nose are missing, which causes incorrect alignment. This is the major limitation of this method.

Three dimensional face recognition systems have many disadvantages. Firstly, their computation is expensive. Although the equipments required to capture 3D images are becoming widely available and their cost is also reducing day by day, their price is still significantly high as compared to a simple high resolution camera. Along with this, the processing the 3D images is also very expensive.

Secondly, the main reason for the effectiveness of the 3D face recognition system because of its capability to achieve invariance to face expressions. On the other hand, while the 3D images are insensitive to the illumination conditions and variation in poses, they are even more sensitive to the face expressions compared to the 2D images.

Thirdly, illumination variations do not cause any problems while processing the 3D data, but it is a problem while capturing it. The oily parts of the face have high reflectance which may cause some problem under certain kinds of light conditions, although it depends on the technology of the sensor used. Therefore the quality of the images finally obtained for the database cannot be relied upon totally as compared to the 2D images.

Finally, one of the most important disadvantages of 3D face recognition is that capturing of 3D images requires cooperation from the subject. The lenses and the scanners used for capturing 3D images require that the subject should be at a certain distance from the camera. Moreover, complete immobility for a few seconds is required by the laser scanner, while a conventional camera can capture images from far away without any cooperation from the subjects. This is the reason that currently there are very few high-quality 3D face databases available for testing and evaluation purposes. The ones that are available are very small as compared to the 2D facial databases.

Chapter 3

PROPOSED FACE RECOGNITION SYSTEM

The entire face recognition system used in this work **c**onsists of three steps:

1) Image preprocessing
2) Feature extraction
3) Classification

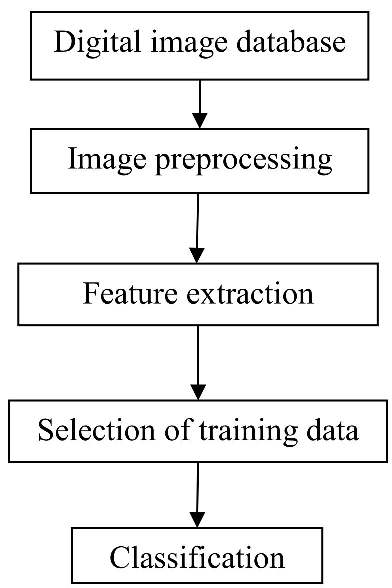


Figure 3.1: Proposed face recognition system

An image is extracted from database and the required preprocessing is done on it. Preprocessing helps in improving the results obtained from the feature extraction methods. Thereafter, features are extracted using the appropriate methods and feature vectors are formed. These feature vectors are then classified using various classifiers. All the above mentioned steps have been elaborated in this chapter.

3.1 Image preprocessing

Image preprocessing commonly comprises of a series of sequential operations including image enhancement or normalization, geometric correction, removal of noise etc. It is done before applying the feature extraction techniques. Using the above mentioned techniques, we alter the image pixel values permanently and use this improved data for later analyses.

3.1.1 Need for preprocessing

Image preprocessing more importantly is done for noise removal which is mainly added during image acquisition. Image acquisition is a very important step for the quality control since it provides the input data for the whole process. Digital image is acquired by an optical sensor which is always a video camera (with one line or a matrix of CCD). It provides accurate and noiseless image. Local illumination is also directly linked with the quality of image acquisition.

Sometimes there might be other reasons also. There may be a need to convert a colored image into grayscale. Image enhancement by various methods also comes under image preprocessing.

3.1.2 Image preprocessing methods commonly used

i) Noise removal

Images taken with both digital cameras and conventional film cameras pick up noise. It is the result of errors in the image acquisition process that result in pixel values that do not reflect the true intensities of the real scene from a variety of sources. For further use of these images, it is required that the noise should be removed. For removal of noise various filters can be applied on images, like high pass filter, Gaussian filter etc.

ii) Contrast adjustment

The contrast of an image refers to the distribution of the dark and light pixels. A low-contrast image means there is a small difference between its light and dark pixel values. This in turn means that the histogram of a low-contrast image is narrow. Since the human eye is sensitive to contrast rather than absolute pixel intensities, a better image can be obtained by stretching the histogram of the image so that the full dynamic range of the image is used.

iii) Intensity adjustment

Image enhancement techniques are used to improve an image. These techniques can be used to increase the signal-to-noise ratio or make certain features easier to be seen by modifying the colors or intensities. Intensity adjustment is an image enhancement technique which maps the image intensity values to a new range.

iv) Histogram Equalization

Histogram equalization distributes evenly the occurrence of pixel intensities so that the entire range of intensities is occupied. This method usually increases the contrast of images globally, especially when the required data of the image is represented by close contrast values. By adjusting the histogram, the intensities can be better distributed. Due to this, the areas of lower local contrast gain a higher contrast. Histogram equalization effectively spreads out the most frequent intensity values.

v) **Normalisation**

Image normalization is done in order to reduce the amount of computation so that the method is more efficient. The pixel values ranging from 1 to 255 are mapped to lie within 0 to 1.

3.2 Feature extraction

In image processing, feature extraction is a special form of dimensionality reduction. It involves simplifying the amount of resources required to describe a large set of data accurately. Image features can refer to the global properties of an image like average gray level, shape of intensity histogram etc. or the local properties of the image like shape of contours, elements composing textured region etc.

The issue of choosing the features to be extracted is critical and the following concerns should be kept in mind while selecting them:

1) Enough information about the image should be carried by the features and there should be no requirement of any domain specific knowledge for their extraction.
2) Their computation should be easy so that the approach is feasible for a large image database and rapid retrieval.
3) They should be meaningful i.e. they should be associated with interesting elements in the image formation process. They should be invariant to the image formation process like invariant to the viewpoint and illumination of the captured digital images.
4) They should relate well with the human perceptual characteristics since the end users will finally determine the suitability of the images retrieved.
5) The extracted features should be able to be detected or located from the images with the help of algorithms. Also they should be easily described by a feature vector.

In this work, the features are extracted from images using four methods: Hu moments, Zernike moments, Legendre moments and cumulants. The extracted features are then subjected to classification.

In image processing, computer vision and related fields, an image moment is a certain particular weighted average (moment) of the image pixels' intensities, or a function of such moments, usually chosen to have some attractive property or interpretation. These are useful to describe objects after segmentation. Simple properties of the image which are found via image moments include area (or total intensity), its centroid and information about its orientation. Moments and their functions have been utilized as pattern features in a number of applications. These features can provide global information about the image.

In the coming sections, all the above mentioned moments and how they can be applied to digital images has been described.

3.2.1 Hu moments

Hu moments have been derived from the geometric moments. Geometric moments are also known as regular or raw moments [44]. These are nonnegative integers, which can be computed by equation (3.1):

$$m_{pq} = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} x^p y^q f(x,y)dxdy \tag{3.1}$$

where p,q = 0,1,2,3……

In the above equation, m_{pq} is the moment of order (p+q) and f(x,y) is a two dimensional, real and continuous density function of which the moment has to be calculated. In our case, the density function will be the pixel values of the grayscale digital image which will take values from 0 to 255. Hence we need a discrete version of the above equation which can be represented as in equation (3.2) given below:

$$m_{pq} = \sum_{x=1}^{M}\sum_{y=1}^{N} x^p y^q f(x,y) \tag{3.2}$$

Here, $M \times N$ is the size of the image i.e. number of pixels in rows and columns respectively.

The central moment of order (p+q) of the image is defined as in equation (3.3):

$$\mu_{pq} = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} (x-\bar{x})^p (y-\bar{y})^q f(x,y)dxdy \tag{3.3}$$

Here, $\bar{x}$ and $\bar{y}$ are the centroids of the image and are defined by:

$$\bar{x} = \frac{m_{1,0}}{m_{0,0}} \quad \text{and} \quad \bar{y} = \frac{m_{0,1}}{m_{0,0}} \tag{3.4}$$

Now, if f(x, y) is the density function for a digital image, the equation for the central moment in discrete format becomes as shown in equation (3.5):

$$\mu_{pq} = \sum_{x=1}^{M}\sum_{y=1}^{N}(x-\bar{x})^{p}(y-\bar{y})^{q} f(x,y) \tag{3.5}$$

Here again, $M \times N$ is the size of the digital image.

Central moments can also be defined in terms of the raw moments as shown in equation (3.6):

$$\mu_{pq} = \sum_{i}^{p}\sum_{j}^{q}\binom{p}{i}\binom{q}{j}\left(-\bar{x}\right)^{(p-i)}\left(-\bar{y}\right)^{(q-j)} m_{ij} \tag{3.6}$$

The above defined central moments are origin independent and therefore they are translation invariant.

Moments of order two or more can be constructed to be invariant to both translation and scale. Scale invariance can be achieved by dividing the corresponding central moment by the scaled energy of the original i.e. the 00^{th} moment as in equation (3.7):

$$\eta_{ij} = \frac{\mu_{ij}}{\mu_{oo}^{\left(1+\frac{i+j}{2}\right)}} \tag{3.7}$$

Hu, in 1962 introduced 'Hu moments' which are invariant to translation, scale as well as rotation. Hu moments are a set of seven moments which are nonlinear combination of normalized central moments up to order three. These are represented as in equation (3.8):

$$
\begin{aligned}
M_1 &= \eta_{20} + \eta_{02} \\
M_2 &= (\eta_{20} - \eta_{20})^2 + 4\eta_{11}{}^2 \\
M_3 &= (\eta_{30} - 3\eta_{12})^2 + (\eta_{03} - 3\eta_{21})^2 \\
M_4 &= (\eta_{30} - \eta_{12})^2 + (\eta_{03} + \eta_{21})^2 \\
M_5 &= (3\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})\left[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2\right] \\
&\quad + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})\left[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2\right] \\
M_6 &= (\eta_{20} - \eta_{02})\left[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2\right] + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03}) \\
M_7 &= (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})\left[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2\right] \\
&\quad + (3\eta_{12} - \eta_{03})(\eta_{21} + \eta_{03})\left[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{30})^2\right]
\end{aligned}
\tag{3.8}
$$

The first one, M_1, is analogous to the moment of inertia around the image's centroid, where the pixels' intensities are analogous to physical density. The last one, M_7, is skew invariant, which enables it to distinguish mirror images of otherwise identical images.

3.2.2 Zernike moments

Zernike in 1934 introduced a set of complex polynomials called the Zernike polynomials. These polynomials form a complete orthogonal basis set defined on the interior of the unit disc, i.e., $x^2 + y^2 = 1$. Let us denote the set of these polynomials by $V_{nm}(x,y)$. In polar coordinates, these are then represented as [45]:

$$V_{nm}(x,y) = V_{nm}(r,\theta) = R_{nm}(r)\exp(jm\theta) \tag{3.9}$$

Where:

n Positive integer or zero

m Positive and negative integers subject to constraints (n-$|m|$) is even and $|m| \leq n$

r Length of vector from origin to (x , y) pixel

θ Angle between vector ρ and x axis in counter clockwise direction

In the above equation, R_{mn} is the Radial polynomial. It is orthogonal and is defined as in equation (3.10):

$$R_{nm}\left(r\right)=\sum_{s=0}^{\left(n-|m|\right)/2}\frac{\left(-1\right)^{s}\left(n-s\right)!\,r^{n-2s}}{s!\left[\frac{n+|m|}{2}-s\right]!\left[\frac{n-|m|}{2}-s\right]!} \tag{3.10}$$

It is to be noted that $\mathrm{R}_{n-m}\left(\mathrm{r}\right) = \mathrm{R}_{nm}(\mathrm{r})$.

Complex Zernike moments are constructed using the set of Zernike polynomials. Zernike moments are the projection of the image density function on these orthogonal basis functions. The two dimensional Zernike moment for a continuous image function f (x, y) that disappears outside the unit circle is defined as in equation (3.11):

$$Z_{nm}=\frac{n+1}{\pi}\iint_{\text{unit disk}} f(x,y)V_{nm}^{*}\left(x,y\right)dxdy \tag{3.11}$$

Since in our case, the image is digital, so the integrals are replaced by summations to get:

$$A_{nm}=\frac{n+1}{\pi}\sum_{x}\sum_{y} f\left(x,y\right)V_{nm}^{*}\left(x,y\right) \tag{3.12}$$

Here, $x^2+y^2\leq 1$

To calculate the Zernike moments of a particular image, we first take the centre of image as the origin and map the region of interest to the range of unit disc. Those image pixels which do not fall inside the unit disc are not used for computation. Thereafter, the image coordinates are described in terms of the length of the vector from the origin, the angle from the axis to the vector etc. Polar coordinates can be mapped into Cartesian coordinates as shown in equation (3.13):

$$\begin{aligned} X &= r\cos\theta \\ Y &= r\sin\theta \end{aligned} \tag{3.13}$$

Rotational invariance is the inherent feature of the Zernike moments. If the image is rotated by an angle θ , then the Zernike moment of the rotated image is given by equation (3.14):

$$Z'_{nm} = Z_{nm} e(-jm\theta) \tag{3.14}$$

Here, Z_{nm} is the Zernike moment of the original image.

We can also achieve translation and scale invariance for which we need to normalize the image first by using the Cartesian moments. Translation invariance is achieved by shifting the origin of the image to the center of mass i.e. the original image defined by $f(x,y)$ is now defined as $f(x+\bar{x}, y+\bar{y})$ where:

$$\bar{x} = \frac{m_{10}}{m_{00}} \text{ and } \bar{y} = \frac{m_{01}}{m_{00}} \tag{3.15}$$

For scale invariance, we alter each object or shape in the image by enlarging or reducing it so that the 0th moment of the image m_{00} is equal to a predetermined value beta. For binary images, m_{00} is equal to the total number of shape pixels in the image. If we define the scaled image as $f(\alpha x, \alpha y)$, then the regular moments of the scaled image can be represented as $m'_{pq} = \alpha^{p+q+2} m_{pq}$, where m_{pq} is the regular moment of the original image.

3.2.3 Legendre Moments

Legendre moments were derived from Legendre polynomials as kernel function. Legendre polynomials were first proposed by Teague. These are orthogonal moments which can represent an image with minimum information redundancy. Thus the moments represent the independent characteristics of an image.

The two-dimensional Legendre moments of order (p+q), are defined as in equation (3.16):

$$L_{pq} = \frac{(2p+1)(2q+1)}{4} \int_{-1}^{1}\int_{-1}^{1} P_p(x) P_q(y) f(x,y) dxdy \tag{3.16}$$

where p,q = 0,1,2,3…….∞ and x,y $\in$ [-1,1].

P_p and P_q are Legendre polynomials and f(x, y) is the continuous image function. These Legendre polynomials define a complete orthogonal basis set over the interval [-1, 1]. In order to maintain orthogonality in the moments as well, the image function is also defined over the same interval i.e. [-1, 1]. The Legendre polynomial of order 'n' is defined as shown below:

$$P_n\left(x\right)=\sum_{k=0}^{n}\left\{\left(-1\right)^{\frac{n-k}{2}}\frac{1}{2^n}\frac{\left(n+k\right)!x^k}{\left(\frac{n-k}{2}\right)!\left(\frac{n+k}{2}\right)!k!}\right\} \tag{3.17}$$

In the above definition, (n-k) should be even. Now, the Legendre moments for a discrete image consisting of pixels $M\times N$ with intensity function f(i, j), are defined as:

$$L_{pq}=\lambda_{pq}\sum_{i=0}^{M-1}\sum_{j=0}^{N-1}P_p\left(x_i\right)P_q\left(y_j\right)f\left(i,j\right) \tag{3.18}$$

where λ_{pq} is called the normalizing constant and is defined as in equation (3.19):

$$\lambda_{pq}=\frac{\left(2p+1\right)\left(2q+1\right)}{MN} \tag{3.19}$$

Where, $P_0\left(x\right)=1$ and $P_1\left(x\right)=x$.

The image is scaled in the region [-1, 1] since the definition of Legendre polynomials exists in this region only. So, x_i and y_j can be normalized as shown below in equation (3.20):

$$x_i=\frac{2i}{\left(M-1\right)}-1 \quad \text{and} \quad y_j=\frac{2j}{\left(N-1\right)}-1 \tag{3.20}$$

3.2.4 Cumulants

Cumulants are quantities that are similar to moments and can be used to extract the inherent features of images. In this paper, we have decided to use cumulants as one of the measures for face recognition, because cumulants can extract features which are otherwise very difficult to extract.

Simple moments are used to derieve cumulants. The rth moment of a real valued continuous random variable *X* with probability density function *f(x)* can be defined as in equation (3.21):

$$\mu_r = E\left(X^r\right) = \int_{-\infty}^{\infty} x^r f\left(x\right)dx \tag{3.21}$$

where *r* is a finite integer i.e. *r* = 0,1, ...

All the moments can be represented with a single expression if the moment generating function has a Taylor expansion about the origin which is defined as in equation (3.22):

$$M\left(\xi\right) = E\left(e^{\xi X}\right) = E\left(1 + \xi X + ... + \xi^r X^r / r! + ...\right) = \sum_{r=0}^{\infty} \mu_r \xi^r / r! \tag{3.22}$$

The r^{th} derivative is the r^{th} moment of *M* at the origin.

Similarly, the cumulants represented as κ_r are the coefficients of the Taylor expansion of the cumulants generating function about the origin defined as in equation (3.23):

$$K\left(\xi\right) = \log M\left(\xi\right) = \sum_r \kappa_r \xi^r / r! \tag{3.23}$$

The second, third and fourth-order cumulants of a zero-mean stationary process *u* (n) are defined as:

$$\kappa_2\left(\tau\right) = E\left\{u\left(n\right)u\left(n+\tau\right)\right\}$$

$$\kappa_3\left(\tau_1,\tau_2\right) = E\left\{u\left(n\right)u\left(n+\tau_1\right)u\left(n+\tau_2\right)\right\}$$

$$\kappa_4(\tau_1,\tau_2,\tau_3) = E\{u(n)u(n+\tau_1)u(n+\tau_2)u(n+\tau_3)\} - E\{u(n)u(n+\tau_1)\}E\{u(n+\tau_2)u(n+\tau_3)\}$$
$$-E\{u(n)u(n+\tau_2)\}E\{u(n+\tau_3)u(n+\tau_1)\} - E\{u(n)u(n+\tau_3)\}E\{u(n+\tau_1)u(n+\tau_2)\} \quad (3.24)$$

Cumulants can be represented in terms of the moments also. The first few cumulants in terms of the moments are:

$$\kappa_1 = \mu_1$$

$$\kappa_2 = \mu_2 - \mu_1^2$$

$$\kappa_3 = \mu_3 - 3\mu_2\mu_1 + 2\mu_1^3$$

$$\kappa_4 = \mu_4 - 4\mu_3\mu_1 - 3\mu_2^2 + 12\mu_2\mu_1^2 - 6\mu_1^4 \quad (3.25)$$

Inversely, we can write:

$$\mu_2 = \kappa_2 + \kappa_1^2$$

$$\mu_3 = \kappa_3 + 3\kappa_2\kappa_1 + \kappa_1^3$$

$$\mu_4 = \kappa_4 + 4\kappa_3\kappa_1 + 3\kappa_2^2 + 6\kappa_2\kappa_1^2 + \kappa_1^4 \quad (3.25)$$

Higher-order cumulants are not the same as moments about the mean.

The first-order cumulant of a stationary process is the mean of the process and the second order cumulant is its variance. The higher-order cumulants are equivalent to the central moments, they are invariant to the mean shift and hence defined under the assumption of zero mean.

The advantage of cumulants is that while the moment generating function of two random variables is the product of their individual moment generating functions, variables, the cumulants generating function is the sum. That is, if Let S = X + Y, then the moment generating function of ‘S’ is defined as in equation (3.27):

$$M_S(\xi) = M_X(\xi)M_Y(\xi) \quad (3.26)$$

On the contrary, the cumulant generating function is defined as:

$$K_S(\xi) = K_X(\xi) + K_Y(\xi) \tag{3.27}$$

It can also be now concluded that the rth cumulant of the sum is the sum of the respective rth cumulants.

i) Polyspectra

The kth order polyspectrum or the kth order cumulants spectrum is the Fourier transform of the corresponding cumulants and is defined as:

$$C_k(\omega_1, \omega_2, \ldots\ldots, \omega_{k-1}) = \sum_{\tau_1=-\infty}^{\infty} \ldots\ldots \sum_{\tau_{k-1}=-\infty}^{\infty} \kappa_k(\tau_1, \tau_2, \ldots\ldots, \tau_{k-1}) \times \exp\left[-j(\omega_1\tau_1 + \omega_2\tau_2 + \ldots\ldots + \omega_{k-1}\tau_{k-1})\right] \tag{0.28}$$

A polyspectrum exists when the corresponding kth order cumulants are absolutely summable. Special cases of the kth order polyspectrum are the power spectrum, bispectrum and the trispectrum [46].

1) For k = 2, we have the ordinary power spectrum defined by:

$$C_2(\omega_1) = \sum_{\tau_1=-\infty}^{\infty} \kappa_2(\tau_1) \exp(-j\omega_1\tau_1) \tag{0.29}$$

2) For k = 3, we have the bispectrum, defined by:

$$C_3(\omega_1, \omega_2) = \sum_{\tau_1=-\infty}^{\infty} \sum_{\tau_2=-\infty}^{\infty} \kappa_3(\tau_1, \tau_2) \exp\left[-j(\omega_1\tau_1 + \omega_2\tau_2)\right] \tag{0.30}$$

3) For k = 4, we have the trispectrum, defined by:

$$C_4(\omega_1, \omega_2, \omega_3) = \sum_{\tau_1=-\infty}^{\infty} \sum_{\tau_2=-\infty}^{\infty} \sum_{\tau_3=-\infty}^{\infty} \kappa_4(\tau_1, \tau_2, \tau_3) \exp\left[-j(\omega_1\tau_1 + \omega_2\tau_2 + \omega_3\tau_3)\right] \tag{0.31}$$

While the power spectrum is real-valued, bispectrum, trispectrum and other higher order polyspectra are complex valued.

ii) Properties of polyspectra

For a real valued process, symmetry properties of cumulants are carried forward to the symmetry properties of corresponding poly-spectra. The power spectrum is symmetric:

$$C_2(\omega_1) = C_2(-\omega_1) \tag{0.32}$$

The symmetry properties of the bi-spectrum are [42]:

$$C_3(\omega_1,\omega_2) = C_3(\omega_2,\omega_1) = C_3(\omega_1,-\omega_1,-\omega_2) = C_3(-\omega_1,-\omega_2,\omega_2) = C_3^*(-\omega_1,-\omega_2) \tag{0.33}$$

The symmetry properties of the tri-spectrum are:

$$C_4(\omega_1,\omega_2,\omega_3) = C_4(\omega_1,\omega_3,\omega_2) = C_4(\omega_2,\omega_1,\omega_3) = C_4(-\omega_1,\omega_2-\omega_1,\omega_3-\omega_1) = C_4^*(-\omega_1,-\omega_2,-\omega_3) \tag{0.34}$$

3.3 Classification

Image classification involves analyzing the numerical properties of various image features and organizing the data into categories. Usually, two phases of classification algorithms are employed: training and testing.

In the initial training phase, characteristic properties of typical image features are identified and based on these, a unique description of each classification category, i.e. training class, is created. The description of training classes is an extremely important component of the classification process.

The motivating criteria for constructing training classes are that they are:

i) independent, i.e. a change in the description of one training class should not change the value of another,

ii) discriminatory, i.e. different image features should have significantly different descriptions, and

iii) reliable, i.e. all image features within a training group should share the common definitive descriptions of that group.

In the next phase i.e. testing, accuracy of the classifier is measured. The accuracy can be determined by applying the classifier to an independent training set of objects with known classifications. Knowledge of the accuracy is necessary both in the application of the classifier and also in comparison of different classifiers.

A convenient way of building a parametric description of this sort is via a feature vector, where 'n' is the number of attributes which describe each image feature and training class. This representation allows us to consider each image feature as occupying a point, and each training class as occupying a sub-space (i.e. a representative point surrounded by some spread, or deviation), within the n-dimensional classification space. Viewed as such, the classification problem is that of determining to which sub-space class each feature vector belongs.

Classification can be linear or non linear. Linear classifiers are usually the fastest classifiers. It classifies the data on the basis of a linear combination of the characteristics based on which the classification is done. The classes are divided by a linear separator in the feature space. If the feature space is two dimensional, the separator is a line, it three dimensional the separator is a plane and if 'p' dimensional then the linear separator is a (p-1) dimensional hyper plane. Non linear classification is required when the class boundaries cannot be approximated well with linear hyper planes. Example of non linear classifier is K Nearest Neighbor.

3.3.1 Methods of classification

There are mainly two ways of classification, supervised and unsupervised. The difference between them lies in the fact that how the data is classified.

i) Supervised classification

In supervised classification, there are predetermined classes. Statistical processes (i.e. based on an a priori knowledge of probability distribution functions) or distribution free processes can be used to extract class descriptors. These classes can be regarded as a previously decided finite set. After classification, certain segment of data will be labeled with these classes. The task of the algorithm is to search for patterns and construct mathematical models. These models are then used to find out the measure of variance in the data and classify it. The examples of supervised classification are decision tree induction, naïve bayes classifier etc.

ii) Unsupervised classification

Unsupervised classification relies on clustering algorithms to automatically segment the training data into prototype classes. In this type of classification, the classes are not pre decided. The basic task of the classifier is to develop the classes or the labels automatically. The algorithm is not told how the data is to be grouped; it is something it has to arrive at by itself. This is a difficult decision to make. The classifier looks for similarities between data and then determines which of these can form a group and can be classified under one label. The classes are also called clusters. The example of unsupervised classification is K-means classification. In K-means, the classifier is told in advance, the no of clusters have to be formed.

In unsupervised classification, since there is no defined strategy, the algorithm starts from one point and performs iterative repetitions to reach a stable configuration that makes sense. The results can vary widely and depend largely on the first few steps taken.

3.3.2 Examples of classifiers

The most common classifiers used are:

i) Support vector machine (SVM)

Support vector machine (SVM) is an example of supervised learning classification. It is a binary linear classifier which takes an input and decides to which of the two classes it belongs. The classifier is first trained using a set of training examples. The training examples are pre marked as belonging to one of the two categories and based on these examples, the SVM classifier builds a model that assigns new examples (data to be tested) their suitable classes. The training examples are represented as points in space and are mapped such that there is a clear gap which divides the examples belonging to separate classes. The new examples are then mapped into the same space by analyzing to which of the two classes they suit better.

Apart from classification, a support vector machine can also be used for regression etc. It constructs a hyperplane that separates the two classes with gap as wide as possible. A hyperplane is regarded as 'good' if it has the largest distance with the nearest training data point of any class. This reduces the error of the classifier.

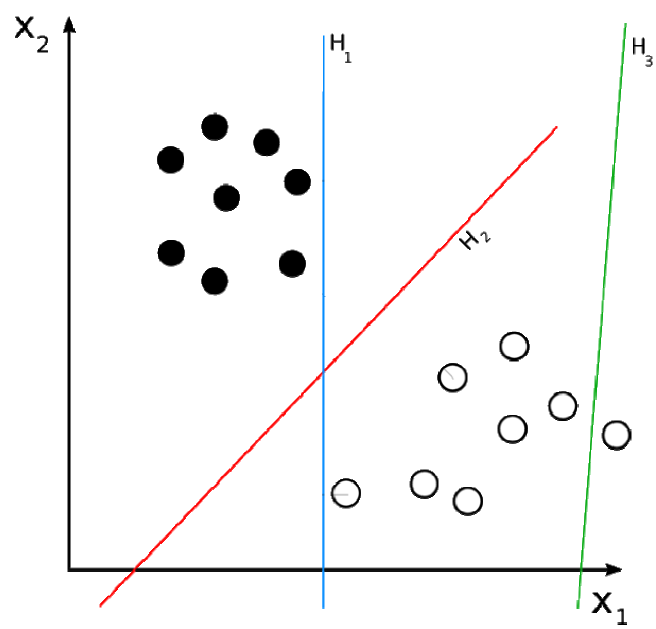


Figure 3.2: Hyperplanes separating two classes

In the above figure, we see that although red and blue hyperplanes both are separating the two classes entirely, the red hyperplane is doing it such that its distance from the nearest points of the two classes is maximum. Hence the red hyperplane is the most optimum hyperplane. The data points which are closest to the hyperplane are called support vectors.

Non linear SVM

Non linear SVM is applied in cases where the data sets have non linear decision boundaries. The trick applied here is to transform the data from its original coordinate space to a new space where a linear decision boundary can be used to separate the instances in the transformed space. Transformation of space may suffer from dimensionality problem which is associated with high dimensional data. Moreover, it is not always easy to find out what mapping must be used to ensure that a linear decision boundary can be constructed in the transformed space. This problem can be solved by 'Kernel' trick. It is a method for computing similarity in the transformed space using the original attribute set. The measure of similarity used is the dot product of the two input vectors. The similarity function computed in the original attribute space is called the 'kernel function'. The kernel trick eliminates the need to know the exact form of the mapping function.

Multi class SVM

Multi class SVM classifies data into more than one class. For multiclass case, we transform the problem into multiple binary classification problems.

ii) Minimum distance classifiers

The minimum distance classifiers, as the name suggests, classify the data on the basis of the distance between the data points in the feature space. It classifies in such a manner, so as to minimize the distance between the data and the class in multi-feature space. The index of similarity here is the distance so that the minimum distance is identical to the maximum similarity. The most common distances often used in this procedure are:

Euclidian distance

The Euclidean distance is also known as the Euclidean metric. It is same as the "ordinary" distance between two points that is measured with a ruler, and is given by the Pythagorous formula. By using this formula as distance, Euclidean space or any other inner product space becomes a metric space. The norm of the Euclidian distance is called the Euclidean norm.

It is used in those problems where the variances of the population classes are different from each other. Theoretically the Euclidian distance is identical to the similarity index.

The Euclidean distance between two points 'p' and 'q' is the length of the line segment connecting them.

If **p** = ($p_1, p_2, \ldots, p_n$) and **q** = ($q_1, q_2, \ldots, q_n$) are two points in Euclidean n dimensional space, then the Euclidian distance from **p** to **q** is given by:

$$d(p,q) = d(q,p) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \ldots\ldots + (p_n - q_n)^2} = \sqrt{\sum_{i=1}^{n} (p_i - q_i)^2} \qquad (0.35)$$

The position of a point in a Euclidean *n*-space is shown by a Euclidean vector. So, **p** and **q** are Euclidean vectors, starting from the origin of the space. The **Euclidean norm**, or **Euclidean length**, or **magnitude** of a vector measures the length of this vector and is given by:

$$x(t), \frac{1}{\sqrt{A}} y(\frac{t-\tau}{A}), z(t) \qquad (0.36)$$

Here the last equation involves the dot product.

Mahalanobis distance

Mahalanobis distance was introduced by P. C. Mahalanobis in 1936. Its basis is the correlations between variables by which different patterns can be identified and analyzed. It measures the similarity of an unknown sample set to a known one. It differs from Euclidean distance in the manner that it takes into account the correlations of the data set and is scale-invariant. To put in other words, it is a multivariate effect size.

The formal definition of the Mahalanobis distance of a multivariate vector $x = (x_1, x_2, \ldots\ldots, x_n)^T$ is:

$$D_M(x) = \sqrt{(x-\mu)^T S^{-1} (x-\mu)} \tag{0.37}$$

Where, μ is the mean vector represented as:

$$\mu = (\mu_1, \mu_2, \ldots\ldots\mu_n)^T \tag{0.38}$$

and S is the covariance matrix.

Mahalanobis distance also called "generalized squared interpoint distance" for its squared value, can also be defined as a dissimilarity measure between two random vectors $\vec{x}$ and $\vec{y}$ of the same distribution with the covariance matrix S as:

$$d(\vec{x}, \vec{y}) = \sqrt{(\vec{x}-\vec{y})^T S^{-1} (\vec{x}-\vec{y})} \tag{0.39}$$

If the covariance matrix is the identity matrix, the Mahalanobis distance becomes same as the Euclidean distance. On the other hand, if the covariance matrix is diagonal, the resulting distance measure is called the normalized Euclidean distance.

iii) Artificial Neural networks (ANN)

Artificial neural network classifiers are inspired by biological neural systems. The nerve cells in human brain are called neurons. These are linked with other neurons via strands of fiber called axons. Whenever the neurons are stimulated, axons transmit nerve impulses from one neuron to another. Extensions from the cell body of the neuron are called dentrites. Dentrites connect one neuron to the axons of other neurons. The connection between a dentrite and an axon is called a synapse. It has been discovered that the human brain learns by changing the strength of the synaptic connection between neurons when stimulated repeatedly by the same impulse.

The ANN has a structure analogous to the human brain. It is composed of an interconnected assembly of nodes and direct links. These models can be trained for the purpose of classification. The simplest model is called perceptron.

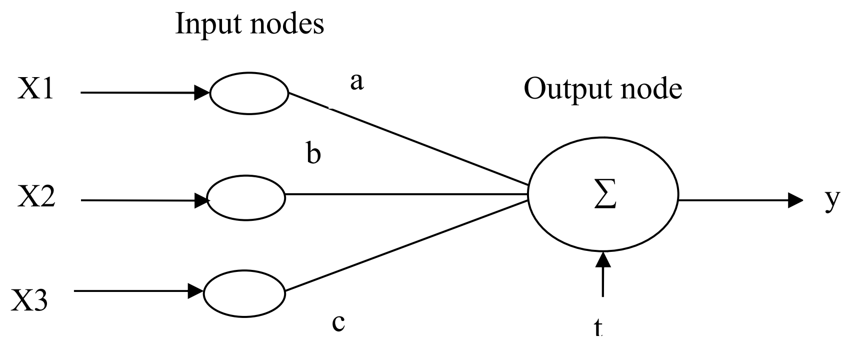


Figure 3.3: Model of a perceptron

The perceptron consists of two kinds of nodes, the input nodes and the output node. Input nodes are used to represent the input attributes and the output node represents the model output. The nodes in the neural network are called neurons are units. In a perceptron, each of the input nodes is connected to the output node via a weighted link. The weight of the link is used to emulate the strength of the synaptic connection between neurons. The perceptron computes the output value by calculating a weighted sum of the inputs, subtracting a bias factor 't' from the sum and then examine the sign of the result. Training of the perceptron network involves changing and adapting the weight of the links until they fit the input output relationship of the training data i.e. the outputs of the perceptron become consistent with the true outputs of training data.

In an ANN model, an input node simply transmits the value it receives to the outgoing link without any transformation. On the other hand, the output node is a mathematical device which computes the weighted sum of its inputs subtracts the bias term and produces an output that depends on the sign of the result. The sign function is an activation function for the output neuron, it value is +1 if positive and -1 if negative.

Once the model is trained, it can now be used to test new examples and classify them. The perceptron learning algorithm converges to an optimum solution for linearly separable classification problems. If the problem is not linearly separable the algorithm fails to converge and in that case, multilayer artificial neural network is needed.

A multilayer neural network has a more complex structure than a perceptron. It contains several intermediate layers between its input and output layers. These intermediate layers are called hidden layers and the nodes in these layers are called hidden nodes. There are two types of networks, feed forward and recurrent networks. In feed forward networks, the nodes in one layer are connected only to the nodes in the next layer while in a recurrent network; the links may connect the nodes in the same layer or in the previous layers. The activation function may be other than sign function also like linear or sigmoid function. These activation functions allow the hidden and the output nodes to produce non linear outputs. This type of complex structure can classify problems which are not linearly separable and have non linear solutions. This model will converge to the right solution when sufficient training data is provided.

iv) K Nearest neighbor (KNN) classifier

In pattern recognition and classification, the k-nearest neighbor algorithm (k-NN) is an algorithm for classifying data objects based on closest training examples in the feature space. The KNN is a type of instance based or lazy learning, where the function is only approximated locally and all computation is postponed till classification. The k-nearest neighbor algorithm is simplest among all machine learning algorithms: an object is classified by a majority vote of its neighboring data, and the test object is assigned to the class which is most common amongst its k nearest neighbors (k is a small positive integer). If ‘k’ is chosen to be 1, then the test object is simply assigned to the class to which its nearest neighbor belongs.

Although there is no explicit need for training in this algorithm, the neighbors can be regarded as training examples and are chosen from a set of objects for which the correct classification is known. The k-nearest neighbor algorithm is affected by the local structure of the data. The nearest neighbor rules effectively compute the decision boundary in an implicit way.

The training examples can be regarded as vectors in a multidimensional feature space, each belonging to a class. So, in the training phase of the algorithm, it is required to only store the feature vectors of the training samples along with their class.

In the classification phase, k is a constant decided by the user and a test vector (can also be called query or a test point) is classified by assigning it the label which is most common among the k training data nearest to that particular test point.

Euclidean distance is most commonly used as the distance measure; however this only applies to the continuous variables. For other types of classification like text classification, other measures such as the overlap metric (or Hamming distance) can also be used. The accuracy of the k nearest neighbor algorithm can be improved significantly by using special algorithms such as Large Margin Nearest Neighbor or Neighborhood components analysis.

The concept of 'majority voting' for classification has one drawback. If more examples in the training sample belong to one class, it tends to dominate the prediction of test sample, since their chances of being present in the k nearest neighbors is high owing to their large numbers. So the training examples have to be chosen very carefully. This problem can also be overcome by taking into consideration the distance from the test vector to each of the k nearest neighbors.

Choosing the value of 'k' also depends on the data. Larger values of 'k' help in reducing the effect of noise by averaging it out. But it makes the boundaries between the classes less distinct. There are various techniques for selecting the optimum value of 'k' like cross validation. The special case when the value of 'k' is '1', i.e. the testing example is assigned the class to which its nearest neighbor belongs, is called the nearest neighbor algorithm.

The presence of noisy and irrelevant data severely degrades the performance of the algorithm. Much research and effort has to be put into selecting features for improved classification. For problems with two classes, ‘k’ should be chosen to be an odd number so that there are no tied votes.

Chapter 4

EXPERIMENTS AND RESULTS

Four approaches were used for the purpose of feature extraction: Hu moments, Legendre moments, Zernike moments and cumulants. The standard ORL database is used for conducting all the experiments. It consists of images of 40 subjects of size 92 x 112 as shown in Figure 4.1

Figure 4.1: Subjects of the ORL database

Each of the 40 subjects has 10 orientations. Some orientations of sample image (subject 2) are shown in figure 4.2.

Figure 4.2 : Different orientations of subject 2 in ORL database

For all the methods implemented, 8 orientations (of each subject) were used for training and two orientations for testing. The entire programming is done using the software 'MATLAB'. An introduction to the programming in matlab has been provided in appendix A.

4.1 Image preprocessing

Image normalization is done in order to reduce computational overhead and improve the performance of the method. The images in the ORL database are grayscale images which contain intensity values ranging from 0 to 255. The pixel values of all the images are divided by the maximum pixel value of the image. After normalization, the values of all the pixels lie between 0 and 1.

4.2 Feature extraction

Feature extraction of all the images is done using the four moments. The procedure is explained below.

4.2.1 Hu moments

All the seven Hu moments were calculated for each facial image. The feature vector is formed using the calculated moment values which makes the length of the feature vector seven. Table 4.1consists of 10 feature vectors for 10 orientations of the subject shown in Figure 4.2.

Image no	M_1	M_2	M_3	M_4	M_5	M_6	M_7
2-1	0.0012	1.1145	1.7522	2.7004	1.0397	2.354	-2.9259
2-2	0.0012	0.9822	1.6067	1.7262	-0.6716	1.909	0.1064
2-3	0.0012	0.9967	2.451	3.2528	-2.0731	2.449	-0.4848
2-4	0.0012	1.0766	1.7206	1.4823	-1.1005	0.287	0.4849
2-5	0.0012	0.8636	0.2311	1.3562	-0.2185	-4.34	-0.0452
2-6	0.0012	0.9797	2.0262	1.8414	-2.2178	-1.646	1.6047
2-7	0.0012	1.1682	3.7579	4.4563	5.4096	13.215	1.0396
2-8	0.0012	1.0005	2.2362	2.5456	0.842	5.723	0.0691
2-9	0.0012	1.5421	2.3615	2.0594	-3.0265	-3.042	2.1438
2-10	0.0012	0.9115	2.5089	2.6352	-2.7682	1.922	0.2446

Table 4.1: Hu moments of subject 2 in ORL database

In the above table, we see that the first moment i.e. M_1 is same for all the orientations of the subject. Also, the variations in the lower order moments are less but in the higher order moments the variations are large.

4.2.2 Zernike moments

In the experiment conducted by us, Zernike moments of different orders (2 to 10) are calculated. The feature vector is formed using these calculated moments. The size of the feature vector is 7. Table 2 consists of 10 feature vectors for 10 orientations of the subject shown in Figure 4.2.

Image no	**Z_{20}**	**Z_{31}**	**Z_{42}**	**Z_{40}**	**Z_{51}**	**Z_{53}**	**Z_{62}**
2-1	-1.102	-6.4727	-1.3869	-7.4029	-3.2025	3.3126	-0.4755
2-2	-1.099	-6.6799	-1.1644	-7.2521	-3.5483	2.2785	0.078
2-3	-1.0975	-6.825	-1.3694	-7.1614	-3.6634	2.586	-0.0683
2-4	-1.0854	-6.7228	-1.2138	-7.1546	-3.5915	2.2853	0.046
2-5	-1.1187	-6.7846	-0.9088	-7.494	-3.6191	2.7186	0.4117
2-6	-1.1037	-6.9002	-1.3914	-7.3231	-3.8162	2.0686	-0.2327
2-7	-1.1259	-7.0017	-2.4854	-7.3039	-3.4261	2.067	-1.2901
2-8	-1.1072	-7.0425	-1.7104	-7.2503	-3.7064	2.4051	-0.3873
2-9	-1.1033	-7.0592	-1.9559	-7.0575	-3.772	1.2888	-0.687
2-10	-1.142	-7.1451	-1.7801	-7.4172	-3.7097	2.1011	-0.3757

Table 4.2: Zernike moments of subject 2 in ORL database

The results using Zernike moments were better than those using Hu moments. This is because we can see in the table above that the variations in the Zernike moments are less as compared to those in the Hu moments.

4.2.3 Legendre moments

Legendre moments of different orders (2 to 4) are calculated. The feature vector is formed using these calculated moments. The size of the feature vector is 12.

Image no	L_{02}	L_{20}	L_{11}	L_{21}	L_{12}	L_{30}	L_{03}	L_{22}	L_{40}	L_{04}	L_{31}	L_{13}
2-1	-233.0684	-234.6376	3.7947	-0.4277	5.5926	12.5533	19.3843	368.8896	273.1376	279.0631	-4.084	2.3298
2-2	-230.3231	-239.4266	-2.1027	-13.6531	4.432	10.4496	14.4886	364.0743	277.4235	277.3921	0.6326	0.5054
2-3	-236.6752	-242.6514	0.7408	-1.7748	5.2131	12.2524	18.8927	376.3221	281.0291	284.9147	-2.5927	0.9635
2-4	-226.3428	-236.1744	-2.8129	-13.2203	1.6416	11.1439	15.0938	355.6779	273.7384	277.1456	1.5852	0.1854
2-5	-234.1067	-240.1649	0.7465	-9.2252	5.7785	12.5979	9.7983	365.0728	281.927	278.9115	-2.8717	1.5969
2-6	-228.5322	-235.1552	-5.0898	-23.2529	3.6978	9.8463	17.8231	353.8425	281.7173	276.2261	3.2303	-0.1103
2-7	-246.2772	-250.0778	1.9193	8.7635	5.2179	14.7457	27.4591	387.8163	296.9202	307.2707	-1.3669	3.0821
2-8	-237.6888	-242.0423	0.3357	0.2377	3.0687	13.5121	22.0919	375.3208	283.5896	290.9639	-0.799	2.9246
2-9	-235.6327	-251.3874	-5.9082	-13.7431	9.5633	8.7386	24.6543	377.4077	295.5176	299.235	4.1938	-0.6792
2-10	-250.6621	-252.5336	1.0875	2.0345	8.0761	14.0975	23.267	389.5401	304.0091	305.0874	-1.7148	2.0354

Table 4.3: Legendre moments of subject 2 in ORL database

The variations in higher order moments were large which degraded the performance of the classifier. Hence, for the construction of feature vectors only lower order moments (up to four) were calculated.

4.2.4 Cumulants

i) Bispectrum calculation

We first find the third order cumulant of the image and then take its Fourier transform to get the bispectrum. For this purpose, we had used the 'HOSA: Higher order Spectral Analysis' toolbox. On calculating the bispectrum of the different facial images, we find that the bispectrum pattern is similar for the images of the same person (in different orientations) and different for the images of different persons. Few orientations of subject 1 are shown in figure 4.3:

Figure 4.3: Four orientations of subject 1

Figure 4.4 shows the bispectrum patterns of the subject in figure 4.3.

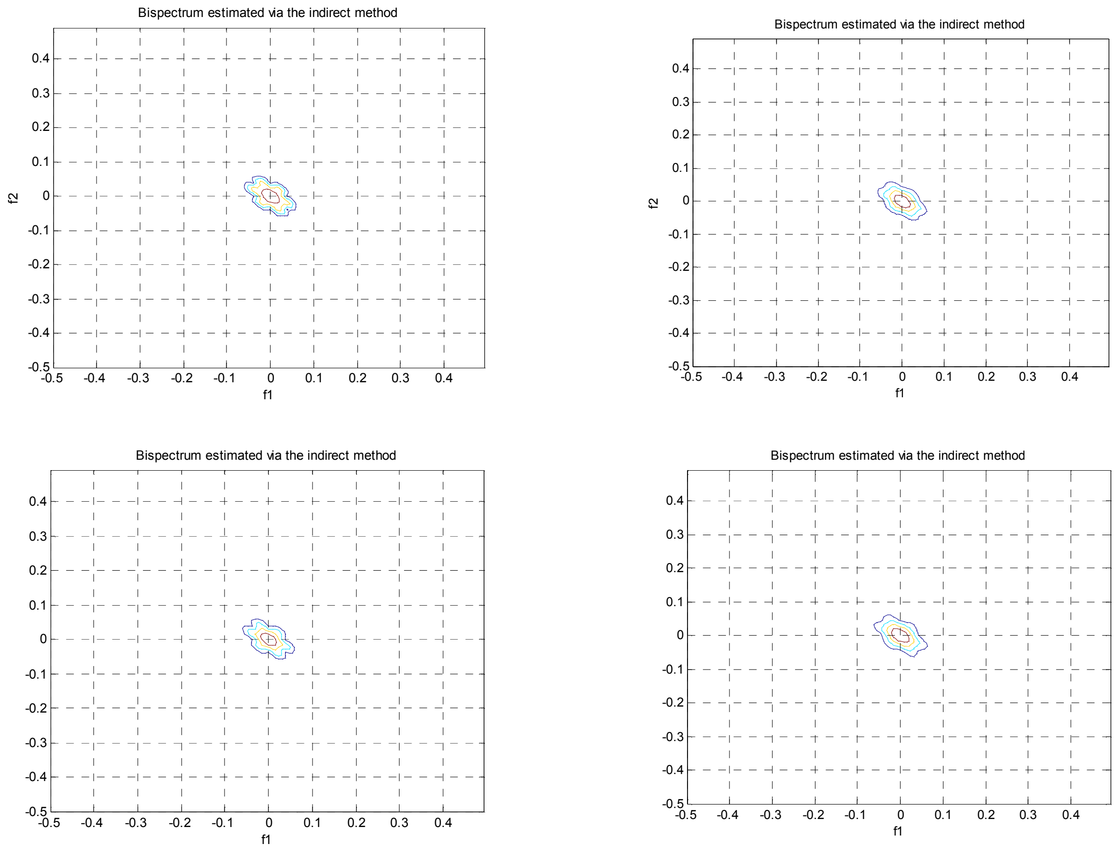


Figure 4.4: Bispectrum patterns of some orientations of subject 1

All the above patterns are similar to each other since they belong to the images of same person with different orientations. Figure 4.6 shows the bispectrum patterns of four orientations subject 4 shown in Figure 4.5.

Figure 4.5: Four orientations of subject 4

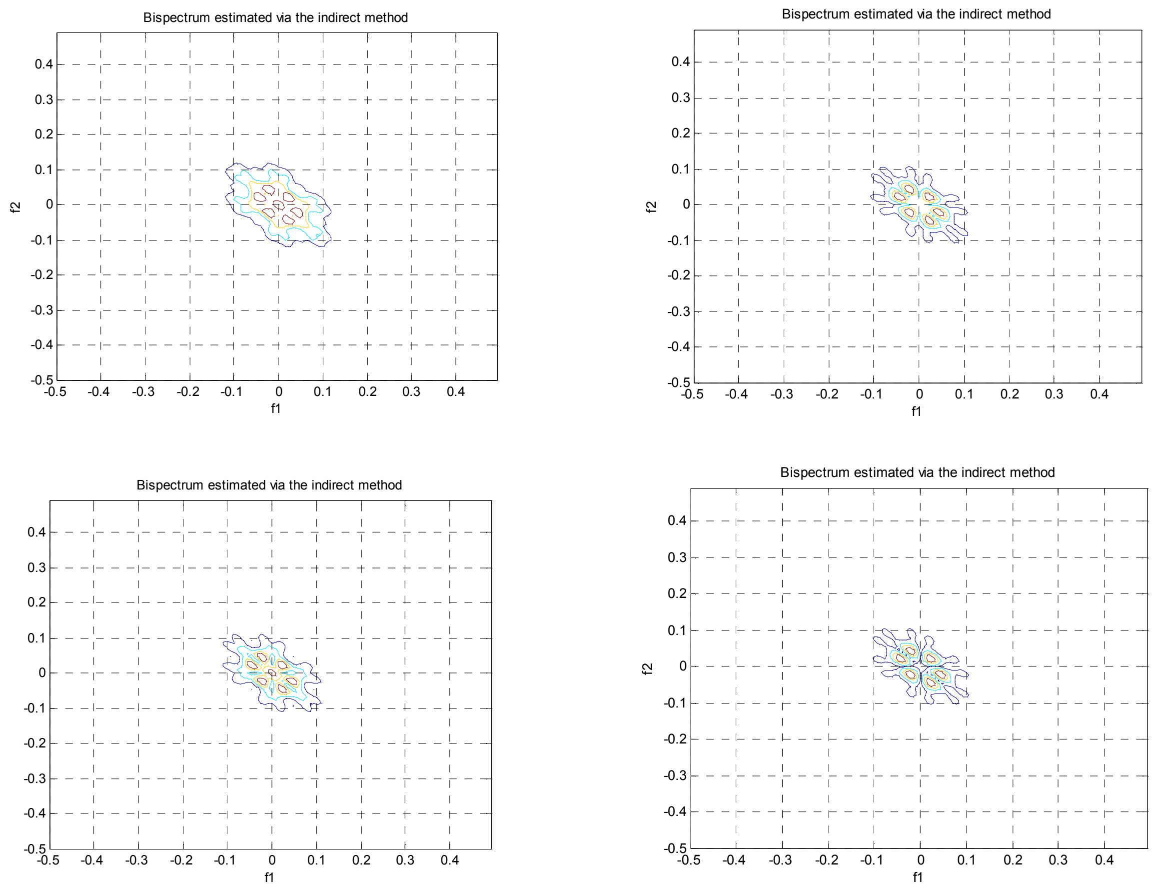


Figure 4.6: Bispectrum patterns of some orientations of subject 4

Again we notice that all the above patterns are similar to each other since they belong to the images of same person with different orientations. Also, the bispectrum patterns of subject 1 are different from those of subject 4, which shows that they can be classified easily.

To ensure that the bispectrum patterns for all the subjects are different, we present below one more example. Figure 4.7 shows four orientations of subject 10 in the ORL database.

Figure 4.7: Four orientations of subject 10

Figure 4.8 shows the bispectrum patterns of the orientations of subject shown in above figure.

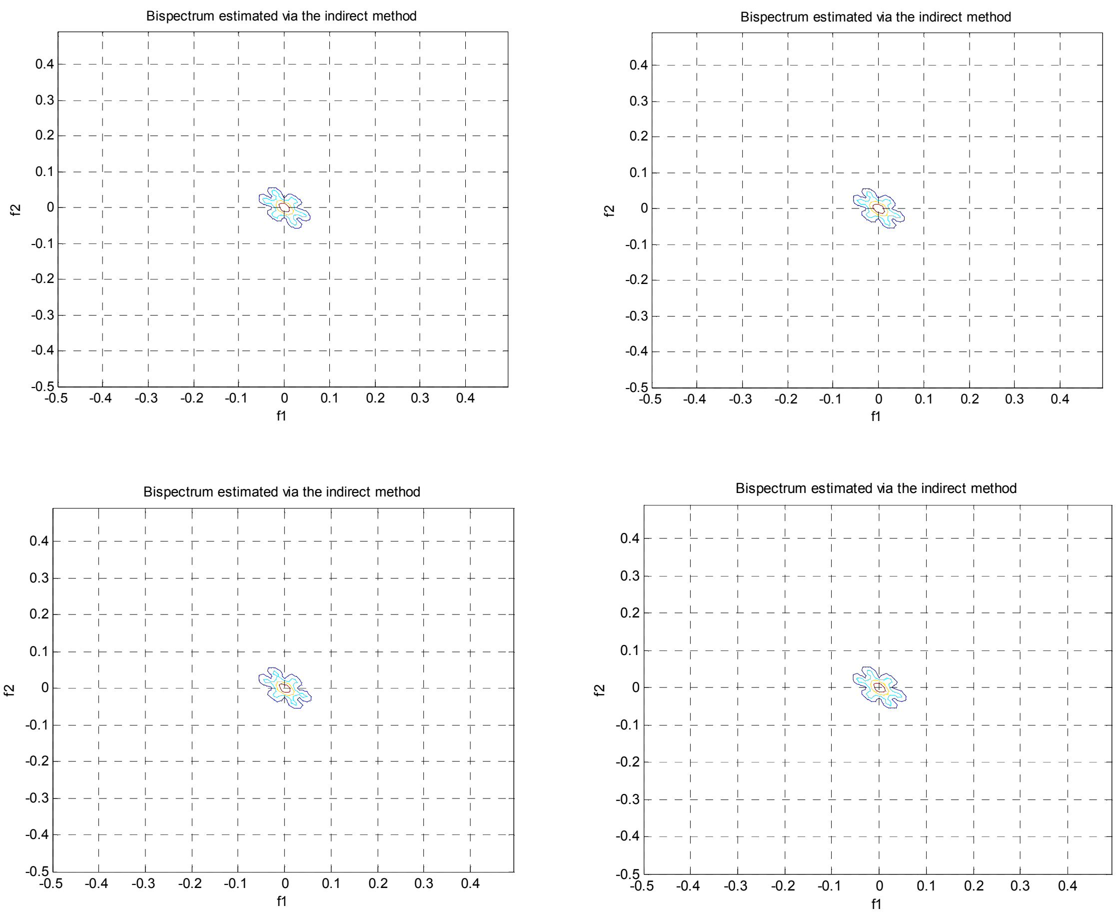


Figure 4.8: Bispectrum patterns of some orientations of subject 10

Again we see that the orientation patterns of subject 10 are different both from subject 1 and subject 4. Hence, we can conclude that the bispectrum patterns of different orientations of same subjects are similar while those of different subjects are different. This fact can be used to classify the different subjects. For ease of classification, we will form the biwavelant of the above patterns.

ii) Biwavelant calculation

For using bispectra for face recognition, the redundant information i.e. the high frequency components have to be removed and only the low pass information which corresponds to the significant features are retained.

The two dimensional bispectrum matrix is converted into one dimensional matrix by concatenating the rows horizontally. The resulting frequency response is given as:

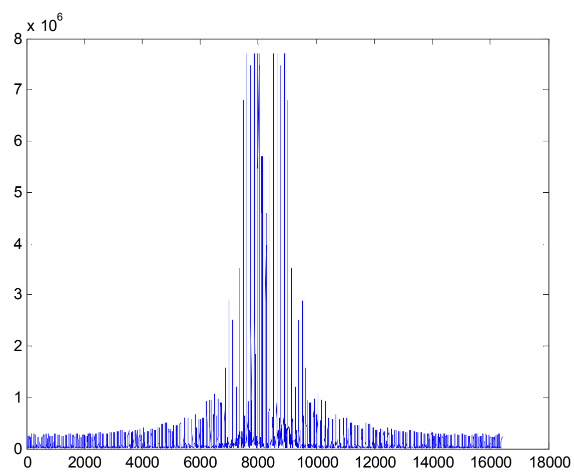


Figure 4.9: One dimensional frequency response of bispectrum

The frequency response above consists of very large number of values. To classify such a vector will result in huge amount of computation. But we can see that the response has a particular shape, so we try to find out its envelope (by removing the high frequency components). Using the envelope for classification also reduces the data size. For this, we need to carry out low pass filtering which can be done using wavelets. Smoothing filters can also be used but they are not able to protect the precious details while removing the high pass information.

Wavelet transform was explained in [39], [40] and [41]. It is the integral transform defined as in equation (4.1):

$$W_{\psi f}(a,b) = \frac{1}{\sqrt{|a|}} \int_{-\infty}^{\infty} \overline{\psi\left(\frac{x-b}{a}\right)} f(x)\, dx \tag{4.1}$$

where f(x) is the signal being transformed and   is the ‘analyzing wavelet’   satisfies the admissibility condition which is equivalent to   i.e. a wavelet has zero mean.

The wavelet coefficients are given by equation (4.2):

$$c_{jk} = W_{\psi f}\left(2^{-j}, k2^{-j}\right) \tag{4.2}$$

Here, $a=2^{-j}$ is called the binary dilation or dyadic dilation, and $b=k2^{-j}$ is the binary or dyadic position.

The wavelet transform has been used as a multi resolution tool for many applications like fractal signal analysis, pitch detection and image compression etc. When we use the continuous wavelet transform, we can detect a signal buried in Gaussian noise. This fact has been used in the concept of wavelants. Before understanding wavelants, we need to know two properties of cumulants [43]:

1. The third order cumulant of a Gaussian (or any symmetrically distributed) random process is zero.
2. If a subset of random variables is independent of the rest, then the third-order cumulants is zero.

The above formulation exhibits properties closely related to those of cumulants. The ideas from the preceding two topics have lead to the motivation for the development of wavelants which is a combination of wavelet and cumulant theory. In the following section we shall consider only the third order wavelant which is defined as in equation (4.3):

$$W_{xxx}^{3}\left(b_1,a_1;b_2,a_2\right)=\frac{1}{\sqrt{a_1a_2}}\int\int x\left(t\right)x\left(\frac{t-b_1}{a_1}\right)x\left(\frac{t-b_2}{a_2}\right)dt \tag{4.3}$$

$$W_{xyz}^{3}\left(b_1,a_1;b_2,a_2\right)=\frac{1}{\sqrt{a_1a_2}}\int\int x\left(t\right)y\left(\frac{t-b_1}{a_1}\right)z\left(\frac{t-b_2}{a_2}\right)dt \tag{4.4}$$

Equation (4.3) represents the third order auto wavelant while equation (4.4) represents the third order cross wavelant.

The 2-D cross-wavelant for an image can be expressed as:

$$W_{xyz}^{3}\left(b_{x_1},b_{y_1},a_{x_1},a_{y_1};b_{x_2},b_{y_2},a_{x_2},a_{y_2}\right)=\frac{1}{\sqrt[4]{a_{x_1}a_{y_1}a_{x_2}a_{y_2}}}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}x\left(t_x,t_y\right)y\left(\frac{t_x-b_{x_1}}{a_{x_1}},\frac{t_y-b_{y_1}}{a_{y_1}}\right)z\left(\frac{t_x-b_{x_2}}{a_{x_2}},\frac{t_y-b_{y_2}}{a_{y_2}}\right)dt \tag{4.5}$$

Properties: When the input used for computing the wavelant is translated and/or followed by dilation then the following properties result:

If $x\left(t\right),y\left(t\right),z\left(t\right)$ maps to $W_{xyz}^{3}\left(b_1,a_1;b_2,a_2\right)$

then, $x\left(t\right),\sqrt{A}y(At-\tau),z(t)$ maps to $W_{xyz}^{3}\left(Ab_1-\tau,a_1A;Ab_2-\tau,a_2A\right)$

and $x\left(t\right),\frac{1}{\sqrt{A}}y(\frac{t-\tau}{A}),z(t)$ maps to $W_{xyz}^{3}\left(b_1+a_1\tau,a_1A;b_2,a_2\right)$

and $x\left(t\right),y(t),\frac{1}{\sqrt{A}},z(\frac{t-\tau}{A})$ maps to $W_{xyz}^{3}\left(b_1+a_1;b_2+a_2\tau,a_2A\right)$ (4.6)

However, if the input is first dilated and then translated, then the results are given by:

If $x(t), y(t), z(t)$ maps to $W_{xyz}^3(b_1, a_1; b_2, a_2)$

then $x(At-\tau), y(t), z(t)$ maps to $W_{xyz}^3(Ab_1-\tau, a_1 A; Ab_2-\tau, a_2 A)$

and $x(t), y(t), \sqrt{A}z(At-\tau)$ maps to $W_{xyz}^3\left(b_1, a_1; \frac{b_2-\tau}{A}, \frac{a_2}{A}\right)$

and $x(t), y(t), \sqrt{A}z(At-\tau)$ maps to $W_{xyz}^3\left(b_1, a_1; \frac{b_2-\tau}{A}, \frac{a_2}{A}\right)$ (4.7)

Now we apply the wavelet transform (db4) at different levels on the frequency response of bispectrum obtained earlier (example of which is shown in Figure 4.9). The result of applying the transform on an image of subject 1 is shown in Figure 4.10.

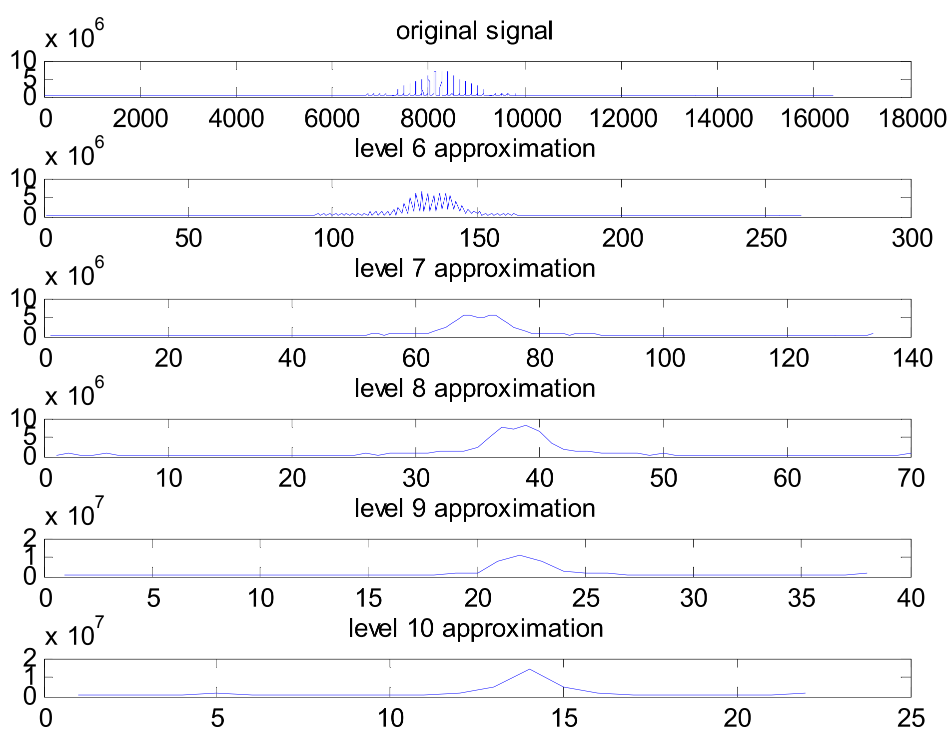


Figure 4.10: waveforms for subject 1 at various approximations

We notice from the above figure, that we obtain a nice envelope at Level 7 approximation. This seems to be the most appropriate approximation level for classification because in further approximations, details are being lost. The similar waveforms for subjects 4 and 10 are shown below.

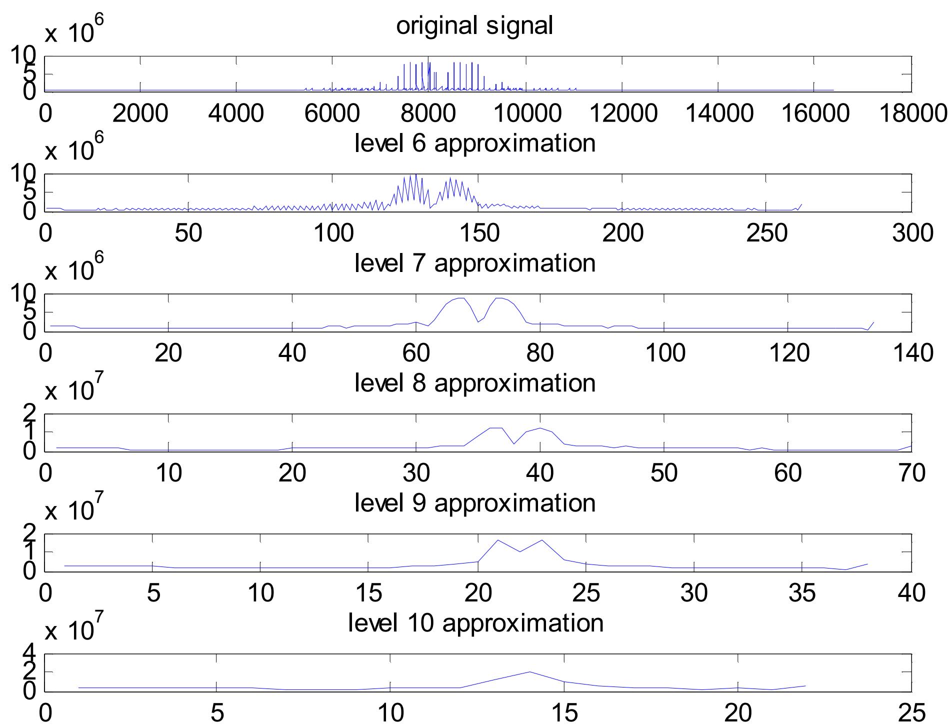


Figure 4.11: waveforms for subject 4 at various approximations

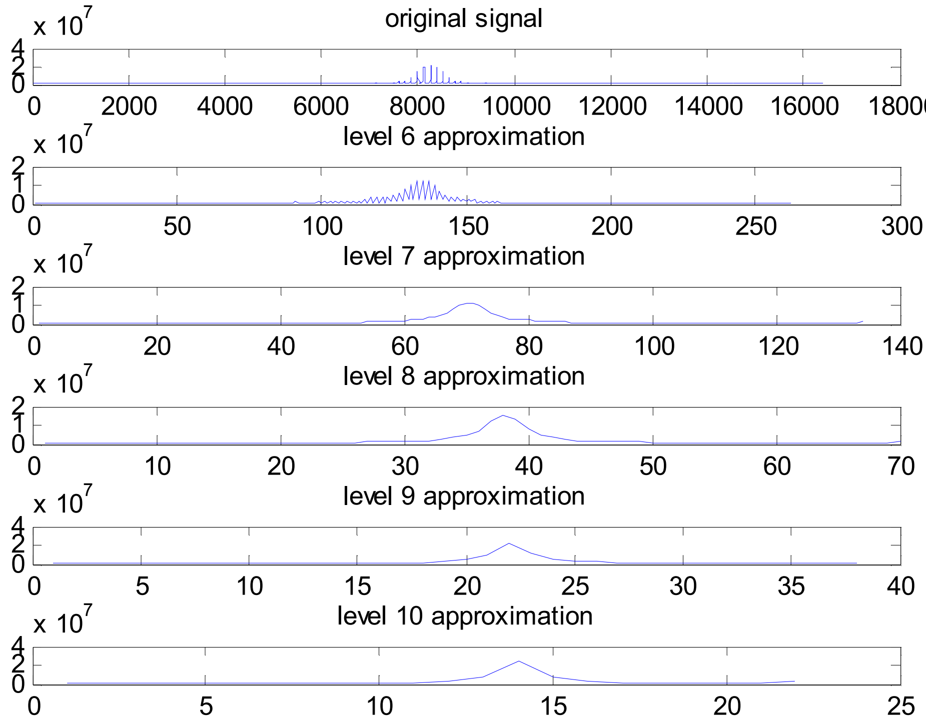
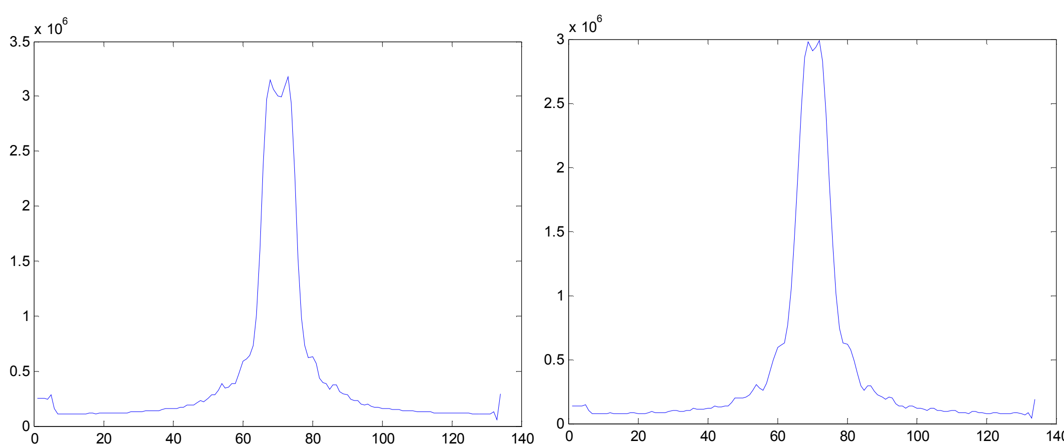


Figure 4.12: waveforms for subject 10 at various approximations

In all the above waveforms, level 7 approximation is most appropriate and also at this level all the three waveforms can easily be distinguished from one another.

Let us now have a closer look at the level seven approximation of the frequency response for few orientations of all the three subjects.



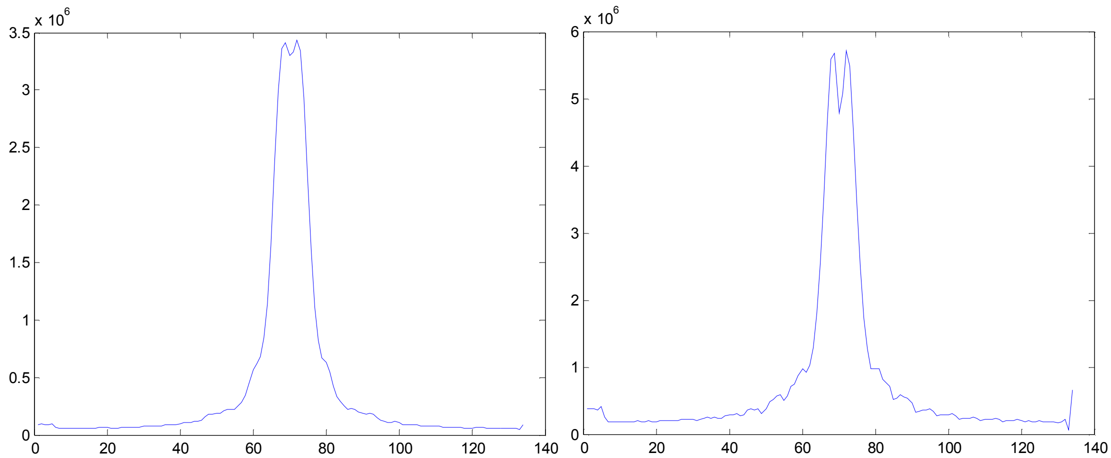


Figure 4.13: Level 7 approximations of frequency responses of bispectrum matrix of subject 1

All the above waveforms belong to subject 1 and are similar to each other.

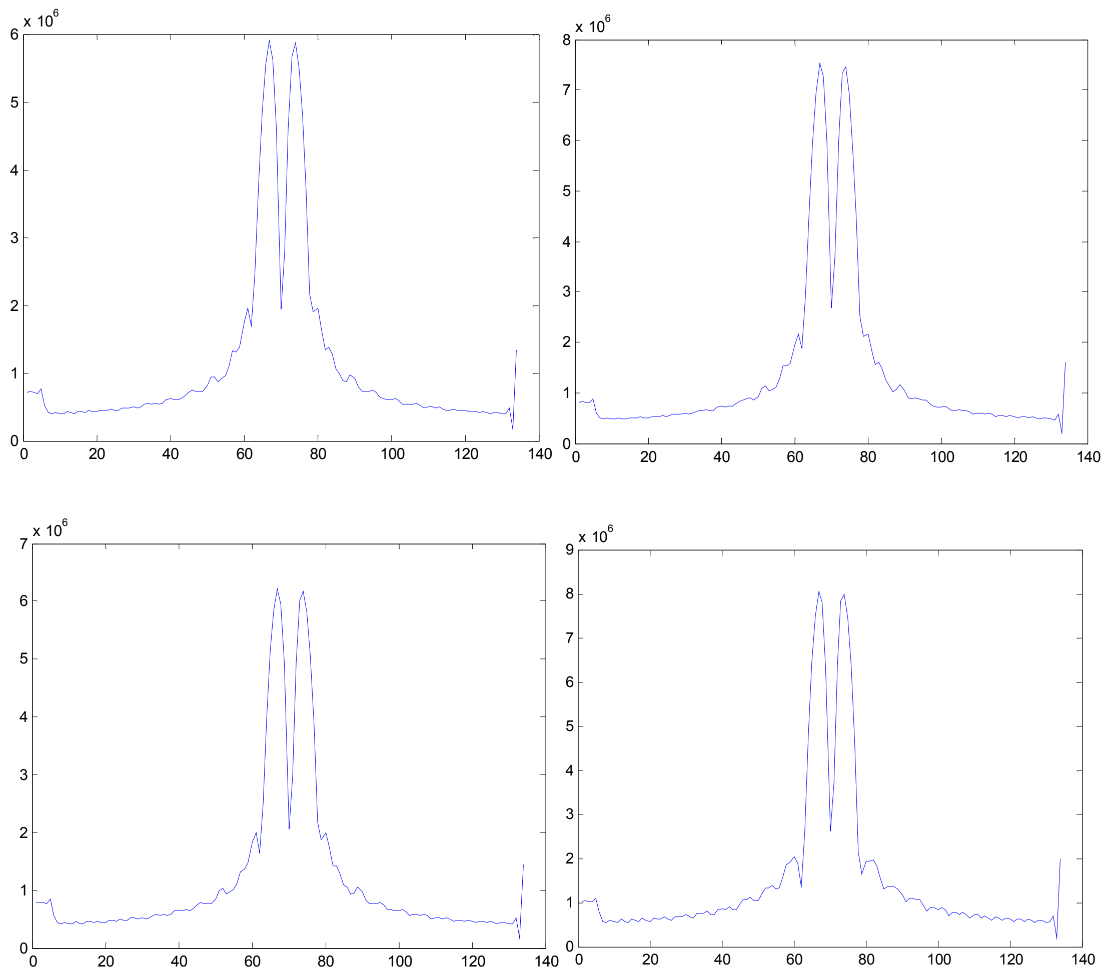


Figure 4.14: Level 7 approximations of frequency responses of bispectrum matrix of subject 4

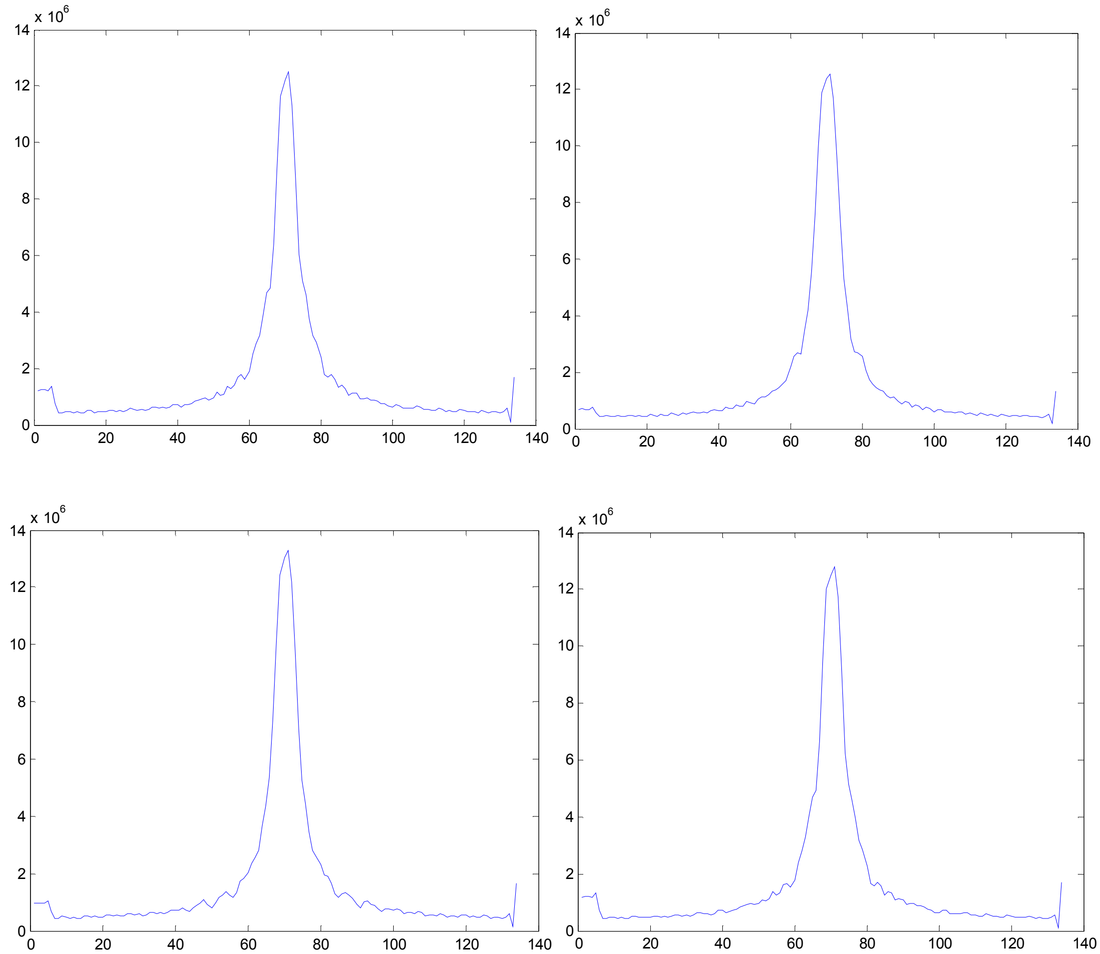


Figure 4.15: Level 7 approximations of frequency responses of bispectrum matrix of subject 10

We notice from the above figures, that the waveforms for different orientations of one subject are similar while for different subjects they are different. This fact makes classification of these waveforms possible.

4.3 Classification

Classification of the feature vectors obtained is as important as the extraction of features. Classification was done using three different classifiers: Minimum distance classifier, Support Vector Machine and K Nearest Neighbor classifier. The results obtained using these classifiers vary slightly but are more or less similar.

4.3.1 Minimum distance classifier

The classification was done using the minimum distance classifier. Euclidean distance has been used for calculating the distance. Since the total number of subjects is 40, there were 40 classes. After classification, following results were obtained:

Sr. No	Feature extraction method	Classification accuracy
1.	Hu moments	60%
2.	Zernike moments	90%
3.	Legendre moments	90%

Table 4.4: Classification accuracy of minimum distance classifier

The results of Hu moments are not as good as those of Zernike and Legendre moments as can be seen from the table above.

4.3.2 Support vector machine

Classification using non linear SVM was done using the following parameters:

Sr. No.	Parameters	Parameter value
1.	Classifier Type	Non linear multi class SVM
2.	Kernel	Poly
3.	C	1000

Table 4.5: Various parameters used to design the SVM

The results of various feature extraction methods using SVM are shown below:

Sr. No	Feature extraction method	Classification accuracy
1.	Hu moments	60%
2.	Zernike moments	85%
3.	Legendre moments	95%

Table 4.6: Classification accuracy of SVM

Here again, the results of Hu moments are the poorest. Also, the results of Legendre moments improved as compared to the ones obtained from minimum distance classifier, while those of Zernike moments have degraded.

4.3.3 K Nearest Neighbor

When classified using the k nearest neighbor algorithm, most accurate results were obtained when the value of k = 3. The following figure shows the classification results obtained for face recognition using the KNN classifier for Hu moments:

Each point in Figure 4.16 represents the feature vector corresponding to an image. Each sample is classified and associated to one of the ten clusters (subjects) i.e. to a particular subject. The clusters are shown with different colors and markers for easy understanding.

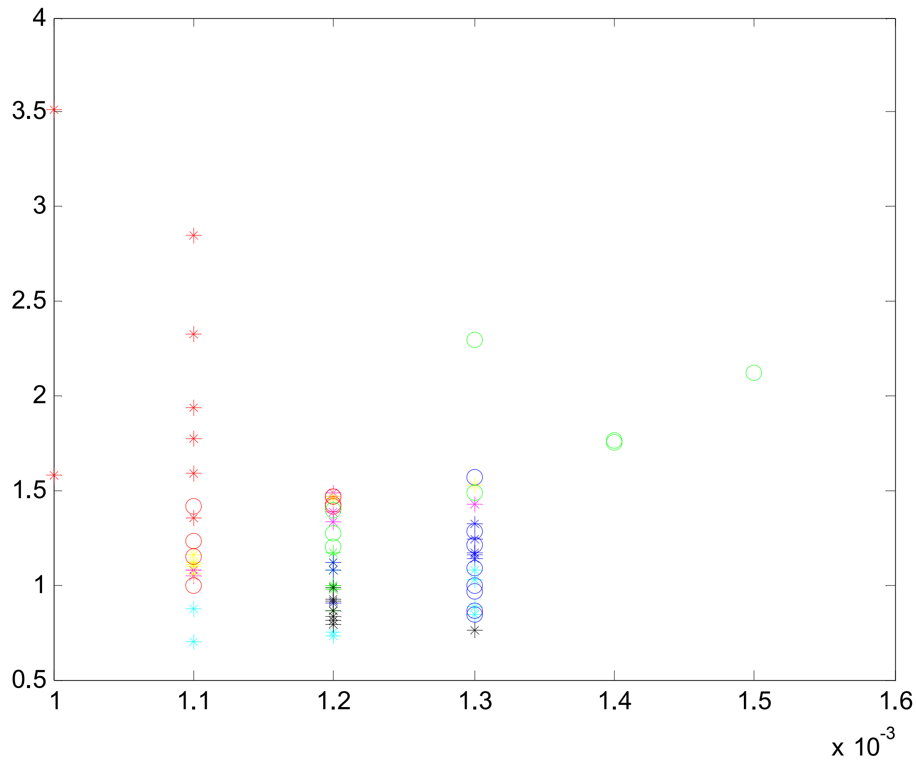


Figure 4.16: Classification results obtained using Hu moments

In the above figure we see that the classes are not very distinct and kind of overlap. Hence the performance of Hu moments is not very good.

Figure 4.17 shows the classification results obtained for face recognition using the KNN classifier for Zernike moments:

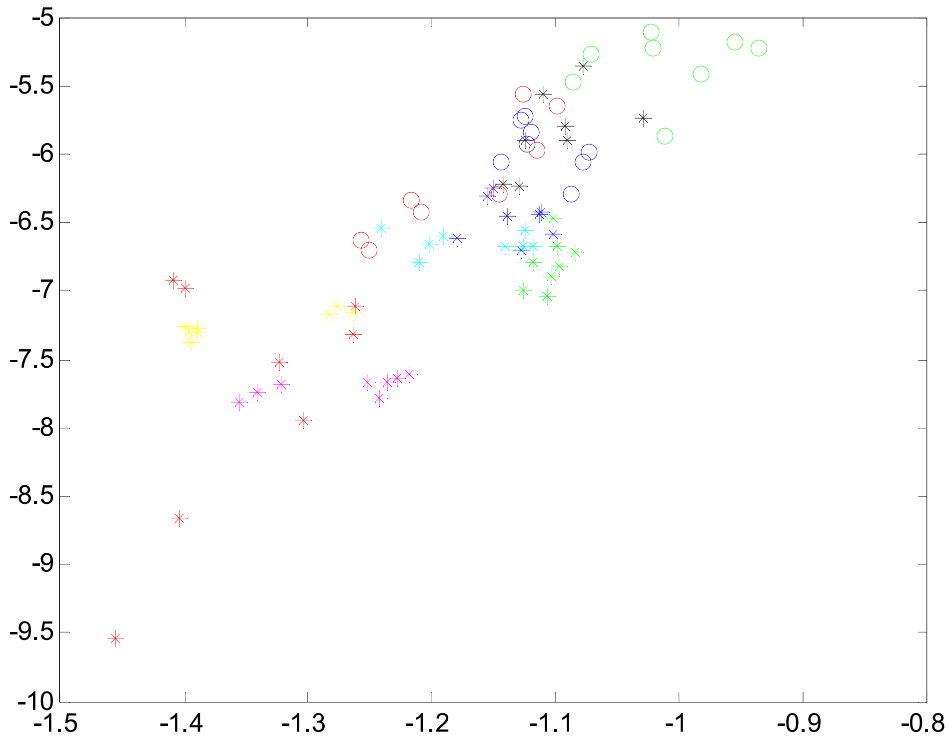


Figure 4.17: Classification results obtained using Zernike moments

In the above figure, the cluster formation is better than that obtained from Hu moments. Hence the classification is better.

Figure 4.18 below shows the classification results obtained for face recognition using the KNN classifier for Legendre moments:

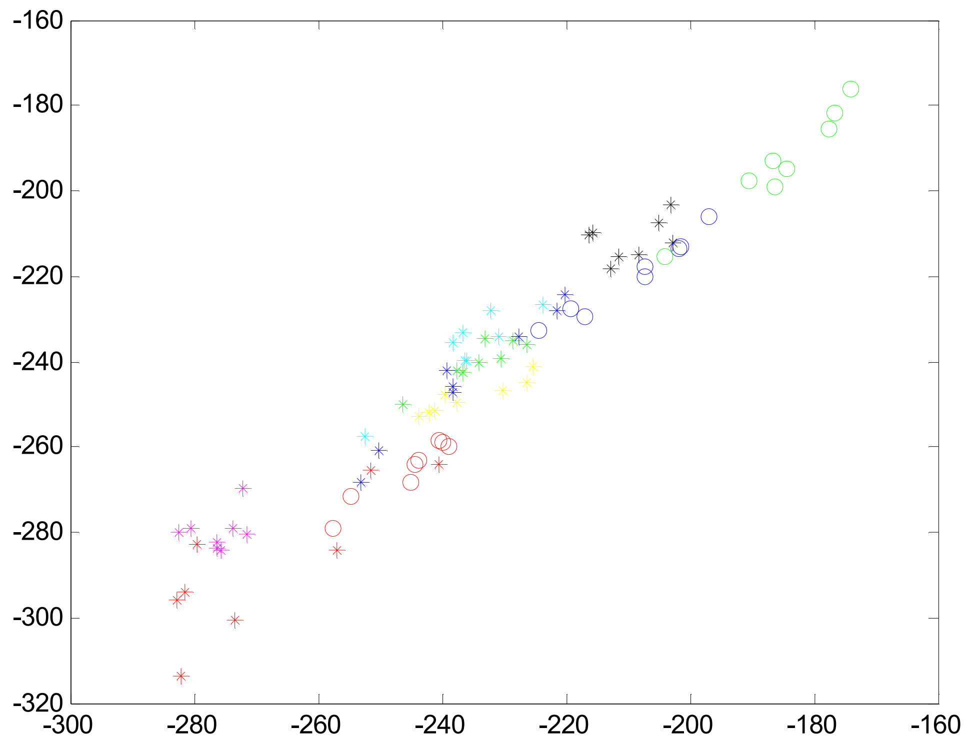


Figure 4.18: Classification results obtained using Legendre moments

Here also the cluster formation is good. The results of using k nearest neighbor classifier on the three feature extraction methods are shown below in Table 4.7:

Sr. No	Feature extraction method	Classification accuracy
1.	Hu moments	40%
2.	Zernike moments	85%
3.	Legendre moments	95%

Table 4.7: Classification accuracy of KNN

Once again the performance of Hu moments is the poorest. The results of Zernike and Legendre are same as those obtained from SVM.

4.4 Results

Four experiments were performed for extracting the features using four approaches: Hu moments, Legendre moments, Zernike moments and cumulants. The feature vectors obtained using these moments were classified using three methods: minimum distance classifier, SVM and KNN. The average performance of the feature extraction methods can be tabulated as below:

Sr. No	Feature extraction method	Classification accuracy
1.	Hu moments	53.33%
2.	Zernike moments	86.66%
3.	Legendre moments	93.33%

Table 4.8: Average performance of the feature extraction methods

From the above table, we can infer that the performance of Hu moments was not satisfactory and hence they cannot be used for the purpose of face recognition using complex facial images. On the other hand, the results obtained using Zernike and Legendre moments are good.

The confusion matrix for first ten subjects (two test cases per subject) using Hu moments and classified using minimum distance classifier is presented below:

subjects	1	1	2	2	3	3	4	4	5	5	6	6	7	7	8	8	9	9	10	10
1	√																			
1		√																		
2			√																	
2				√																
3													√							
3														√						
4																	√			
4													√							
5			√																	
5				√																
6											√									
6												√								
7													√							
7														√						
8															√					
8			√																	
9																	√			
9																		√		
10											√									
10																				√

Table 4.9: Confusion matrix for Hu moments using minimum distance classifier

The confusion matrix for first ten subjects (two test cases per subject) using Zernike moments and classified using minimum distance classifier is presented below:

subjects	1	1	2	2	3	3	4	4	5	5	6	6	7	7	8	8	9	9	10	10
1	√																			
1		√																		
2			√																	
2				√																
3					√															
3						√														
4							√													
4								√												
5									√											
5										√										
6											√									
6												√								
7													√							
7														√						
8															√					
8																√				
9																	√			
9													√							
10																			√	
10															√					

Table 4.10: Confusion matrix for Zernike moments using minimum distance classifier

The confusion matrix for first ten subjects (two test cases per subject) using Legendre moments and classified using minimum distance classifier is presented below:

subjects	1	1	2	2	3	3	4	4	5	5	6	6	7	7	8	8	9	9	10	10
1	√																			
1			√																	
2			√																	
2				√																
3					√															
3						√														
4							√													
4								√												
5									√											
5										√										
6											√									
6												√								
7													√							
7														√						
8															√					
8																√				
9																	√			
9																		√		
10																			√	
10													√							

Table 4.11: Confusion matrix for Legendre moments using minimum distance classifier

For the experiment performed using cumulants, we have obtained waveforms which are similar for different orientations of same subject and different for different subjects. These waveforms can be classified using any suitable method.

Chapter 5

CONCLUSION AND FUTURE WORK

5.1 Conclusion

In the presented work, a novel method for face recognition using moments and cumulants has been proposed. Moments have already been used for handwriting recognition, fingerprint recognition etc. In the presented work we have used them for recognizing faces.

Four experiments were conducted using Hu moments, Zernike moments, Legendre moments and cumulants. Features were extracted using the above mentioned moments and feature vectors were formed. The size of the feature vector for Hu moment was 7, for Zernike 7 and for Legendre moments it was 12. In all the experiments, three different classifiers were used for classification: minimum distance classifier, SVM and KNN.

From the experimental results it was observed that the feature representation using Hu moments provides a maximum recognition rate of 60%. Orthogonal moments like Zernike and Legendre which are also invariant in nature are capable of representing the image features with minimum number of coefficients and the percentage of accuracy is also superior. Hence these moments have been found very much suitable for feature representation of complex face images which are almost similar with respect to variations in size, pose, illumination and orientation within a smaller area.

In this work, we also experimented with cumulants and used them for face recognition. The method can be used to successfully identify different faces. The proposed technique using cumulants is sensitive to the structural changes in the images and is able to distinguish them successfully. But, the same sensitivity makes the method vulnerable to noise in the samples, so the images have to be noise free for expected results.

5.2 Future scope

As one of the most successful applications of image analysis and understanding, face recognition has recently gained significant attention. Over the last ten years or so, it has become a popular area of research in computer vision and one of the most successful applications of image analysis and understanding.

Face recognition is a technology just reaching sufficient maturity for it to experience a rapid growth in its practical applications. Much research effort around the world is being applied for expanding the accuracy and capabilities of this domain, with a consequent broadening of its application in the near future. Verification systems for physical and electronic access security are available today, but the future holds the promise and the threat of passive customization and automated surveillance systems enabled by face recognition.

The largest example of face recognition systems in the world operates in the U.S. Department of State for visa processing. It consists of over 75 million photographs. There can be several other uses of face recognition in future. Some of them can be:

1. Face recognition can be used to prevent ATM frauds. A database of all ATM customers can be prepared with the banks. High resolution cameras will have to be deployed at all ATMs and face recognition software has to be used. Now, whenever a user will enter the ATM, his photograph will be taken and access to the ATM machine will be permitted only after the photo is matched with the database.

2. Duplicate votes are being reported in India. To prevent this, a database of all voters, of all constituencies, will have to be prepared. While voting, a camera installed at the voting site will capture a photo of the person and match it with the database. The person will be allowed to vote only after his photo matches with the database. If it is found that the person has already voted once, then he can be prevented from voting twice.

3. Similarly, passport and visa verifications can also be done using face recognition technology. Driving license verification can also be performed using face recognition technology as mentioned earlier.

4. For identifying and verifying terrorists at airports, railway stations and shopping malls also, face recognition technology will be the best choice in India as compared with other biometric technologies since other technologies cannot be helpful in crowds.

Along with all the advantages, face recognition technology has its own problems. With the widespread installation of security cameras and the increasing financial and technological feasibility of automating this surveillance, public fears have also increased about the potential for invasion of privacy that this technology can bring about.

Despite the potential benefits of this technology, many citizens are concerned that their privacy will be invaded. Some fear that it could lead to a “total surveillance society,” with the government and other authorities having the ability to know where you are, and what you are doing, at all times. This is not to be an underestimated concept as history has shown that states have typically abused such access before.

REFERENCES

[1] Stefano Berretti, Alberto Del Bimbo and Pietro Pala, “3D Face Recognition Using Isogeodesic Stripes,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 32, no. 12, December 2010

[2] Chaua C. Queirolo, Luciano Silva, Olga R.P. Bellon, and Maurı´Cio Pamplona Segundo, “3D Face Recognition Using Simulated Annealing and the Surface Interpenetration Measure,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 32, no. 2, February 2010

[3] Ira Kemelmacher Shlizerman and Ronen Basri, “3D Face Reconstruction From a Single Image Using a Single Reference Face Shape,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 33, no. 2, February 2011

[4] R. Salalloncha, E. Kokiopouloub, I. Tosic and P. Frossard, “3D Face Recognition with Sparse Spherical Representations,” Pattern Recognition 43 (2010) 824 – 834

[5] R. Ksantini, B. Boufama, Djemelziou, Bernardcolin, “A Novel Bayesian Logistic Discriminant Model: An Application to Face Recognition,” Pattern Recognition 43 (2010) 1421–1430

[6] Antonio Rama, Francesc Tarres and Jurgen Rurainsky, “Aligned Texture Map Creation for Pose Invariant Face Recognition,” Multimed Tools Appl (2010) 49:545–565 Doi 10.1007/S11042-009-0447-9

[7] Gaoyun An, Jiying Wu and Qiuqi Ruan, “An Illumination Normalization Model for Face Recognition Under Varied Lighting Conditions,” Pattern Recognition Letters 31 (2010) 1056–1067

[8] Chao Kuei Hsieh, Shang-Hong Lai and Yung-Chang Chen, "An Optical Flow-Based Approach to Robust Face Recognition Under Expression Variations," IEEE Transactions on Image Processing, Vol. 19, No. 1, January 2010

[9] Jian Yang A, Chengjunliu B and Leizhang, "Color Space Normalization: Enhancing the Discriminating Power of Color Spaces for Face Recognition," Pattern Recognition 43 (2010) 1454–1466

[10] Huiyu Zhou and Abdul H. Sadka, "Combining Perceptual Features with Diffusion Distance for Face Recognition," IEEE Transactions on Systems, Man, and Cybernetics Part C: Applications And Reviews, Vol. 41, No. 5, September 2011

[11] Yin Zhang and Zhi Hua Zhou, "Cost-Sensitive Face Recognition," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 32, No. 10, October 2010

[12] Weiwen Zou and Pong C. Yuen, "Discriminability and Reliability Indexes: Two New Measures to Enhance Multi-Image Face Recognition," Pattern Recognition 43 (2010) 3483–3493

[13] Weihong Deng A, Jianihua, Junguo A, Weidongcai B and Daganfeng, "Emulating Biological Strategies for Uncontrolled Face Recognition," Pattern Recognition 43 (2010) 2210–2223

[14] Xiaoyang Tan and Bill Triggs, "Enhanced Local Texture Feature Sets for Face Recognition under Difficult Lighting Conditions," IEEE Transactions On Image Processing, Vol. 19, No. 6, June 2010

[15] Zhiming Liu, Jian Yang and Chengjun Liu, "Extracting Multiple Features in the Cid Color Space for Face Recognition," IEEE Transactions on Image Processing, Vol. 19, No. 9, September 2010

[16] Zhen Lei, Shengcai Liao, Matti Pietikäinen and Stan Z. Li, "Face Recognition by Exploring Information Jointly in Space, Scale and Orientation," IEEE Transactions on Image Processing, Vol. 20, No. 1, January 2011

[17] Richard M. Jiang, Danny Crookes and Nie Luo' "Face Recognition in Global Harmonic Subspace," IEEE Transactions on Information Forensics and Security," Vol. 5, No. 3, September 2010

[18] Baochang Zhang, Yongsheng Gao, Sanqiang Zhao and Jianzhuang Liu, "Local Derivative Pattern Versus Local Binary Pattern: Face Recognition With High-Order Local Pattern Descriptor," IEEE Transactions on Image Processing, Vol. 19, No. 2, February 2010

[19] Gui-Fu Lu, Zhonglin and Zhongjin, "Face Recognition Using Discriminant Locality Preserving Projections Based on Maximum Margin Criterion," Pattern Recognition 43 (2010) 3572–3579

[20] Yong Wangn and Yiwu, "Face Recognition Using Intrinsic faces," Pattern Recognition 43 (2010) 3580–3590

[21] Saeed Dabbaghchian, Masoumeh P. Ghaemmaghami and Aliaghagolzadeh, "Feature Extraction Using Discrete Cosine Transform and Discrimination Power Analysis with a Face Recognition Technology," Pattern Recognition 43 (2010) 1431–1440

[22] Shufu Xie, Shiguang Shan, Xilin Chen and Jie Chen, "Fusing Local Patterns of Gabor Magnitude and Phase for Face Recognition," IEEE Transactions on Image Processing, Vol. 19, No. 5, May 2010

[23] Zhiming Liu and Chengjun Liu, "Fusion of Color, Local Spatial and Global Frequency Information for Face Recognition," Pattern Recognition 43 (2010) 2882–2890

[24] Christian Tenllado, Jose Ignacio Gomez, Javier Setoain, Dario Mora and Manuel Prieto, "Improving Face Recognition By Combination of Natural and Gabor Faces," Pattern Recognition Letters 31 (2010) 1453–1460

[25] Miao Cheng, Bin Fang, Yuan Yan Tang, Taiping Zhang and Jing Wen, “Incremental Embedding and Learning in the Local Discriminant Subspace with Application to Face Recognition,” IEEE Transactions on Systems, Man and Cybernetics—Part C: Applications And Reviews, Vol. 40, No. 5, September 2010

[26] Annalisa Franco, Dariomaio and Davide Maltoni, “Incremental Template Updating for Face Recognition in Home Environments,” Pattern Recognition 43 (2010) 2891–2903

[27] Imran Naseem, Roberto Togneri and Mohammed Bennamoun, “Linear Regression for Face Recognition,” IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 32, No. 11, November 2010

[28] Wen Chung Kao, Ming Chaihsu and Yueh Yiing Yang,” Local Contrast Enhancement and Adaptive Feature Extraction for Illumination-Invariant Face Recognition,” Pattern Recognition 43 (2010) 1736–1747

[29] Shuicheng Yan, Huan Wang, Jianzhuang Liu, Xiaoou Tang and Thomas S. Huang, “Misalignment-Robust Face Recognition,” IEEE Transactions on Image Processing, Vol. 19, No. 4, April 2010

[30] Ritwik Kumar, Angelos Barmpoutis, Arunava Banerjee and Baba C. Vemuri, “Non-Lambertian Reflectance Modeling and Shape Recovery of Faces Using Tensor Splines,” IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 33, No. 3, March 2011

[31] Tae Kyun Kim, Josef Kittler and Roberto Cipolla, “On-Line Learning of Mutually Orthogonal Subspaces for Face Recognition by Image Sets,” IEEE Transactions on Image Processing, Vol. 19, No. 4, April 2010

[32] Nes¸E Alyüz, Berk Gokberk and Lale Akarun, Regional Registration for Expression Resistant 3-D Face Recognition,” IEEE Transactions on Information Forensics and Security, Vol. 5, No. 3, September 2010

[33] Jiwen Lu and Yap Peng Tan, “Regularized Locality Preserving Projections and its Extensions for Face Recognition,” IEEE Transactions on Systems, Man and Cybernetics Part B: Cybernetics, Vol. 40, No. 3, June 2010

[34] Yueming Wang, Jianzhuang Liu and Xiaoou Tang, Robust 3D Face Recognition by Local Shape Difference Boosting,” IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 32, No. 10, October 2010

[35] Lishan Qiao, Songcan Chen and Xiaoyang Tan, “Sparsity Preserving Discriminant Analysis for Single Training Image Face Recognition,” Pattern Recognition Letters 31 (2010) 422–429

[36] Sang-Ki Kim, Youn Jung Park, Kar-Ann Toh and Sangyoun Lee, “SVM Based Feature Extraction for Face Recognition,” Pattern Recognition 43 (2010) 2871–2881

[37] Chuan-Xian Ren and Dao-Qing Dai, “Incremental learning of bidirectional principal components for face recognition,” Pattern Recognition 43 (2010) 318 – 330

[38] Yong Wang and YiWu, “Complete neighborhood preserving embedding for face recognition”, Pattern Recognition 43 (2010) 1008 -1015

[39] J. M. Combes, A. Grossman and P. Tchamitchan, “Wavelets, time frequency methods and phase space”, Springer-Verlag, 1989.

[40] A. Grossman and R. Kronland Martinet, “Time and scale representation obtained through continuous wavelet transform”, Signal Processing IV, Theories and Applications, vol. Elsevier Science Pub. B. V. 1988, pp. 475-482.

[41] S. G. Mallat, “A theory for multiresolution signal decomposition”, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 11 (7), Jul. 1989.

[42] Rajiv Kapoor and Amit Dhamija, “A New Method for Identification of Partially Similar Indian Scripts”, International Journal of Image Processing (IJIP), Volume (6): Issue (2): 2012

[43] M. Devlin, T. S. Durrani, "Wavelants - An introduction to higher order wavelets"

[44] Paraskevi Tzouveli, Klimis Ntalianis and Stefanos Kollias, "Video object watermarking using Hu moments," 13th International Conference on Systems, Signals and Image Processing (IWSSIP), Budapest, Hungary ,6 pg, 21-23 September 2006

[45] Hasan Abdel Qader, Abdul Rahman Ramli and Syed Al-Haddad, "Fingerprint recognition using Zernike moments." The International Arab journal of Information Technology, Volume 4, Number 4, October 2007

[46] Simon Haykin , "Adaptive Filter Theory", (4th Edition), Publisher: Prentice Hall, Published: 2001

Appendix A

Abbreviations

1) ANN - Artificial Neural Network
2) BLD - Bayesian Logistic Discriminant
3) HMM - Hidden Markov Model
4) KNN - K Nearest Neighbor
5) LDA - Linear Discriminant Analysis
6) ORL - Olivetty Research Laboratories
7) RGB - Red Green Blue
8) SVM - Support Vector Machine

Appendix B

An Introduction to Image Processing in Matlab

MATLAB is a high performance language for technical computing. It integrates computation, visualization and programming in an easy-to-use environment where problems and solutions are expresses in familiar mathematical notation. It is an interactive system whose basic data element is an array that does not require dimensioning. This allows you to solve many technical computing problems, especially those with matrix and vector formulations in a fraction of the time it would take to write a program in a scalar non interactive language such as C, C++ or JAVA.

Image formats supported by Matlab

The following image formats are supported by Matlab:

- BMP
- HDF
- JPEG
- PCX
- TIFF
- XWB

Working formats in Matlab

If an image is stored as a JPEG-image on your disc we first read it into Matlab. However, in order to start working with an image, for example perform a wavelet transform on the image, we must convert it into a different format.

Intensity image (gray scale image)

It represents an image as a matrix where every element has a value corresponding to how

bright/dark the pixel at the corresponding position should be colored. There are two ways to represent the number that represents the brightness of the pixel: The double class (or data type). This assigns a floating number ("a number with decimals") between 0 and 1 to each pixel. The value 0 corresponds to black and the value 1 corresponds to white. The other class is called uint8 which assigns an integer between 0 and 255 to represent the brightness of a pixel.

Binary image

This image format also stores an image as a matrix but can only color a pixel black or white (and nothing in between). It assigns a 0 for black and a 1 for white.

Indexed image

This is a practical way of representing color images. An indexed image stores an image as two matrices. The first matrix has the same size as the image and one number for each pixel. The second matrix is called the color map and its size may be different from the image. The numbers in the first matrix is an instruction of what number to use in the color map matrix.

RGB image

This is another format for color images. It represents an image with three matrices of sizes matching the image format. Each matrix corresponds to one of the colors red, green or blue and gives an instruction of how much of each of these colors a certain pixel should use.

Reading Image Files

The command to read an image from file filename and store it in matrix variable p is:

```
p = imread('filename');
```

The filename may be an absolute pathname or a relative pathname from the current working directory. Omitting the ';' at the end of the command causes the value to be printed to the command window.

Writing Image Files

The command to write an image from variable p and store it in file filename is:

imwrite (p, 'filename');

The filename may be an absolute pathname or a relative pathname from the current working directory.

Displaying Image Files

The command to display the image from variable p to a figure window is:

imshow(p);

An additional parameter may be used to set the number of display levels or set the range of display levels. Various controls, such as dynamic display of index and value for the cursor position, are available in the image display tool.

The figure command can be used to create a new current figure for the display:

figure, imshow (p);

Standard Arrays

MATLAB has standard arrays for creating arrays of a defined size with zeros, ones, true, false, or random values. For example: p = zeros(M,N);

Major built in Functions (used in this work) are:

1) **Classify**

Its format is:

class = classify (sample, training, group)

It classifies each row of the data in sample into one of the groups in training. Group is a grouping variable for training. ‘Class’ indicates which group each row of SAMPLE has been assigned to, and is of the same type as ‘group’.

2) Factorial

Its format is:

factorial(n)

This command calculates the factorial on ‘n’ which is the product of all the integers from 1 to n.

3) Bispeci

It calculates the bispectrum using the indirect method. This command is in ‘HOSA’ toolbox.

Its format is:

[Bspec,waxis] = bispeci (y,nlag,segsamp,overlap,flag,nfft, wind)

Here, ‘y’ is the data vector or time-series, ‘nlag’ is the number of lags to compute, ‘segsamp’ is the number of samples per segment, ‘overlap’ is the percentage of overlap, ‘flag’ stands for ‘biased' or 'unbiased' , ‘ nfft’ is the FFT length to use, ‘wind’ is the window function to apply.

The output ‘Bspec’ is the estimated bispectrum and it is an nfft x nfft array with origin at the center, and axes pointing down and to the right. ‘Waxis’ is the frequency-domain axis associated with the bispectrum.

4) Size

Its format is:

[M, N] = SIZE(X)

It calculates the size of matrix X and returns the number of rows and columns in X as separate output variables.

5) Sort

Its format is:

Y = SORT(X, DIM, MODE)

This command sorts in ascending or descending order. ‘Dim’ selects a dimension along which to sort. ‘Mode’ selects the direction of the sort, 'ascend' results in ascending order and 'descend' results in descending order. The result is in Y which has the same shape and type as X.