2-1-2014

Illumination tolerance in facial recognition

Aishat Dan-Ali Mahmoud

The American University in Cairo

School of Sciences and Engineering

ILLUMINATION TOLERANCE IN FACIAL RECOGNITION

A Thesis Submitted to
The Department of Computer Science

in partial fulfillment of the requirements for
the degree of Master of Science (Computer Science)

By

Aishat Mahmoud Dan Ali

Under the supervision of

Dr. Mohamed N. Moustafa.

DEDICATION

I hereby dedicate this work to my beloved husband, whose moral support and understanding make everything possible and worthwhile.

And to my two lovely children whose love and patience kept me going even in tough times.

Finally, I dedicated this work to my family, for all the prayers and support.

ACKNOWLEDGEMENT

First of all, I would like to express my profound gratitude and appreciation to my supervisor, Dr. Mohamed N. Moustafa. For his sincere devotion, his constant encouragement and tolerance to the plight of his students, this makes working with him a priceless experience. This thesis is as a result of his constant support and constructive criticism.

My sincere appreciation and kindest regards goes to the administration of Umar Musa Yaradua University (UMYU), katsina- Nigeria, for their financial support throughout my postgraduate years. My special gratitude goes to DVC admin Prof. Sada Abdullahi for his constant academic and moral support.

My earnest gratitude also goes to the HOD Computer Science and Mathematics UMYU, the dean Faculty of Natural and Applied Sciences UMYU, the DVC academic UMYU for their various support and encouragement.

Finally, my heartfelt and warmest regards goes to my family members and many friends for their constant prayers and moral support. Above all, I thank almighty Allah for making this journey possible and fruitful.

ABSTRACT

The American University in Cairo
School of Sciences and Engineering

ILLUMINATION-TOLERANT FACE RECOGNITION SYSTEM

Aishat Mahmoud Dan Ali

Supervision: Dr. Mohamed N. Moustafa.

In this research work, five different preprocessing techniques were experimented with two different classifiers to find the best match for preprocessor + classifier combination to built an illumination tolerant face recognition system. Hence, a face recognition system is proposed based on illumination normalization techniques and linear subspace model using two distance metrics on three challenging, yet interesting databases. The databases are CAS PEAL database, the Extended Yale B database, and the AT&T database. The research takes the form of experimentation and analysis in which five illumination normalization techniques were compared and analyzed using two different distance metrics. The performances and execution times of the various techniques were recorded and measured for accuracy and efficiency. The illumination normalization techniques were Gamma Intensity Correction (GIC), discrete Cosine Transform (DCT), Histogram Remapping using Normal distribution (HRN), Histogram Remapping using Log-normal distribution (HRL), and Anisotropic Smoothing technique (AS). The linear subspace models utilized were principal component analysis (PCA) and Linear Discriminant Analysis (LDA). The two distance metrics were Euclidean and Cosine distance. The result showed that for databases with both illumination (shadows), and lighting (over-exposure) variations like the CAS PEAL database the Histogram remapping technique with normal distribution produced excellent result when the cosine distance is used as the classifier. The result indicated 65% recognition rate in 15.8 ms/img. Alternatively for databases consisting of pure illumination variation, like the extended Yale B

database, the Gamma Intensity Correction (GIC) merged with the Euclidean distance metric gave the most accurate result with 95.4% recognition accuracy in 1ms/img. It was further gathered from the set of experiments that the cosine distance produces more accurate result compared to the Euclidean distance metric. However the Euclidean distance is faster than the cosine distance in all the experiments conducted.

TABLE OF CONTENTS

LIST OF TABLES

TABLE NO. TITLE. PAGE NO.

LIST OF FIGURES

FIGURE NO. TITLE. PAGE NO.

LIST OF ABBREVIATIONS

Abbreviation	**Meaning**
2D Gabor	Two Dimensional Gabor
3D	Three Dimensional
AS	Anisotropic Smoothing
AT&T	American Telephone & Telegraph
CAS PEAL	Chinese Academic of Science Pose Expression, Accessories and Lighting
CMC	Cumulative Match Curve
DCT	Discrete Cosine Transform
EPC	Expected Performance Curve
FERET	Face Recognition Technology
FR	Face Recognition
GIC	Gamma Intensity Correction
HE	Histogram Equalization
HRL	Histogram Remapping using Log normal distribution
HRN	Histogram Remapping Using Normal Distribution
ICA	Independent Component Analysis
KFA	Kernel Fisher Analysis
KPCA	Kernel Principal Component Analysis
LBP	Local Binary Pattern
LDA	Linear Discriminant Analysis
LEMs	Line Edge Maps
LTP	Local Ternary Pattern

MATLAB	Matrix Laboratory
NIR	Near Infra Red
NN	Nearest Neighbor
PCA	Principal Component Analysis
PhD	Pretty Helpful Device
ROC	Receiver Operating Characteristic
SFS	Statistical Shape From Shading

CHAPTER 1

INTRODUCTION

In the last 30 years much interest, time, energy and research have been ventured into the field of face recognition. These resulted in the development of vast computer algorithms and technologies that tries to solve the task of robust face recognition. The resulting outcome was numerous face recognition systems that perform well in a controlled environment where lightning and other challenges were not a problem. The task that remain to be solved is face recognition system that meets all the challenges; these challenges includes lighting or illumination challenges, pose, expression, accessories and others as would be enumerated in the subsequent subsection.

Another reason for the boost in face recognition and detection systems lies in the academic domain because of interest for building fast and efficient algorithms that perform the tasks, for instance, following the wake of Face Recognition Grand Challenge FRGC (1996) more fast and efficient face recognition systems have been built ever since. Similarly, the need for efficient and robust face recognition system arises due to security and surveillance reasons, as a result more fast and accurate systems are needed in airports, government buildings, and commercial areas to cater for the growing number of population and crime. Therefore, much effort is made towards building effective and robust face recognitions systems that meet these challenges. These developed systems play a vital role in today's security measures taken by various government agencies and commercial enterprises alike.

The research work carried out here focuses on the preprocessing stage in designing a system of face recognition based on various preprocessing techniques to alleviate the effect of lighting and illumination. This is achieved by trying to find the best possible match between the five (5) processing methods and the (2) classifiers. The proposed system comprises of the preprocessing stage and then the PCA/LDA subspace model to overcome the effect of illumination and produce a robust face recognition system.

This chapter is structured as follows: section 1.1 gives introduction to face recognition systems in general and illumination invariant face recognition systems in particular. Section 1.2 introduces the research background of face recognition systems. In section 1.3 a formal definition of the problem is given. Aims and objectives of the study are highlighted in section 1.4. Section 1.5 states the scope of the study. In section 1.6, an outline of the thesis is given, while section 1.7 summarizes the chapter.

1.1 Introduction to Face Recognition Systems

Face recognition systems are systems that are designed to recognize a given input face image from previously known database of faces. If the given input image (the probe) is present in the database, the system returns the matching image, otherwise it returns failure.

Face recognition is one of the branches or applications of Pattern Recognition that deals with capture, analysis and identification of human faces. Other application areas of pattern recognition include speech recognition, character (letter/number) recognition (OCR), and computer aided diagnosis [29], among others. Face recognition as one of the techniques in face processing, is related in part to image processing, image analysis, and computer vision. Depending on the context, these different approaches are sometimes considered as one while sometimes different. A definition of the terms (though not universally accepted) is:

Face recognition, in its simplest form, is the process of comparing a test image to a database of images to determine if there is a match and return it. Face recognition is one of the successful applications of image analysis and understanding [1].

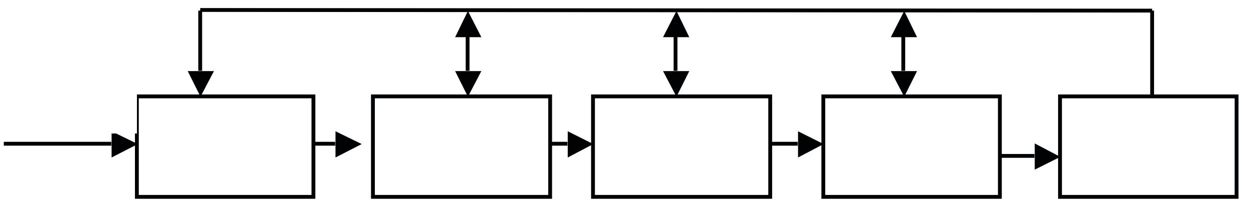


Fig. 1.1 Block diagram of a pattern recognition system.

While Image Processing and Image Analysis are methods of transforming 2-dimensinal images into another by applying processes to the image such as contrast enhancement, edge detection/extraction, noise removal, or geometrical transformations such as rotating the image. Therefore, this shows that Image processing/analysis, does not produce interpretations nor require assumptions about the image content.

Computer vision is a field that is concerned with methods for acquiring, processing, analyzing, and understanding 3-dimensional images from 2-dimensional images so as to produce numerical or symbolic information, necessary for making decisions.

Face processing techniques are those processing techniques that utilize human face to carry out some important processes and transformations to the face. These processes include face detection, face localization, face recognition, face identification, face verification, face authentication, face tracking, facial expression recognition, similarity/kinship recognition, and facial feature extraction, among others. The figure below shows the relationship between various fields of pattern recognition and machine learning.

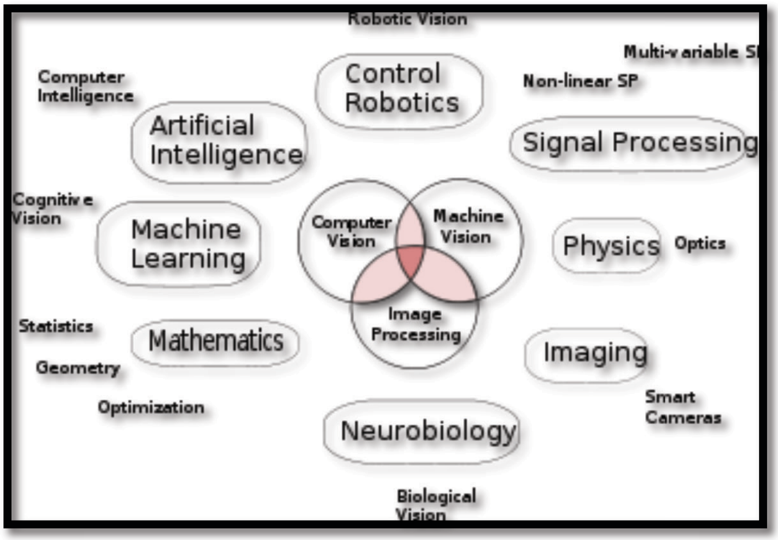


Fig1.2 Relationships between computer vision, image processing, and various other fields (source: wikipedia)

Face recognition has applications mainly in the fields of biometrics, access control, law enforcement, security, and surveillance systems [30]. Biometrics are technological methods that capture, measure and analyze human body characteristics automatically verifying or identifying an individual's physiological or behavioral traits. Biometric technologies proposed for authentication purposes include [30],[31]:

- The *DNA Sequence matching* is the best biometric as it is very invariant to any factor of change. DNA is a unique sequence of code for each individual. DNA matching is used mostly in forensic applications and it is not useful in automatic real-time recognition applications.

- The *signature recognition* has been widely used and accepted biometric as a verification protocol. Nevertheless the signature could be affected by physical and emotional status of a subject and it can be changed over a period of time. Moreover the signature is susceptible to fraud and imitation by other party.

- The *fingerprint recognition* has been the major source of biometric technique in previous decades; it has a very high matching accuracy within a reasonable price, but one of its drawbacks is that fingerprint of a person can get damaged by cuts or burning thereby rendering the biometric useless, more also, building the system requires large amount of computational resources.

- The *hand geometry* recognition system is one of the earliest automated biometric systems. It is very simple to implement, easy to use and relatively cheap, however, the hand geometry is not very characteristic. It can be used in verification mode.

- The *iris recognition* is the process of measuring and matching the annular region of the eye bounded by the pupil and the sclera -the white of the eye- known as the Iris. The texture of the iris provides very useful information for recognition. Considerable user participation was required in the early iris-based recognition systems, and the systems were pretty expensive, but the newer systems have become more user-friendly and cost-effective [31]. However the system is intrusive and data collection can become tedious.

- The *infrared thermogram* of facial vein, and hand vein is often used as a biometric technique. These can be captured by infrared cameras, like face recognition, it is not an intrusive method, but on the other hand, image acquisition is fairly difficult and setting up the system is quite expensive.

- The *ear* recognition is based on measuring and comparing the distances of significant points on the pinna. However, this biometric technique is not very effective in establishing the identity of a user.

- The *retinal scan* is one of the most secure biometric since it is not easy to change or replicate. The retina possess unique characteristic of each individual and for each eye. High cooperation of the user is required during acquisition and the user need to use eyepiece and focus on a specific spot so that a predetermined part of the retinal vasculature can be captured. Consequently, these factors can affect the public acceptability of retinal biometric.

- The *face* is probably one of the most common and suitable biometric characteristics ever used. The system is convenient and data collection can be done in a passive, non intrusive manner.

- The *voice recognition* is seldom used and it is not very distinctive and it changes a lot over a period of time. Moreover, the voice is not useful in large-scale identification.

Among all these biometric techniques, Face Recognition is the most feasible because the face recognition system has better advantages like the face is always available (i.e. it cannot be forgotten, stolen or misplaced), the system does not pose any health hazard to the subject, and it does not require the full cooperation of the subject to gather data while the other systems cannot be constructed without the full consent of the subject.

1.2 Challenges in Face Recognition

Many sources of inconsistency could be encountered when dealing with images in a face recognition system. The major challenges found in human face recognition are listed in this section. There are many challenges cited in the literature such as [30] and [32] that provides most of the challenges commonly found in designing face recognition systems. The major challenges are here-by given the acronym ASPIRE.

- *Accessories and Facial hair*: Difference in facial hair, and accessories, like eye glasses or scarf between the training samples and the test image can result in difficult classification.

- *Aging*: Prolong interval between training set and query set - for instance, images taken in one session and another taken after 10 years drastically changes the accuracy of the system.

- *Size of the image*: If a test image is much smaller in dimension – say, of size 10x10 –than the training set of larger dimensionality (100 x 100) then it may be hard to classify.

- *Pose*: Frontal profile always gives a better classification. The angle in which the photo of the individual was taken with respect to the camera changes the system accuracy.

- *Illumination*: The variations due to illumination and viewing direction between the images of the same face are almost always larger than image variations due to change

Fig.1. 3 Sample images from the PIE database showing different variations in Expression, Lighting (Illumination) Accessory and Pose.

in face identity [8]. The direction of illumination greatly effects face recognition success.

- *Rotation*: Rotation of the individual's head clockwise or counter clockwise even if the image stays frontal with respect to the camera affects the performance of the system. There is in-plane and out-of -plane rotation.

- *Expression*: Different facial expression can affect facial recognition system significantly. Examples of facial expression are neutral face (no expression), closed eyes, laughing, screaming, *etc*.

1.3 Introduction To Research Background

Research into face recognition has been carried out for the past three decades, as highlighted earlier. Many systems were proposed using different algorithms and designs that try to solve the problem of face recognition. None the less, the problem is far from being a solved one [1]. One of the major challenges affecting the robustness of the existing systems is that of lighting and or illumination. Lately, attention is focused on building illumination –invariant face recognition systems that works well in the presence of lighting variation, but most of these systems doesn't work well in extreme illumination conditions.

To curb the effect of variable illumination problem, many approaches have been proposed which can be broadly classified into three main categories: (1) Invariant Features Extraction, (2) Normalization and Preprocessing, and (3) Face Modeling [4].

According to the literature, general face recognition algorithms are broadly divided into two classes, the first group is termed global approach or appearance-based while the second group is termed feature-based or component-based [2]. In the first category holistic texture features are extracted and it is applied to the face or specific region of it. Whereas in the second category, the geometric relationships between the facial features like eyes, nose, and mouth are utilized [2]. Appearance-based approach includes Principal Component Analysis- PCA and Linear Discriminant Analysis- LDA, which provides much better result in terms of performance and ease of usage. Face recognition algorithms try to solve the problem of both verification and identification [1]. When verification is needed, the face recognition system is given a face image and its claimed identity. The system is expected to either reject or accept the claim. Whereas, the identification problem is defined as: given a test image to the system which is initially trained with some images of known individuals; decides which individual the test image belongs to.

1.4 Problem Definition

The problem of face recognition can be stated as follows: Given the gallery set which comprises of set of face images labeled with the person's identity and the query set which comprises of an unlabeled set of face images from the same group of people, the task is to identify each person in the query images. The first step requires the face be located in the image; this process is known as face detection - which is not the concern of this work. The second step involves extraction of a collection of descriptive measurements known as a feature vector from each image. In the third step, a classifier is trained to assign to each feature vector a label with a person's identity. Actually, the classifiers are simply mathematical functions which return an index corresponding to a subject's identity when given a feature vector. The problem described above is already trivially solved using numerous methods and algorithms under normal standard conditions (i.e., frontal profile, no illumination, pose or rotation). The problem that needed to be solved here is designing such a system that is invariant to illumination and facial expression. Therefore, the problem being tackled in this thesis work is finding the right preprocessing technique, i.e. the best possible match between preprocessing method and classifier, and designing such a system that is invariant to illumination and mild facial expression for accurate and illumination invariant face recognition.

1.5 Aims and objectives of the study

The aim and objective of the task here at hand is to find the best possible match between different preprocessing methods and two different classifiers to design an efficient FR system that performs well in the presence of extreme illumination condition and mild facial expressions. The method proposed in this research work is a system of face recognition consisting of special preprocessing techniques to tackle the effect of variation due to illumination using a confusion of preprocessing chain and Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA) and

Euclidean or Cosine classification using nearest neighbor (NN) algorithms. The sub-objectives needed to accomplish the main objectives of the research are:

- To experiment with different combination of preprocessing techniques and classifiers to find the right one for face recognition systems.
- To apply the sequence of the chosen preprocessing techniques that eliminate or minimize the effect of illumination variation and mild facial expression challenges in face recognition.
- To develop an FR system that correctly performs the task of face recognition in the presence of illumination and facial expression challenges.
- To build a fast and efficient FR system that meets expectations.

1.6 Scope of the study

The main purpose of this research work is to investigate the appropriate preprocessing technique plus the classification algorithm for illumination tolerant face recognition and to design a face recognition system using image preprocessing techniques, and PCA/LDA algorithms that is tolerant of extreme illumination changes and mild facial expressions variations. In all the databases used, only frontal and near frontal images are included in this research, and the system will only tackle illumination and mild expression variation. The system does not try to overcome other challenges such as ageing, pose and accessories, etc.

The major steps in developing the system include:

- Preprocessing steps
- Dimension reduction using PCA,
- Feature extraction using LDA, and
- Classification using nearest neighbor classifier.

Combination of PCA and LDA is used for improving the capability of LDA when a few samples of images are available and nearest neighbor classifier is a method for classifying objects based on closest training examples in the feature space.

PCA is normally used in face recognition system for dimensionality reduction. This is done by extracting the most significant features from original face images which spans over high dimension. It captures the variance between training samples and turns them into a small set of characteristic feature images called principal components or “eigenfaces” [3].

LDA is the projection that best separates the data in a least-square sense. It uses face class information to find a subspace for better discrimination of different face classes. Essentially, LDA tries to find the best direction of projection in which training samples belonging to different classes are best separated.

Nearest neighbor classification (NN): is a method for classifying objects based on closest training examples in the feature space [27]. NN is a type of instance-based learning, the function is only approximated locally and all computation is deferred until classification.

1.7 Development tools

The following packages were used for developing the proposed face recognition system. In some cases only the platform is used (Matlab) and in others the algorithms presented were modified to suit the needs of the program.

1.7.1 Matlab

Matlab is a collection of software packages that is designed for easy computation, visualization, and programming [59]. It was created in 1984 by *The MathWorks Inc*. Matlab is a language for technical computation where problems and solutions are expressed in familiar mathematical notation with high efficiency and high performance.

The uses of Matlab includes: Math and computation, Algorithm development, Data acquisition, Modeling, simulation, and prototyping, Data analysis, exploration, and visualization, Scientific and engineering graphics, Application development, including tools for building graphical user interface. MATLAB can be described as an environment for numerical computations and a programming language. Its capability for easy usage makes it popular for matrix manipulations implementations of algorithms, plotting of graphs, creations of GUIs, and interfacing with programs in other languages.

Matlab has wide range application toolboxes such as Aero toolbox, Bioinfo toolbox, Neural Network toolbox, Images toolbox, Image processing toolbox, signal processing toolbox, Fuzzy toolbox, finance toolbox and many more. These toolboxes allow users to perform various computations and simulations in their respective fields. Matlab has been a defector instrument for instruction in various Universities for various courses such as Mathematics, Engineering and Science both for introductory and advanced students.

1.7.2 The PhD Face Recognition Toolbox

The PhD (Pretty helpful Development functions for) face recognition toolbox is a collection of Matlab functions and scripts intended to help researchers working in the field of face recognition. The toolbox was produced by Struct [22, 23] as a byproduct of his research work and is freely available for download.

The PhD face recognition toolbox includes implementations of some of the most popular face recognition techniques, such as Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), Kernel Principal Component Analysis (KPCA), Kernel Fisher Analysis (KFA). It features functions for Gabor filter construction, Gabor filtering, and all other tools necessary for building Gabor-based face recognition techniques.

In addition to the listed techniques there are also a number of evaluation tools available in the toolbox, which makes it easy to construct performance curves and performance metrics for the face recognition one is currently assessing. These tools allows the user to compute ROC (Receiver Operating Characteristics) curves, EPC (Expected performance curves), and CMC (cumulative match score curves) curves.

1.8 Outline of the Thesis

This thesis is organized into seven (7) chapters and an appendix.

Chapter 1 provides general introduction to the research work and its sequence of execution.

Chapter 2 gives overview of the previous literature on the study of face recognition systems in general and illumination invariant face recognition systems in particular.

Chapter 3 explains the image preprocessing techniques for face recognition that would to be experimented with. The techniques include gamma intensity correction, discrete cosine transform, histogram remapping techniques, and anisotropic smoothing method.

In **chapter 4** details of the linear subspace models PCA and LDA are tabled out. These models were used after the preprocessing stage.

Chapter 5 elaborates on the various experiments carried out. That is the various preprocessing/illumination normalization techniques and the two different distance metrics. The chapter presents the databases used, experimental set-up and results.

In Chapter 6 analyses and discussions on the various results were made.

Finally, **Chapter 7** gives summary, conclusion of the research work, Recommendation and suggestion for future work.

CHAPTER 2

LITERATURE REVIEW

2.0 Introduction

This chapter contains a general overview of the existing methods for face recognition algorithms under the subfield of illumination normalization. This survey analyzes and compares some of the approaches to facial recognition algorithms of 2- dimensional static images that have been done by various researchers under different sub categories of illumination –normalization / compensation or elimination, for the past three decades.

2.1 Techniques for Normalization of Illumination Variation

Many algorithms have been proposed in the previous decades that tackle the task of face recognition. Among all the challenges of face recognition mentioned above, illumination together with pose variation are the most challenging and the ones that have recently been given more thought and research. In an attempt to solve the problem of robust face recognition in out-door (uncontrolled) environment, many researchers have tried to develop face recognition (FR) techniques that are tolerant to image deterioration caused by many factors, such as camera resolution, background influence, and natural lighting. The following survey highlighted recent literature in the area.

A recent survey by K. R Singh et al [4] highlights the major challenges in face recognition techniques and cited the illumination condition together with the pose variation as the most critical. Various researches have tried to categorize the techniques for overcoming the effect of lighting on face recognition. They come up with slightly different classifications based on addition of some techniques in the list or considering two techniques as one. According to the survey in [4] and other surveys such as [33] and [34], the techniques for overcoming the illumination challenge can be broadly categorized as:

1 Transformation of images into a canonical representation; Modeling of illumination variation

2 Preprocessing and Photometric normalization

3 Extraction of illumination invariant features

4 Utilization of 3-d morphable models

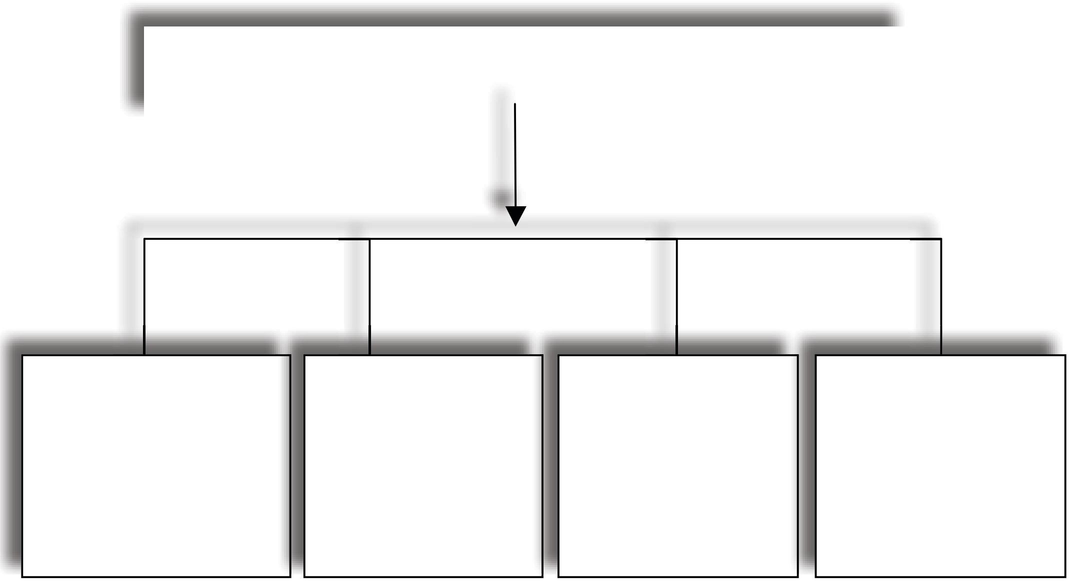


Fig. 2.1 Block diagram of lighting variation normalization techniques

2.1.1 Transformation of images into a canonical representation

Transformation of images into a canonical representation was one of the first attempts to alter the effect of illumination from images. Following the introduction of principal component analysis in the 1980's and their use for face recognition by Turk and Pentland [3] in the 1990's, a lot of variations of the eigenface technique have been suggested by many researchers such as in [2][5][22][23] to tackle the effect of illumination variation in images. The Eigenface technique has been comprehensively utilized for the sole aim of face detection and recognition and now for the purpose of illumination normalization in face recognition systems. Zhao and Chellappa [5] proved that the effectiveness PCA and LDA algorithms were significantly improved by using prototype images and by combining the symmetric SFS algorithm and a generic 3D head model. They produced an improved face recognition technique under varying illumination condition.

Similarly, Belhumeur *et al.* [6] noted that the first three (3) principal components in PCA algorithm only capture lighting variations; therefore, they modified the PCA by discarding the first three principal components. Consequently, they achieved better performance for images under different illumination variations. However, the drawback of this method is that some of the discarded principal components can influence the face recognition under normal illumination conditions. In a related work, Bartlett *et al.* [7] used a version of ICA - which is a generalization of PCA- that is derived from the principle of optimal information transfer through sigmoidal neurons. They achieved recommended performance on the FERET database. However, most of these works presented here does not work effectively in the presence of complex illumination conditions.

2.1.2 Modeling of Illumination Variation

This approach is similar to the appearance-based method. The main difference is that only a small number of training images are required to create new images under changes in illumination direction. Techniques in this category can be further divided into statistical model and physical model Zou *et al*. [33]. Statistical approaches include applying customized PCA and LDA algorithms to the images to alter the effect of illumination variation. While in physical modeling basic assumption about the formation of images is based on the properties of the object's surface reflectance, for instance, the Lambertian reflectance. The statistical approaches to modeling face image can be further classified as linear subspaces, spherical harmonics, nine point lights and generalized photometric [33]. These subdivisions are highlighted below.

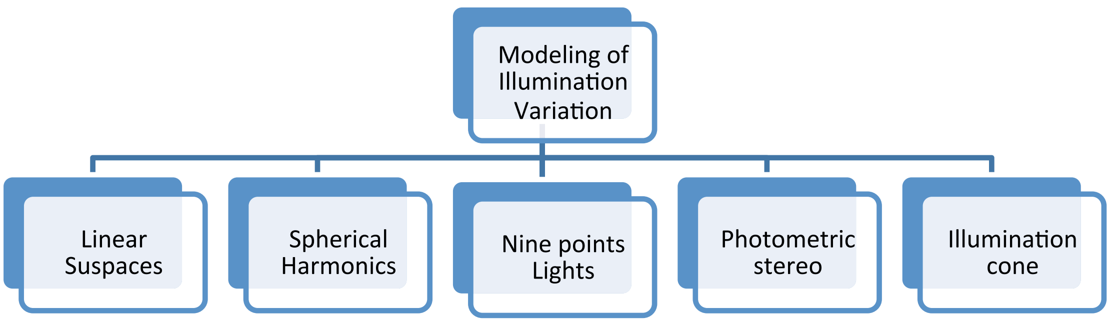


Fig. 2.2 Various techniques for modeling of illumination variation

2.1.2.1 Linear subspaces:

In this subcategory low dimensional linear subspaces are used for modeling facial images under various illumination conditions. For instance, 3D linear subspace method was presented by Belhumeur *et al.* [6] for illumination invariant face recognition. Their method includes making use of three or more images of the same face taken under different lighting to construct a 3D basis for the linear subspace. Recognition is done by comparing the distance between the test image and each linear subspace of the faces belonging to each identity. In another work, Belhumeur *et al.* [8] make use of a single image to construct virtual eigenspace, however, the real eigenspace cannot be constructed directly from a single image. They reported considerable improvement in recognition rate.

Hallinan [35] proposed a model that can handle non-lambertian and self shadowing surface such as the face and showed that five eigenfaces were adequate for representing the face images under a broad range of lighting conditions. Batur and Hayes [36] performed a k-means clustering on segmented linear subspace model. The model generalizes the 3D illumination linear model and is robust to shadow areas in the image. Images are segmented according to areas with similar surface normals. Recognition is done by calculating the minimum distance between the image and the illumination subspaces of the objects in the training select.

2.1.2.2 Spherical harmonics

Spherical Harmonics technique is performed by analyzing the best subspace that approximate the convex Lambertian reflection properties of the object taken from a fixed viewpoint, but under varying distant illumination conditions. This method was first proposed by Basri and Jacobs [38], and later by Ramamoorthi and Hanrahan [39]. Basri and Jacobs [37] Assumed an arbitrary point or diffuse light sources that are distant from an object of Lambertian reflectance property and shows that based on a Spherical Harmonic representation the intensity of object surface can be approximated by a 9-dimensional linear subspace ignoring cast shadow. Principal Component Analysis (PCA) is applied, and a low-dimensional approximation of illumination cones is obtained.

Alternatively, Zhang and Samaras [41] showed that only one image and no knowledge of 3D information is required to recognize faces under different lighting condition by using the spherical harmonics representation. Collections of 2D basis image vectors are used to build a statistical model of spherical harmonics in their first method. While in their second method [42], they combined a 3D morphable model and the harmonic representation to perform face recognition with both illumination and pose variation. Recognition of a face is based on a weighted combination of basis images that is the closest to the test face image.

2.1.2.3 Nine Points Lights

This is special configuration for the direction of light source in which nine (9) point sources of lights are arranged in a particular way and an image is captured with each light source on and the others off, making a total of nine different images with same pose and same facial expression but with different sources of light. This work was pioneered by Lee *at al.* [55], [52] and they showed that the subspace that resulted from these nine images is sufficient for recognition under different illumination conditions. Moreover, this technique has the advantage that no 3D information of the surface is needed to construct the model compared to the Spherical harmonics approach and there is no need for large data collection.

2.1.2.4 Generalized Photometric stereo

Photometric stereo is process of recovering the surface normal and albedo using 3 images that lie in 3D linear subspace of high dimensional image space of known linearly independent light sources. This work was proposed by shashua [53] and he claimed that there is no adverse effect caused by the attached shadows on the scheme. Additionally, a recent technique called generalized photometric stereo was proposed by Zhou et al. [54]. They use both the Lambertian reflectance model and the linear subspace model for analyzing the images of the face class. Taking the human face as a Linear Lambertian Object which is an object with Lambertian surface and having a collection of objects with lambertian surfaces, they try to recover the albedo and surface normal vectors of each basis object for the face class from a matrix called class-specific albedo/shape matrix using the Generalized Photometric Stereo process. The authors use the bootstrap set using Vetter's 3D face database [54]. They reported excellent performance from the trained model.

2.1.2.5 Illumination Cone

Another low dimensional linear subspace model for illumination invariant face recognition is the Illumination Cone method. Georghiades, Kriegman and Belhumeur [40] Shows that the set of images of an object under all possible illumination conditions in fixed pose, forms a convex cone in the images' space. Furthermore, they showed that three properly chosen images using different lighting directions can be used to construct the cone for a particular object (a face) using the shape and albedo of the images whose basis vectors are estimated using a generative model. Recognition is performed by assigning to a test image the identity of the closest illumination cone. When the system is presented with an image with a side pose, the models can be used to warp the image back into canonical profile form and the right lighting condition.

2.1.3 Extracting illumination invariant features

Features that are invariant to illumination changes also played an important role in illumination invariant face recognition. Other researchers concentrated thoroughly on extracting features that are invariant to changes in the direction of light. This has been achieved by extracting only those features that are not affected by variations in lighting conditions. Some of the representations under this technique comprises of features from image derivatives e.g. gradient faces [12], convolution of images with 2D Gabor Filters [13][14][22],[23], LBP and LTP Feature Extraction [15], [28], and Discrete Cosine Transform (DCT) Coefficients [10, 11].

2.1.3.1. Gradient faces:

One such method of extracting illumination insensitive features for face recognition under varying illumination is Gradient faces. It is derived from the image gradient domain such that it can discover underlying inherent structure of face images since the gradient domain explicitly considers the relationships between neighboring pixel points. Gradient face method is able to apply directly to only a single face image and it does not require any prior information or many training images. Moreover, Gradient face technique has low computational cost such that it can be applied to practical applications.

Another feature extraction method is the LEMs. Gao and Leung [19] proposed a novel face representation called Line Edge Maps LEMs. LEMS was an extension of simple edge map technique. In this technique of deriving features from image derivatives, the authors used the Sobel operator to extract the edge pixels of each image and grouped them into line segments which made up the LEMs. They claimed that the LEM face representation is invariant to illumination and expression changes. However, the performance of the gradient based Sobel operator used in the developed face recognition system deteriorates under extreme lighting conditions.

Chen *et al.* [11] proved that for objects with Lambertian surface there are no discriminative functions that are invariant to illumination. Consequently, they showed that the probability distribution of the image gradient is a function of the surface geometry and reflectance, which are the intrinsic properties of the face. They discovered that the direction of image gradient is insensitive to change in illumination. Similarly, Wei and Lai [43] and Yang *et al.* [44] applied Relative Image Gradient magnitude for robust face recognition under lighting variation. They used iterative optimization procedures for precise face matching using the relative image gradient. Symmetric Shape from Shading is a method presented by Zhao and Chellappa [7] for illumination insensitive face recognition. They make use of the symmetry of every face and the shape similarity among all faces. They use a single training image that is obtained under arbitrary illumination condition to obtain a prototype image with normalized illumination. The performance of the PCA and LDA based face recognition was considerably improved using this prototype image technique.

A statistical shape from shading (SFS) model was developed by Sim and Kanade [45] to recover face shape from a single image and to synthesize the same face under new illumination. The surface radiance for a particular position is modeled as the image's surface normal and albedo multiply by the light source vector plus an error term *e* which models shadows and spectacular reflections. For training the statistical model, a bootstrap set of faces with labeled different illuminations is needed for obtaining the surface normal and albedo and the error term *e*. They used kernel regression based on the bootstrap set to estimate the illumination for an input image subsequently, the surface normal and albedo can be obtained by Maximum *a* Posterior estimation and the input face under a new illumination can be synthesized.

2.1.3.2 DCT Coefficients

Chen *et al.* [11] employed the Discrete Cosine Transform (DCT) in the logarithmic domain for compensating illumination variations. The basic idea in this technique is that illumination variations generally lie in the low-frequency band, therefore, by truncating or discarding these DCT coefficients that corresponds to low level frequencies better recognition is achieved. Very encouraging results were obtained using this technique, however, the authors failed to discuss in details about important issues such as the relation between the number of selected DCT coefficients, the size of the images, and the cutoff frequency.

2.1.3.3 2D Gabor filters

Gabor filter or wavelet is one of the promising feature extraction techniques that is used to preserve the edges of an image or a signal. It has been used in most cases to extract features of the facial images. It has been applied as in [14] to specific areas of the face region, corresponding to nodes of a rigid grid, where for each node the Gabor coefficients are extracted. Gabor filters as feature extraction proved to be an efficient approach; however, they are not computationally efficient. More recently, Vitomir *et al.* [22] proposed a novel face classifier for face recognition called the Complete Gabor-Fisher Classifier that exploits both Gabor magnitude features as well as features derived from Gabor phase information. Unlike the majority of Gabor filter-based methods from the literature, which mainly rely only on the Gabor magnitude features for representing facial images, their proposed method combined the Gabor magnitude and the Gabor phase information to achieved robust illumination normalization.

2.1.3.4 Local Binary Pattern (LBP)

The Local Binary Pattern (LBP) is an algorithm for invariant feature extraction. It was first proposed for the use of texture description by Ojala *et al.* [46] and it has been used in the previous years to compensate and normalize illumination in the contexts of face detection and recognition. In the LBP algorithm the local neighborhood around each pixel is taken, the pixels of the neighborhood at the value of the central pixel are then thresholded and the resulting binary-valued image patch are used as a local image descriptor. The algorithm was originally defined for 3×3 neighborhoods, giving 8 bit codes based on the 8 pixels around the central pixel. The major drawback of the original LBP algorithm is that the center pixel cannot be compared with itself. So in some cases LBP cannot capture the local structure of the image area under analysis correctly. For overcoming this drawback, the modified LBP (LTP) is given by Tan and Triggs [28] called the Local ternary pattern (LTP) was proposed as an extension of LBP.

Tan and Triggs developed robust illumination normalization technique together with local texture based face representations and distance transform based matching metrics. They built local ternary patterns (LTP) on top of the local binary pattern (LBP) code and applied it on the images after a series of image preprocessing. They achieved an optimum illumination invariant system.

2.1.3.5 Near Infra-Red Techniques (NIR)

Li *et al.* in [21] presented a novel solution for achieving illumination invariant face recognition for indoor, cooperative- user applications, using active near infrared imaging techniques (NIR), and for building accurate and fast face recognition systems. Initially, they showed that face images of good condition can be obtained regardless of visible lights in the environment using active near infrared (NIR) imaging system. Then they utilized binary pattern (LBP) features to compensate for the monotonic transform, finally, they used statistical learning algorithms to extract most discriminative features from a large pool of invariant LBP features and constructed an accurate face recognition system. However, the drawback is that it is not yet suitable for uncooperative user applications such as face recognition in video surveillance. Moreover, because of strong NIR component in the sunlight, it is not appropriate for outdoor use.

2.1.4 Photometric normalization and preprocessing

Recently, attention has been focused on utilizing general purpose image processing techniques such as Histogram equalization [16], gamma intensity correction [49], and contrast equalization [15] to overcome the effect of illumination variation. Other sophisticated illumination normalization techniques include homomorphic filtering [16], isomorphic filtering [20], Anisotropic smoothing [20] among others.

A process called Local normalization was proposed by Xie and Lam [56] that is used for normalizing illumination variation in face recognition system. The process involves dividing the face image into triangular grids and each facet is normalized to zero mean and unit variance.

Moreover, illumination normalization is performed in the wavelet domain by Du and Ward [18] whereby Histogram equalization is applied to low-low subband image of the wavelet decomposition, thereafter, simple amplification is performed to accentuate high frequency components.

Arandjelovic *et al.* [9] proposed a novel framework for automatic face recognition in the presence of pose and varying illumination. The framework is based on simple image processing filters that are compared with unprocessed greyscale input to yield a single matching score between individuals. In the first place, they constructed the framework by extracting information regarding the change in illumination conditions in which data was acquired, and then they used it to optimally exploit raw and filtered imagery in casting the recognition decision [9]. This technique has yielded a 50–75% recognition error rates reduction, and having a recognition rate of 97% of the individuals. Other techniques under this category are discussed in the next chapter.

2.1.4 Utilization of 3-d morphable models

The basic idea behind the 3D morphable model approach is to utilize as little images as possible at enrolment time by extracting their 3D information to syntactically generate new images under different and strange pose and illumination. These syntactically generated images can be used to match any incoming query image and to determine a match based on parameter settings. Alternatively, these models can be used in an iterative fitting process whereby the model for each face is aligned and or rotated and artificially illuminated to best match the probe image. Example of this approach is given by Zhang and Cohen [58], in which they morphed 3D generic model of images from multi-view images by the use of a cubic polynomial. Blanz and Vetter [57] also proposed a face recognition system based on fitting a 3D morphable model. They used PCA analysis of the shape and texture of the images obtained from a database of 3D scans to describe the 3D shape and texture of each face separately. A new face image under novel pose and illumination is fitted to the model by an optimization process whereby the shape coefficients, texture coefficients and other parameters needed for representation of the image to minimize the difference of the input image and the rendered image based on those coefficients are also optimized. The rendering parameters are 3D translation, pose angles, ambient light intensities, directed light intensities and angles, and other parameters of the camera and color channels.

2.2 Conclusion

This chapter provides an overview of the recent literature in illumination normalization techniques. The methods for tackling the illumination problem includes Transformation of images into a canonical representation, Modeling of illumination variation, Preprocessing and Photometric normalization, Extraction of illumination invariant features and Utilization of 3-d morphable models.

CHAPTER 3

PREPROCESSING METHODS FOR FACE RECOGNITION

3.0 Introduction

A face recognition system would fail to match all the test images with the images in the target set correctly based on only computing the distance between the unprocessed gray-level images. To overcome this problem, the use of various image preprocessing techniques is employed. Preprocessing is the use of general purpose image processing techniques to eliminate irregularities in an image such as illumination variation, noise, rotation, and scale etc. The use of preprocessing in face recognition is generally used to overcome the effect of lighting, enhancing image contrast and normalizing the image in terms of rotation and scale.

Preprocessing play a vital role in face recognition algorithms by bringing the gallery, training and the test images into a normalized canonical form. Image preprocessing for face recognition include general purpose image preprocessing techniques such as- Contrast Equalization (CE), Histogram Equalization (HE), Gamma Intensity Correction (GIC) and specialized lighting normalization procedures such as Homomorpic filtering, Anistropic filtering, DCT coefficients, principal Components analysis (PCA), and logarithm transform, etc. Figure 3.1 below shows images with different preprocessing techniques.

In this chapter, various methods for preprocessing images for face recognition would be explored. In Section 3.1 Gamma Intensity Correction (GIC) methods would be explained, in Section 3.2 I would discuss Discrete Cosine Transform (DCT) coefficients. While in Section 3.3 Histogram Equalization (HE) and its variations (global and local) would be explained.

Section 3.4 Histogram remapping with normal distribution, Section 3.5 Histogram remapping with Log-normal distribution, Section 3.6 highlights Anisotropic filtering.

Figure 3.1: Example of images with different illumination conditions.

Top: Images from Yale B database. (Down):Images from CAS PEAL database.

3.1 Gamma Intensity Correction (GIC)

Gamma correction also called gamma nonlinearity or gamma encoding is the name of a nonlinear operation used to code and decode luminance values in video or still image systems. [49] Gamma intensity correction is used to control the overall brightness of an image by changing the gamma parameter and it can be used to correct the lighting variations in the face image [47].
The gamma correction is the process of taking the exponential of the input image by making pixel transform in which the output image is given as:

$$f(I(x, y)) = I(x, y)^{1/\gamma} \tag{3.1}$$

where $f(I(x, y))$ is the output and $I(x, y)$ is input, the input and output values are non-negative real values, and $\gamma \in [0,1]$. There are two processes associated with gamma correction; the first one is gamma compression in which gamma value $\gamma < 1$, and is sometimes called an encoding gamma. The second process is called gamma expansion whereby the gamma value $\gamma > 1$, and is also called a decoding gamma.

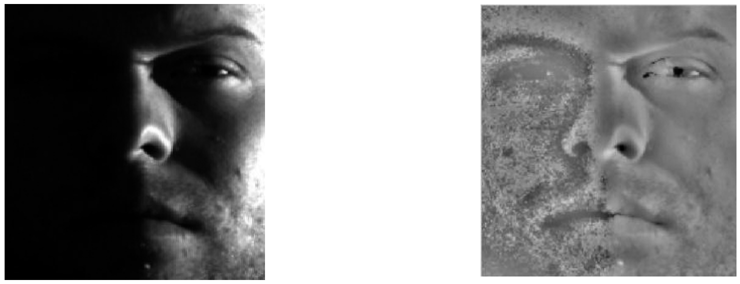
(a) (b)

Figure 3.2: Gamma Intensity Correction on Image from CMU PIE database.
(a) Before processing. (b) After Gamma processing.

The figure above shows an image from CMU PIE database before and after gamma processing. The output image would be darker or brighter depending on the value of gamma γ. In this work a value of gamma = 0.2 has been used. Gamma correction has been used in [15] and [47] for illumination normalization.

3.2 Discrete Cosine Transform (DCT)

The Discrete Cosine Transform is a novel approach for illumination normalization under varying lighting conditions used in face recognition algorithms that keeps facial features intact while removing excess lighting variations [11], [34]. The basic idea is that low frequency DCT coefficients are correlated with illumination variations; therefore, by truncating the DCT coefficients the variation in illumination can be significantly reduced. The discrete cosine transform (DCT) represents an image as a sum of sinusoids of varying magnitudes and frequencies [31]. It is a popular technique for image compression. Example of application of DCT is in JPEG image compression. The figure below shows an image from CMU PIE database before and after DCT processing.



(a) (b)

Figure 3.3: Discrete cosine Transform (DCT) on Image from CMU PIE database.
(a) Before processing. (b) After DCT processing.

Illumination variation typically lies in the low-frequency coefficients, so it can be reduced by removing low-frequency components in the logarithm domain. This is done by simply setting the low-frequency coefficients to zero, this works like a typical high-pass filter.

3.3 Histogram Equalization (HE)

Histogram Equalization is the approach that is most frequently used [1] for removing the effects of illumination in face recognition algorithms. Histogram Equalization produces an image with equally distributed brightness levels over the whole image in the brightness scale. It normalizes the illumination of the image by modifying the dynamic range of the image [34]. After the application of histogram equalization, the pixel intensities in the resulting image are flat. It has been shown in many works, for instance in [47] that application of histogram equalization offers a considerable performance gain in face recognition.

Histogram equalization is applied for compensating changes in illumination brightness, and differences in camera response curves. Sample image from CMU database is shown below, before and after Histogram equalization.

(a) (b)

Figure 3.4: Histogram equalization (HE) on Image from CMU PIE database.
(a) Before processing. (b) After (HE) processing.

There is global and local histogram Equalization. In Global Histogram Equalization the process is applied to enhance the contrast of the whole image, while in Local Histogram Equalization the process is only applied to a specific region of the face, however, this produces an unrealistic output of the image.

3.4 Histogram Remapping

However, since histogram equalization is a specific case of a more general model of histogram remapping techniques, there are other cases of this phenomenon that can be exploited. By investigating the characteristics of histogram equalization, it can be noted that histogram equalization remaps the histogram of a given facial image to a *uniform distribution*. Consequently, since there are numerous distributions, the target distribution could easily be replaced with another data distribution. This remapping can be justified as there is no theoretical evidence suggesting that the normal distribution is the only distribution that could be used in the process or is the most preferred in relation to other target distributions.

The question that can arise is how can other (non-uniform) target distributions be used in histogram remapping, and how can they influence the face recognition process and are they better suited for the recognition task [48]. To investigate these possibilities experiments are conducted using the Normal distribution and Log-normal distribution in the histogram remapping algorithm as suggested by this paper

[48]. Other distribution that is considered here is the exponential distribution. The figure below show images normalized with Histogram remapping with normal distribution (HRN) and Histogram remapping with log-normal distribution (HRL).

Figure 3.5: Histogram remapping using normal distribution (HRN).
(Upper row) Original unprocessed CAS PEAL images
(Middle row) The corresponding processed images with HR with Normal distribution (HRN)
(Lower row) The corresponding processed images with Log-normal distribution (HRL)

3.5 Anisotropic smoothing

This technique is also based on the reflection perception model. According to the two widely accepted and closely related assumptions about vision in humans: 1) human vision is mostly sensitive to scene reflectance and mostly insensitive to the illumination conditions, and 2) human vision responds to local changes in contrast rather than to global brightness levels, the AS technique is initiated. The two assumptions are related since local contrast is a function of reflectance. This work is pioneered by Gross and Brajovic [20] in which they find an estimate of L(x,y) such that R(x,y) is produced by dividing I(x,y) by L(x,y). These ensure that the local contrast is suitably improved.

Here I(x, y) is taken as the input stimulus, R(x, y) as the perceived sensation, while L(x, y) is then called perception gain which maps the input sensation into the perceived stimulus, that is:

$$I(x,y)*(1/L(x,y)) = R(x,y) \tag{3.2}$$

In this approach, the authors gathered evidence from experimental psychology using Weber's Law and derived their model. Weber's Law stated that the sensitivity threshold to a small intensity change increases proportionally to the signal level [20]. They defined the perception gain model L(x,y) as:

$$L(x,y)=I\psi(x,y)= I(x,y) \tag{3.3}$$

where I ψ (x, y) is the stimulus level in a small neighborhood ψ in the input image. The authors further regularized eqn (2) by imposing a smoothing constraint and solve for the perception gain model L(x,y) by minimizing the equation J(L), which is given by:

$$\mathrm{J(L)} = \iint_\Omega \rho(x,y)(\mathrm{L}-\mathrm{I})^2 dxdy \; + \; \lambda \iint_\Omega (\mathrm{L}_x^2 \; + \; \mathrm{L}_y^2) dxdy \tag{3.4}$$

where the first term finds the solution to follow the perception gain model, while the second term imposes a smoothness constraint. The space varying permeability weight ρ(x, y) controls the anisotropic nature of the smoothing constraint; Ω refers to the image region, while the parameter λ controls the relative importance of the two terms. Lx and Ly are the spatial derivatives of L, and I is the intensity image. The isotropic version of function J(L) can be obtained by discarding ρ(x, y). Examples of images preprocessed with Anisotropic Smoothing are given below:

Fig. 3.6 (Upper) unprocessed images from the CAS Peal database
(Lower) the corresponding images processed with Anisotropic filtering.

3.5 Conclusion

In this chapter, six preprocessing methods for illumination normalization used in face recognition algorithms have been studied: the Gamma Intensity Correction method (GIC), the Discrete Cosine Transform method (DCT), the Histogram Equalization method (HE); both global and local, Histogram Remapping with normal and Log-normal distributions and finally, the Anisotropic smoothing technique.

The experimental results on CASPEAL, Yale B and ATT databases showed that a simple illumination normalization method, like GIC, DCT or HE, can generally improve the appearance of a facial image, and consequently the performance of the facial- features recognition, compared with a non-preprocessed image. The methods have been only tested for images with frontal profiles and neutral facial expression and the results are very promising. Further experiments with two different classifiers with the preprocessing methods showed that Gamma Intensity Correction (GIC) and Historam Equilization (HE) are best matched with Euclidean distance measure, while Discrete Cosine Transform (DCT) and Anisotropic Smoothing (AS) are best matched with cosine distance measure.

CHAPTER 4

STATISTICAL APPROCHES/LINEAR SUBSPACES

4.0 Introduction

Linear subspace models were developed in the late 1980s by Sarovitch and Kirby [60] to efficiently represent human faces for the purpose of human face detection and recognition. Turk and Pentland [3] later improved this technique for face recognition. Principal Components Analysis (PCA) was one of these earlier linear subspace models. The PCA [3][6][8] algorithm has been very much utilized for the purpose of linear data projection. It projects the original image into a lower dimensional space that is most suitable for representing the image in a least squared sense, i.e. using minimum squared error. The goal of the PCA algorithm is to find the best subspace that well represents the data.

The Linear Discriminant Analysis (LDA) algorithm [2] is also a linear data projection technique. The LDA algorithm creates clusters of points based on the available different classes. Therefore, each cluster is labeled as a distinct class. LDA look for the projection that best minimize the distance of the points within each clusters while at the same time maximizing the distance between the different clusters. Generally LDA is enhanced with the use of PCA as a preprocessing stage.

4.1 The PCA Algorithm

The PCA is one of the most successful techniques that have been used in image recognition and compression. PCA is a statistical method under the broad title of factor analysis [26]. The purpose of Principal Component Analysis (PCA) is to reduce the large dimensionality of the data space to a smaller feature space with reduced dimensionality which is needed to describe the data economically. Typically, this is the case when there is a strong correlation between the observed variables. The preprocessed images from the previous section would then be used to find the principal components of the distribution.

PCA can do prediction, redundancy removal, feature extraction, data compression, etc. Here, PCA is used for feature extraction and as a preprocessing stage for the Linear Discriminant Analysis (LDA). The process is initiated by finding the mean from the entire training sample, and then the mean is subtracted from all the images from which the PCA algorithm would be applied.

The PCA algorithm is given as follows: Let X = $(x_1, x_2, \ldots, x_i, \ldots, x_N)$ represents the $\boldsymbol{n} \times \boldsymbol{N}$ data matrix, where each x_i is a face vector of dimension $\boldsymbol{n}$, concatenated from a $\boldsymbol{p} \times \boldsymbol{q}$ face image. Here n represents the total number of pixels $(\boldsymbol{p.q})$ in the face image and N is the number of face images in the training set. The PCA can be considered as a linear transformation (4.1) from the original image vector to a projection feature vector, i.e.

$$Y = W^T X \tag{4.1}$$

Where Y is the m × N feature vector matrix, m is the dimension of the feature vector, and transformation matrix W is an n ×m transformation matrix whose columns are the eigenvectors corresponding to the m largest eigenvalues computed according to the formula (4.2):

$$\lambda e_i = S e_i \tag{4.2}$$

Here the total scatter matrix **S** and the mean image of all samples are defined as [2]

$$S = \sum_{i=1}^{N} (x_i - \mu)(x_i - \mu)^T \,, \ \mu = \frac{1}{N} \sum_{i=1}^{N} x_i \tag{4.3}$$

After applying the linear transformation $\mathbf{W^T}$, the scatter of the transformed feature vectors $\{ y_1, y_2, , ,..., y_N \}$ is $\mathbf{W^T S W}$.

In PCA, the projection $\mathbf{W}_{\mathbf{opt}}$ is chosen to maximize the determinant of the total scatter matrix of the projected samples [4], i.e.,

$$W_{opt} = \arg\max|W^T SW| = [w_1 w_2 \dots w_m] \tag{4.4}$$

Where $\{w_i \mid i = 1,2,..., m\}$ is the set of n –dimensional eigenvectors of $\mathbf{S}$ corresponding to the m largest eigenvalues. In other words, the input vector (face) in an n -dimensional space is reduced to a feature vector in an m -dimensional subspace. It can be seen that the dimension of the reduced feature vector m is much less than the dimension of the input face vector ***n***, since ***m<n*** .

However, the major drawback of this technique is that the scatter being maximized is due not only to the between-class scatter that is useful for classification, but also to the within-class scatter that is due to unwanted illumination changes. Thus if PCA is presented with images of faces under varying illumination, the projection matrix $\mathbf{W_{opt}}$ will contain principal components (i.e., eigenfaces) which retain the variation due to lighting in the projected feature space. To overcome this drawback, The Principal Component Analysis PCA is performed as a preprocessing step for the linear discriminant analysis LDA, which takes the within class scatter into consideration.

4.2 LDA Algorithm

The projected images from the PCA would be used as input to the LDA algorithm. The goal of the Linear Discriminant Analysis (LDA) is to find an efficient way to represent each face vector space. PCA constructs the face space using the whole face training data as a whole, and not using the face class information. On the other hand, LDA uses class specific information which best discriminates among classes. LDA produces an optimal linear discriminant function which maps the input into the classification space in which the class identification of this sample is decided based on some metric such as Euclidean distance or cosine distance. LDA takes into account the different variables of an object and works out which group the object most likely belongs to [2].

There are many works that describe the principles of LDA algorithm, and all the descriptions are similar. The algorithm employed here is mainly based on [2]. Let a training set contain N face images representing C different classes of subjects. The face images in the training set are two-dimensional intensity arrays, represented as a vectors of dimension N. Different images of the same person are categorized as belonging to the same class while faces of different subjects are categorized as belonging to different classes. The LDA algorithm start with finding the mean image of the entire sample images using:

$$\mu_i = \frac{1}{N} \sum_{k=1}^{N} x_k \qquad (4.5)$$

Where N_i is the number of training samples in class X_i, and μ_i is the mean vector of samples belonging to class X_i, X_k represents the samples belonging to class X_i. Then the within scatter matrix and the between scatter matrix of the data are calculated, using:

$$S_W = \sum_{i=1}^{C} \sum_{x_k \in X_i} (x_k - \mu_i)\,(x_k - \mu_i)^T \qquad (4.6)$$

$$S_B = \sum_{i=1}^{C} (\mu_i - \mu)(\mu_i - \mu)^T \qquad (4.7)$$

Here $\mathbf{S_B}$ is the between-class scatter matrix and $\mathbf{S_W}$ is the within-class scatter matrix, of the training image set. In the above equation, C is the number of distinct classes, and μ is the overall mean of the sample space. Then the LDA selects W in (4) in such a way that the ratio of the between-class scatter and the within-class scatter is maximized [2]:

$$W_{opt} = argmax \frac{w^T s_B w}{w^T s_W w} \tag{4.8}$$

Assuming that **S** $_{\mathbf{W}}$ is non-singular (that's why we do PCA first), the basis vectors in **W** correspond to the first m eigenvectors with the largest eigenvalues of $\mathbf{S}_{\mathbf{W}}{}^{-1}\mathbf{S}_{\mathbf{B}}$.

The optimal projection $\mathbf{W}_{\mathbf{opt}}$ is chosen as:

$$W_{opt} = argmax \frac{w^T s_B w}{w^T s_W w} = [w_1 w_2 \dots w_m] \tag{4.9}$$

where {w i | i = 1,2,..., m} is the set of generalized eigenvector of S_B and S_W corresponding to the largest generalized eigenvalues{ λ_i | $i = 1,2,\dots, m$}, i.e.,

$$S_B w_i = \lambda S_W W_{i\,,} \quad i = 1,2,\dots,m \tag{4.10}$$

If we assume the number of classes is C, then there are at most C −1 nonzero generalized eigenvalues, and so an upper bound on m is C −1. The m –dimensional representation is then obtained by projecting the original face images onto the subspace spanned by the m eigenvectors. The representation w_i should enhance the separability of the different face objects under consideration.

In most practical face recognition problem, the within class scatter matrix S_W is singular because of the rank of W S is at most N − C. In general, N, the number of images in the training set is much smaller than n, the number of pixels in each image. This problem can be avoided by projecting the image set to a lower dimensional space so that the resulting S_W is nonsingular. This is first achieved by using PCA to reduce the dimension of the feature space to N − C, and then applying the standard LDA to reduce the dimension to C −1 [2]. Therefore, the number of fisher faces used are the number of distinct images in the training sample C minus one i.e., C - 1.

LDA transformation is strongly dependent on the number of classes, the number of samples, and the original space dimensionality.

4.3 Nearest Neighbor Classification

The *k*-nearest neighbor algorithm is amongst the simplest of all machine learning algorithms: an object is classified by a majority vote of its neighbors, with the object being assigned to the class most common amongst its ***k*** nearest neighbors (***k*** is a positive integer, typically small). If ***k* = 1**, then the object is simply assigned to the class of its nearest neighbor. Subsequently, when a probe image is brought for recognition, it would be passed through the preprocessing chain explained earlier and then the nearest neighbor classifier would be used to classify the new image based on cosine distance metric.

The training examples are vectors in a multidimensional feature space, each with a class label. The training phase of the algorithm consists only of storing the feature vectors and class labels of the training samples. In the classification phase, *k* is a user-defined constant, and an unlabeled vector (a query or test point) is classified by assigning the label which is most frequent among the *k* training samples nearest to that query point.

4.4 Conclusion

In this chapter, linear subspace models otherwise known as statistical models were discussed. The Principal component Analysis (PCA) was used to reduce the large dimensionality of the data space to a feature space of lower dimensionality that is needed for economic representation of the data. In the case of Linear Discriminant Analysis (LDA), the output of the PCA was used as input to the LDA to reduce the size of the data to a more manageable size. The LDA preserves the class peculiarities for easy separation.

Nearest Neighbor classification was employed for easy classification of classes, in which cosine or Euclidean distance measures were utilized.

CHAPTER 5

EXPERIMENTS

5.0 Introduction

In this chapter, experiments were carried out that illustrate the effectiveness of the proposed method using three publicly available face databases with considerable illumination variations. This chapter explains the various experiment carried out in this work, the preprocessing techniques used in conjunction with the linear subspace model, and the various databases used and the standard protocol used in evaluating the result.

5.1 Databases Used

Three publicly available databases are used in this research work, the databases are: CAS PEAL database [51], Extended Yale Face Database B (‘Extended Yale-B’) [25] and AT&T Database [24].

5.1.1 CAS PEAL-R1 Database

The CAS-PEAL face database has been constructed under the sponsorship of National Hi-Tech Program and ISVISION [51]. The goals to create the PEAL face database include: providing the worldwide researchers of Face Recognition community a large-scale Chinese face database for training and evaluating their algorithms; facilitating the development of Face Recognition by providing large-scale face images with different sources of variations, especially Pose, Expression, Accessories, and Lighting (PEAL); advancing the state-of-the-art face recognition technologies aiming at practical applications especially for the oriental. Currently, the CAS-PEAL face database contains 99,594 images of 1040 individuals (595 males and

445 females). A selected subset of the database (CAS-PEAL-R1, containing 30 863 images of the 1040 subjects) is available to other researchers now.

5.1.2 Extended Yale B Database

The Yale Face Dataset [25] containing 10 people under 64 different illumination conditions has been the standard database for studies of variable lighting over the past two decades. It was recently updated to the Extended Yale Face Database B [30], containing 38 subjects under 9 poses and 64 illumination conditions. In both cases the images are divided into five subsets according to the angle between the light source direction and the central camera axis (12°, 25°, 50°, 77°, 90°). Only the Extended Yale B Database is used for these experiments.

5.1.3 ATT Database.

The AT&T database of faces, formally ORL database, contains ten different images of each of 40 distinct subjects. For some subjects, the images were taken at different times, varying the lighting, facial expressions (open / closed eyes, smiling / not smiling) and facial details (glasses / no glasses). All the images were taken against a dark homogeneous background with the subjects in an upright, frontal position (with tolerance for some side movement) [24]. The images contain hair and ear regions (i.e., not cropped).

5.2 Experimental Setup

The results obtain from these experiments and the experimental settings are given below. In these experiments only frontal face views were used in the experiment, but lighting, expression and identity may all vary. All of the images in the CAS PEAL-R1 lighting subset, and all the images in the Extended Yale B database undergo the same geometric normalization before the analysis, that is, conversion to 8 bit gray-scale images; rigid scaling and image rotation to place the centers of the two eyes at fixed positions, using the eye coordinates supplied with the original datasets; while for the AT&T database, the images are 8 bit gray-scale images containing hair and ear regions. All the images in the databases were resized to *100 x 100* pixels.

For the testing, analysis and evaluation of the proposed method, a toolbox containing MATLAB scripts named "The PhD face recognition toolbox" was used. The tool was made publicly available free of charge by Struct [22]. Another collection of MATLAB face recognition files called "FaceRecEvaluator" was also utilized. The FaceRecEvaluator was developed by Brian and Enrique [50], and is available freely for use in academic and research domains.

5.3 Results

This section outlines the results obtained from applying the afore mentioned preprocessing techniques in chapter three and the linear projection techniques mentioned in chapter 4 on the databases listed in the above section using the development tools highlighted in chapter 1.

Five (5) preprocessing techniques and two (2) distance metrics were experimented with using these databases; the preprocessing techniques are Gamma intensity correction (GIC), Discrete Cosine Transform (DCT), Histogram remapping using Normal distribution (HRN), Histogram remapping using Log-normal distribution (HRL), and Anisotropic Smoothing technique (AS). The distance metrics are Euclidean (EUC) and cosine (COS) distance metrics. In total we have, ten (10) set of experiments using these databases namely: (GIC+EUC), (GIC+COS), (DCT+EUC), (DCT+COS), (HRN+EUC), (HRN+COS), (HRL+EUC), (HRL+COS), (AS+EUC), (AS+COS).

Table 1. Preprocessing Method/Classifier

Preprocessing Method/Classifier	Cosine (COS)	Euclidean (EUC)
GIC	GIC+COS	GIC+EUC
DCT	DCT+COS	DCT+EUC
HRN	HRN+COS	HRN+EUC
HRL	HRL+COS	HRL+EUC
AS	AS+COS	AS+EUC

Table2. Recognition Accuracy and Total Execution time of CASPEAL and Yale B databases using GIC technique and the cosine distance metric

Accuracy (%)						
Dataset/ Algorithm	CASPEAL	YALEB2	YALEB3	YALEB4	YALEB5	ATT
lda	62.7	100	100	92.8	95.4	95.6
Total Execution Time (ms/img)						
lda	16.5	1.9	2.1	2.1	2	1.5

Table 3. Recognition Accuracy and Total Execution time of CASPEAL and Yale B databases using GIC technique and the Euclidean distance metric

Accuracy (%)						
Dataset	CASPEAL	YALEB2	YALEB3	YALEB4	YALEB5	ATT
lda	64.9	100	100	94.7	95.4	94.4
Total Execution Time (ms/img)						
lda	3.2	0.9	1	1.1	1	0.9

5.3.1 Result of the CAS PEAL-R1 Database

The CAS peal contains six (6) different subsets in the frontal category. These subsets are Accessory, Lighting, Distance, Expression, Time and Background subsets. Since this work is concerned with the lighting problem in face recognition systems, only the Lighting subset is used. 10 images of randomly chosen 77 subjects making a total of 770 images were used from the Lighting subset. Sixty 60% of the images were used as training set while the other 40% were used as testing set.

Table 4. Recognition Accuracy and Total Execution time of CASPEAL and Yale B databases using DCT technique and the cosine distance metric

Accuracy (%)						
Dataset	CASPEAL	YALEB2	YALEB3	YALEB4	YALEB5	ATT
lda	22.7	96.7	83.6	64.5	59.2	28.8
Total Execution Time (ms/img)						
lda	15.5	2	2	2	2	1.5

Table 5. Recognition Accuracy and Total Execution time of CASPEAL and Yale B databases using DCT technique and the Euclidean distance metric

Accuracy (%)						
Dataset	CASPEAL	YALEB2	YALEB3	YALEB4	YALEB5	ATT
lda	16.6	96.7	78.9	51.3	50	28.8
Total Execution Time (ms/img)						
lda	3.3	1	1	0.9	1	0.7

Table 6. Recognition Accuracy and Total Execution time of CASPEAL and Yale B databases using HRL technique and the cosine distance metric

Accuracy (%)						
Dataset	CASPEAL	YALEB2	YALEB3	YALEB4	YALEB5	ATT
lda	59.1	100	100	78.3	77.6	95
Total Execution Time (ms/img)						
lda	15.3	2	1.9	2	0.9	1.4

Table 7. Recognition Accuracy and Total Execution time of CASPEAL and Yale B databases using HRL technique and the Euclidean distance metric

Accuracy (%)						
Dataset	CASPEAL	YALEB2	YALEB3	YALEB4	YALEB5	ATT
lda	59.4	100	100	78.9	73.7	95
Total Execution Time (ms/img)						
lda	3.1	0.9	1	1	0.9	0.7

Table 8. Recognition Accuracy and Total Execution time of CASPEAL and Yale B databases using HRN technique and the Euclidean distance metric

Accuracy (%)						
Dataset	CASPEAL	YALEB2	YALEB3	YALEB4	YALEB5	ATT
lda	63	100	100	80.3	80.3	95
Total Execution Time (ms/img)						
lda	3.3	0.9	1.1	0.9	1	0.8

Table 9. Recognition Accuracy and Total Execution time of CASPEAL and Yale B databases using HRN technique and the cosine distance metric

Accuracy (%)						
Dataset	CASPEAL	YALEB2	YALEB3	YALEB4	YALEB5	ATT
lda	65.6	100	100	78.3	77.6	95
Total Execution Time (ms/img)						
	15.8	2	2	2	1.9	1.5

Table 10. Recognition Accuracy and Total Execution time of CASPEAL and Yale B databases using AS technique and the cosine distance metric

Accuracy (%)						
Dataset	CASPEAL	YALEB2	YALEB3	YALEB4	YALEB5	ATT
lda	51.6	100	99.3	82.9	92.1	87.5
Total Execution Time (ms/img)						
lda	4.5	1.5	1.5	1.5	1	1.5

Table 11. Recognition Accuracy and Total Execution time of CASPEAL, Yale B and ATT databases using AS technique and the Euclidean distance metric

Accuracy (%)						
Dataset	CASPEAL	YALEB2	YALEB3	YALEB4	YALEB5	ATT
lda	28.6	100	99.3	88.2	90.8	86.3
Total Execution Time (ms/img)						
lda	4.5	1.5	1.5	1.5	1	0.8

5.3.2. Result Of The Extended Yale B Database

The Extended Yale B database is classified into 5 subsets, from subset 1 to 5 according to degree of illumination variation. Subset 1 contains images with no illumination variation, while subsets 2 to 5 are arranged from slight to extreme illumination variation. For these experiments, Subset 1 contains 7 images each for the 38 subjects but this subset is not used. Subset 2 contains 12 images each, subset 3 contains 12 images each while subset 4 contains 14 images each and subset 5 contains 19 images plus one ambient (background) image (2414 images of 38 subjects for the Extended dataset) but only 10 images were used per subject in each subset 2 to 5, making the total number of images used in this database to be 10x4x38 which is equivalent to 1520 images. Examples of images in this database are shown in fig. 3.1.

5.3.3. Result Of The ATT Database

This database contains ten (10) images of forty (40) subjects, as stated in the database section. All the images are 8-bit gray scale containing no illumination variation. This database is used as control database to monitor the performance of the preprocessing methods employed on this type of database that contain no illumination challenge. All the 400 images were used for these experiments.

5.4 Conclusion

This chapter gave a detailed explanation about the databases used and the result obtained from various permutations of five different preprocessing methods and two distinct classifiers. The databases used were CASPEAL database in which only the frontal category was used, then the extended Yale B database in which subset 2 through subset 5 were used, and finally the AT&T database in which the database was used as a control database to monitor the performance of the preprocessing methods used.

The total execution times of all the experiments were taken whereby it was observed that the more the illumination variation the longer the execution time recorded. Apart from this fact, it was also observed that the Euclidean classifier is always faster than the cosine classifier. Analysis of the result is given in the subsequent chapter.

CHAPTER 6

ANALYSES AND DISCUSSIONS

6.0 Introduction

In this chapter, analysis and discussion of the various performances i.e. recognition accuracy and execution time of the techniques employed are made. The recognition accuracy is the rank one recognition accuracy measure, meaning that a face is only counted as success if it is the first face returned by the matching algorithm. The analysis and discussion is structured according to each of the three databases CAS PEAL, Extended Yale B, and ATT used and the five illumination normalization techniques with the two different distance metrics. This produces 30 set of experiments.

6.1 CAS PEAL Database:

Ten (10) experiments were conducted using this database namely- Gamma Intensity correction, Discrete Cosine Transform, Histogram remapping using normal distribution, Histogram remapping using Log-normal distribution, and Anisotropic Smoothing technique, each technique using Euclidean and Cosine distance metrics.

According to figure 6.1 below, Gamma Intensity Correction (GIC) together with Euclidean distance yielded the second best performance with 64.9% recognition rate in 3.2ms/img, while Gamma Intensity Correction (GIC) together with Cosine distance gave 62.7% recognition rate in 16.5ms/img. However, Discrete Cosine Transform (DCT) in either case i.e., with Euclidean or Cosine distance metric gave the lowest performance for this database. The DCT with Euclidean distance produces 10.4%, while DCT with Cosine distance produces 17.2% in 3.1ms/img and

15.1ms/img respectively. Histogram remapping using normal distribution (HRN) with Euclidean distance gave 63% recognition accuracy in 3.3 ms/img while Histogram remapping using normal distribution (HRN) with cosine distance yielded the best performance for this database with 65.6% recognition rate in 15.8 ms/img. However, the only minor setback of this composition (HRN + Cos) is the execution time, which is slower than the (HRN + Euc) combination.

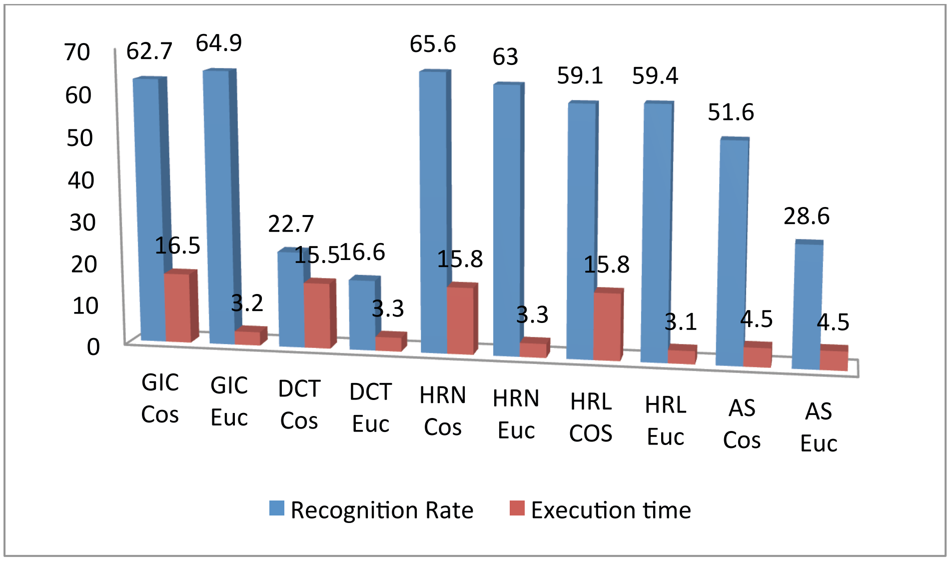


Fig. 6.1 Performances of different preprocessing techniques on the CAS PEAL database using two distance metrics.

On the other hand, As can be seen from the above figure, histogram remapping using Log-normal distribution with cosine distance generates 59.1% recognition accuracy in 15.3ms/img, while this method together with the Euclidean distance produces 59.4 in 3.1ms/img. Is it important to note that using this technique the Euclidean distance gives better performance in terms of accuracy and execution time. The last experiment conducted with this database is one of the sophisticated illumination normalization techniques which is the Anisotropic Smoothing (AS) method. This technique was also applied to the CAS PEAL database using the Euclidean and Cosine distance metrics. The first experiment with Euclidean distance measure yielded an accuracy of 51.6% in 4.5 ms/img, while the second experiment produced 28.6% in 4.5 ms/img with the cosine distance measure.

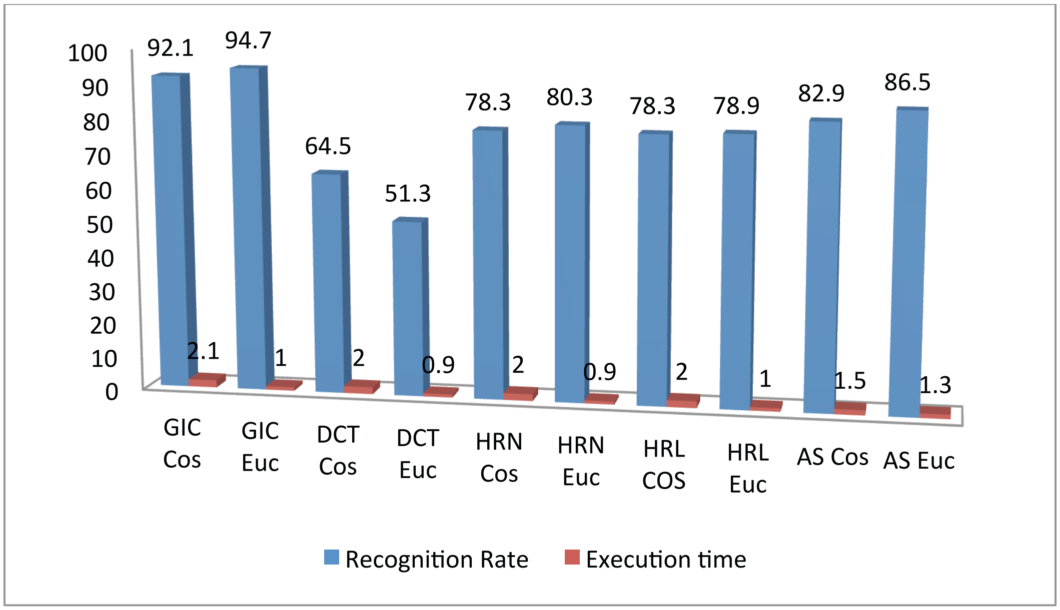


Fig. 6.2 Performances of different preprocessing techniques on subset 4 of the Yale B database using two distance metrics

6.2 Extended Yale B Database

As noted earlier the Extended Yale database is divided into subsets as initially suggested by the authors of the database. These experiments were conducted on each subset of the database and the results were analyzed separately.

6.2.1 Yale B Subset 2 Result:

The first experiment carried out on this database is the Gamma Intensity Correction (GIC) technique with the Euclidean distance and the second is the same technique using the cosine distance, both results showed a 100% in recognition accuracy in 0.9ms/img and 1.9ms/img respectively. This also showed that the Euclidean distance measure is faster than the cosine distance metric. Experiment with the Discrete Cosine Transform (DCT), which is implemented using Matlab's dct2 implementation, yielded 96.7% recognition rate in both cases of Euclidean and cosine distance metric. The only difference is in the execution time, the Euclidean distance is faster as usual with 1ms/img whereas the cosine distance yielded the performance in 2 ms/img.

In histogram remapping using normal distribution the recognition accuracy using the Euclidean distance measure was 100% in 0.9ms/img. The same experiment using cosine distance gives same result but with an execution time of 2ms/img. Similarly, in histogram remapping using log-normal distribution the recognition accuracy was 100% also using the Euclidean distance measure, and 100% again using the cosine distance measure. Both the execution times resemble those of the previous experiments respectively. A similar result was obtained using the anisotropic Smoothing method in which 100% recognition rate was produced by both distance metrics with the similar execution time.

6.2.2 Yale B Subset 3 Result

This subset contains images that have more illumination variation than the previous subset. The GIC plus Euclidean and the GIC plus the cosine distance generates 100% recognition rates each in 1ms/img and 2.1ms/img respectively. A recognition rate of 83.6% in 2ms/img was obtained from the DCT method using the cosine distance, while 78.9% was obtained from this subset using DCT + Euc combination in 1ms/img. In the histogram remapping technique using normal distribution, a recognition rate of 100% was also obtained in both cases using the two distance metrics. The execution times differ with 0.9ms/img, with Euclidean distance having 1.1ms/img and cosine distance having 2ms/img.

Histogram remapping using log-normal distribution yielded 100% recognition rate in 1.9ms/img using cosine distance and same method using Euclidean distance gave same result but in 1ms/img. The Anisotropic smoothing technique is the final experiment carried-out using this subset. The result obtained showed a recognition rate of 99.3% in 1.5ms/img and 98% in 1.1ms/img using the cosine and Euclidean distance metrics.

6.2.3 Yale B subset 4 result

The fourth subset of this database contains images with second highest degree of illumination variation. As shown in figure 6.2 above, experiments carried out with this database produce the following results: GIC + Cos yielded a recognition rate of 92.1% in 2.1ms/img, while GIC + Euc 94.7% in 1.1ms/img. It can be realized here also that the Euclidean distance gave the best performance. In DCT experiments, DCT + Cos produces 64.5% in 2 ms/img while, DCT + Euc generates 51.3 % in 0.9 ms/img. Analyzing this and the previous result under this technique closely, one can see that the DCT method is best matched with the cosine distance measure.

On the other hand, experiments with histogram remapping using the normal distribution (HRN + Euc) on this subset produce 80.3% in 0.9 ms/img using Euclidean distance and (HRN + Cos) produces 78.3 % in 2 ms/img using cosine distance. Similar results were obtained using this remapping technique with lognormal distribution. Experiments with cosine distance produce 78.3 % in 2 ms/img, while that of Euclidean distance produce 78.9% in 1 ms/img. Between these two methods the HRN + Euc gives the best result. The last experiment on this subset is the anisotropic smoothing method. The AS + cos technique gives 82.9 % in 1.5 ms/img while the AS + Euc gives 86.5 % in 1.3 ms/img.

It can be noted that the anisotropic technique (AS) perform better than the three methods on this subset; that is the DCT, HRN and HLN. However the gamma intensity correction (GIC) out perform all the techniques on this subset.

6.2.3 Yale B subset 5 result

This subset is last and most complex of the extended Yale B database. Experiments conducted on this database yielded the following results, see the figure 6.3 below: GIC with cosine distance produces 95.4% recognition rate in 2 ms/img whereas GIC with Euclidean distance produces the same result in 1 ms/img. The next experiment is the DCT with the cosine distance measure which gave a recognition accuracy of 59.2% in 2 ms/img while the DCT + euc drops performance by generating only 50% recognition rate in 1ms/img. It

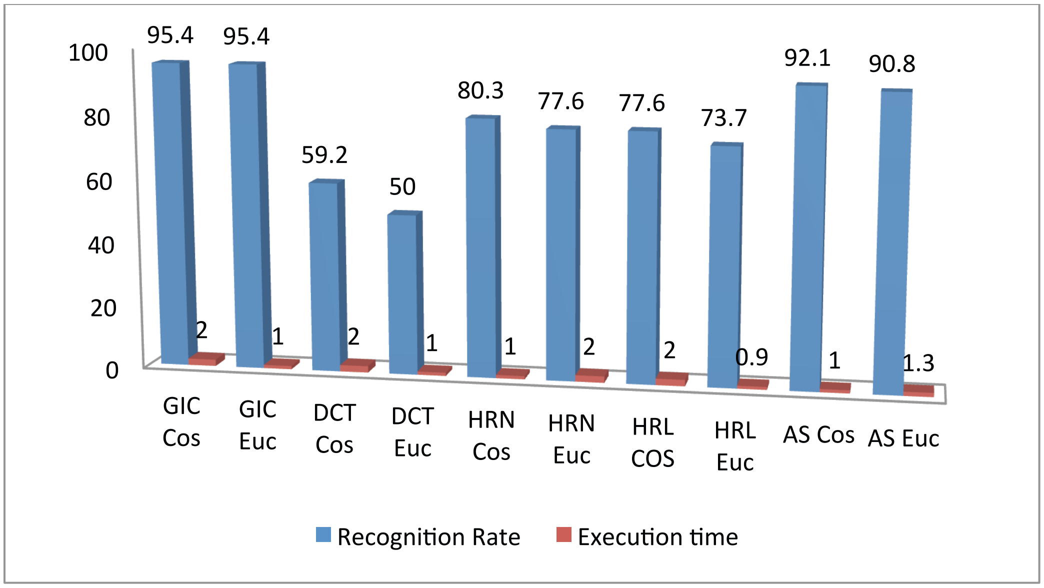


Fig. 6.3 Performances of different preprocessing techniques on subset 5 of the Yale B database using two distance metrics.

is relevant to point out that the DCT always perform better when combined with cosine distance metric.

In the next set of experiments histogram remapping with normal distribution (HRN) was applied to this subset, the result indicated HRN perform better than HRL in both cases using the two distance metrics. HRN+euc gave 80.3% in 1 ms/img whereas HRN+cos gave 77.6% in 2ms/img. Similarly, HRL with cosine distance produced 77.6% in 2ms/img and HRL+euc produced 73.7% in 0.9ms/img. In the anisotropic smoothing experiment, the AS + cos combination yields an amazing 92.1% recognition accuracy in 1ms/img, and the AS + euc method yields 90.8% recognition accuracy in just 1ms/img. It can be gathered from the experiments conducted on this complex subset that Gamma Intensity correction performed better than all the techniques used on the subset.

6.3 AT&T Database

This database was used as a means of assessing the performance of the techniques on a database devoid of any illumination challenge. However, the database contains slight variations due to pose, accessories (glasses), and expression. Figure 6.4 below gave the details of the experiments on this dataset. The gamma Intensity correction with the cosine distance yielded 95.6% in 1.4ms/img, while the same technique with Euclidean distance yielded 94.4 % in 0.9 ms/img. DCT technique with the cosine distance resulted in 28.8% recognition rate in 1.5ms/img whereas DCT +euc indicated a recognition accuracy of 28.8% in 0.7ms/img.

When this database was evaluated with the histogram remapping techniques the following results were obtained: HRN+COS produced 95% in 1.5ms/img recognition accuracy whereas HRN + EUC produced 95% accuracy in 0.8ms/mg. In the same way, HRL+COS resulted in 95% recognition rate in 1.4ms/img while HRL+EUC yielded the same 95% accuracy in 0.7ms/img. These show that the remapping techniques produce somewhat similar results with little difference in execution time. The next technique evaluated on this database is anisotropic smoothing technique. The results indicated a drop in recognition rate from the previous results; AS+COS produced 87.5% in 1.5ms/img but AS + EUC produced 86.3% in 0.7ms/img.

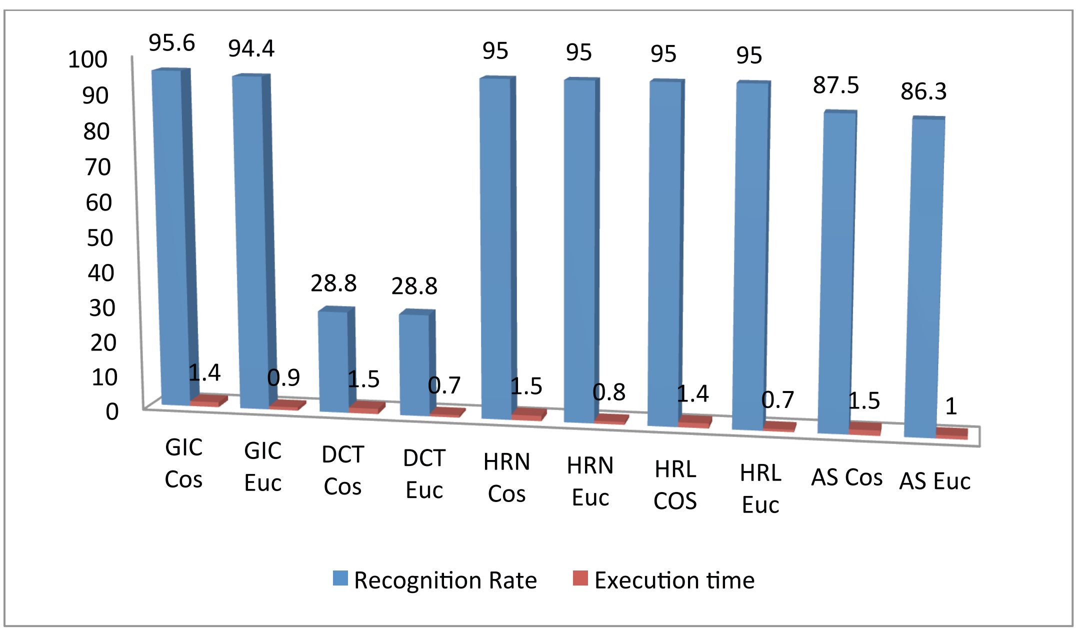


Fig. 6.4 Performances of different preprocessing techniques on ATT database using two distance metrics

6.4 Comparison with the performance of other preprocessing methods

The work conducted here which is mainly searching for the best processing method and its suitable classification method for illumination tolerant face recognition does not have, as far as the author's knowledge is concerned, any similar work to compare the results with judiciously. I make this assertion because firstly, other available works are concerned with finding the optimum (preprocessing) technique for illumination insensitive face recognition not comparing preprocessing techniques vs. classification method for robust face recognition system, on the other hand, other researchers are comparing different preprocessing methods than the one that were compared in this work. Secondly, the number of the training and testing images are not consistent with the ones used in other methods, as here the main concern is not trying to find the effect of training vs. test set in face recognition.

6.5 Conclusion

This chapter provides a concise analysis and discussion of the overall experiment conducted in the previous chapter. To facilitate accurate examination and interpretation of the result, rank 1 recognition rate which is given as the percentage of tested images that were correctly identified, was recorded together with the execution time per image. It has been observed that the more the database contains illumination variation the more time it takes to execute. Moreover, it has been gathered that among the preprocessing methods and the classifiers, GIC is best matched with Euclidean distant metric, while DCT, HRN, HRL and AS are best coupled with Cosine distance metric.

In the case of the CASPEAL database which contained less illumination condition but sharper shadows making it difficult for accurate separation, the HRN+EUC produced better result than the other four preprocessing methods. However, in the case of Extended Yale B database – subset 4 and subset 5 – which contained complex illumination, the GIC+EUC provides better result in terms of performance and speed. Similarly, in the case of the ATT database which is used as a control database the GIC+EUC also produced the best results among the other preprocessing methods.

CHAPTER 7

CONCLUSION, RECOMMENDATION AND FURTHER WORK

7.0 Introduction

This chapter provides a concise Summary, conclusion, recommendations and the area of further research.

7.1 Summary

This thesis is mainly focused on designing a face recognition system which is invariant to illumination variation based on finding the best combination between the preprocessing method and the classifier. Attention is concentrated on the preprocessing part of the system. Different image preprocessing techniques for face recognition were proposed and experimented with. The sequence of execution of the proposed method includes the preprocessing step, PCA/LDA subspace, and cosine/Euclidean classifiers.

To facilitate a comprehensive study and analysis, five different preprocessing techniques were implemented on the PCA/LDA model using two different classifiers. These yielded ten (10) set of experimentations on each of the three databases used. The preprocessing techniques used were Gamma Intensity Correction (GIC), Discrete Cosine Transform (DCT), Histogram Remapping using Normal distribution (HRN), Histogram Remapping using Log-normal distribution (HRL), and Anisotropic Smoothing method (AS).

7.2 Conclusion

In this thesis work, different preprocessing techniques for face recognition systems have been proposed and implemented using hybrid approach and linear subspace modeling for feature extraction, and dimensionality reduction and cosine or Euclidean distance metric for classification. The proposed preprocessing techniques are Histogram Remapping using Normal distribution (HRN) combined with cosine distance metric and Histogram Remapping using Log-normal distribution (HRL) also combined with cosine distance classifier on databases that contain lighting variations or over exposure like the CAS PEAL lighting subset. While for a database that contain multiple dark regions and shadows the Gamma Intensity Correction combined with Euclidean distance classifier would be sufficient, provided that the parameters were set right. Other preprocessing techniques experimented with are Discrete Cosine transform (DCT), and Anisotropic Smoothing (AS). The above mentioned techniques resulted in ten types of face recognition methods: (GIC+EUC), (GIC+COS), (DCT+EUC), (DCT+COS), (HRN+EUC), (HRN+COS), (HRL+EUC), (HRL+COS), (AS+EUC), (AS+COS).

The performances of these ten methods have been evaluated in terms of percentage of recognition accuracy, and for the total execution time to monitor efficiency considering the CASPEAL, YALE B and ATT (Formally ORL) databases. The following conclusions are made based on the results and analyses of the above mentioned face recognition techniques:

- The Gamma Intensity correction provides very good performance on all the databases particularly those with extreme illumination condition like the subset 4 and subset 5 of the Extended Yale B database. However, the technique was out performed by histogram remapping using normal distribution on the CASPEAL lighting subset. Generally, the GIC method performed at its best when combined with the Euclidean distance metric, i.e. the (GIC + EUC) arrangement.

- The Histogram remapping technique play a vital role in the CASPEAL complex database in which the lighting and illumination variation is at the extreme with some images very much over-exposed while others are very much under-exposed. In this category of illumination variation technique, the HRN performed better than the HRL in almost all the cases. HRN provides better results when combined with the cosine distance metric.

- The Discrete cosine transform method (DCT) generates the worst recognition accuracy in all the databases used. It can be concluded from these set of experiments that this method is not the best in terms of illumination normalization for face recognition purposes. However, despite this low performance, the DCT method perform better when merged with the cosine distance metric even though it takes some time to finish execution.

- The Anisotropic Smoothing technique (AS) provides second best performance on the Yale B database subsets, however, not so good result was obtained when applied to the CASPEAL lighting subset. This method work well when combined with the Cosine distance metric.

- Between the two distances metrics studied, the cosine distance produce superior performance in almost all the experiments carried out with some minor exceptions like in the GIC technique. The only drawback of this method is the execution time with is slower than the Euclidean distance measure.

7.3 Recommendation

In this research project various lighting normalization techniques were presented and compared according to recognition accuracy and execution time. The lighting or illumination normalization technique that would be recommended for used are itemized as follows:

✓ When tackling extreme illumination condition with shadows and dark regions (e.g. the subset 5 of the Yale B database), the Gamma intensity Correction would be adequate provided that the associated parameters are set right.

✓ The gamma intensity correction is best combined with the Euclidean distance metric.

- ✓ When tackling lighting variation problems with over exposure (e.g. lighting subset of the CASPEAL database), histogram remapping with the normal distribution (HRN) would be the best choice for the problem.
- ✓ The Histogram remapping technique (HRN) is best paired with cosine distance metric.
- ✓ The Discrete Cosine Transform (DCT) method is best combined with the cosine distance metric as a classifier.
- ✓ The DCT method should be least considered when trying to solve illumination or lighting variation problems as it generates the least accurate results for the experiments conducted in this research work.
- ✓ The cosine distance metric produce more accurate result compared with the Euclidean distance metric.
- ✓ The Euclidean distance metric is always faster than the cosine distance metric.
- ✓ The anisotropic smoothing (AS) technique provides considerable performance on the databases used but other techniques provides superior performance. i.e. GIC and HRN.
- ✓ The AS technique is best matched with the cosine distance metric.
- ✓ The performances of the lighting\illumination techniques tested here should be verified on a much larger scale database to assess their performances on such databases.

7.4 Further work

The face recognition system developed here works with limitations; it considers only frontal profile images.

- Further work will include dealing with more variations such as pose and extreme facial expression variations and partial occlusion.

- On the other hand, other image preprocessing techniques used for illumination normalization in face recognition systems would be explored.

- The algorithm would be expanded to accommodate large scale databases for accurate analysis and assessment.

7.5 Conclusion

In this chapter, the summary of the whole thesis was given whereby the main aim and objectives of whole work was elucidated. The main aim of the work was to find the right match between the preprocessing method and the classifier for efficient illumination tolerant face recognition system. Conclusions, recommendations and area of further work were also highlighted in the chapter. Some of the recommendations includes what preprocessing method is best match with what classifier, and which preprocessing technique + classifier is best for which kind of database. Further work would focus on expanding the scope of the algorithm to incorporate many variations such as pose and partial occlusion, etc.

REFERENCES

1. W. Zhao, R. Chellappa, A. Rosenfeld, and P.J. Phillips, "Face Recognition: A literature Survey", Technical Report, Univ. of Maryland, 2000.

2. A. Eleyan & H. Demirel, "PCA and LDA based Nearest neighbors for Human Face Recognition", Face Recognition, by Kresimir Delac and Mislav Grgic, I-Tech Education and Publishing, Chap. 6, pp. 93-106, July 2007, ISBN 978-3-902613-03-5.

3. M. Turk, & A. Pentland, "Eigenfaces for Recognition", Journal of Cognitive Neuroscience, Vol. 3, (1991) 71-86. 0898-929X

4. K. R. Singh, M. A. Zaveri, M.M. Raghuwanshi, "Illumination and Pose Invariant Face Recognition : A Technical Review," International Journal of Computer Information Systems and Industrial Management Applications (IJCISIM) ISSN: 2150-7988 vol. 2, pp. 29–38, 2010.

5. W.Y. Zhao, and R. Chellappa, "Illumination insensitive face recognition using symmetric shape-from shading", IEEE Conference on Computer Vision and Pattern Recognition, vol.1, 2000. pp. 286-293.

6. P.N Belhumeur, J.P Hespanha, and D.J Kriegman, "Eigenfaces vs. Fisherfaces: recognition using class specific linear projection," IEEE Transactions on pattern analysis and machine, vol.19, 1997, pp.711-720.

7. M.S. Bartlett, J.R. Movellan, T.J. Sejnowski, "Face recognition by independent component analysis", IEEE Transaction on Neural Networks 13 (2002) 1450–1464.

8. P.N Belhumeur, & D.J Kriegman, "What is the set of images of an object under all possible illumination conditions," International Journal Computer Vision, vol. 28, no.3, 1998 pp.245–260.

9. O. Arandjelovic, and R. Cipolla, "A Methodology for rapid illumination-Invariant face recognition using image processing filters," Int. J. Computer Vision and Image Understanding, vol. 113, 2009. pp. 159-171.

10. V. V. Kohir, and U. B. Desai, "Face recognition using DCTHMM approach," In proceedings of the fourth IEEE workshop on applications of computer vision, 1998, pp. 226-231.

11. W. Chen, Er. M. J, and S. Wu, "Illumination compensation and normalization for robust face recognition using discrete cosine transform in logarithm domain," IEEE Transactions on Systems, Man and Cybernetics, vol.36, no.2, 2006, pp. 458-466.

12. T. Zhang, Y. Y. Tang, Z. Shang, and X. Liu, "Face Recognition Under Varying Illumination using Gradientfaces," IEEE Transactions, vol.18, no.11, 2009, pp.2599 – 2606.

13. M. Lades, J. C. Vorbruggen, J. Buhmann, J.Lange, C. Malsburg, R. P. Wurtz, and W. Konen, “Distortation invariant object recognition in the dynamic link architecture,” IEEE Transactions., vol.42, 1993, pp.301-311.

14. L. Nanni, and D. Maio,“Weighted sub-Gabor for face recognition,” Pattern Recognition Letters, vol.28, no.4, pp. 487-492. 2007.

15. X. Tan, and B. Triggs, “Enhanced local texture feature sets for face recognition under difficult lighting conditions,” In Proceedings of the IEEE international workshop on analysis and modeling of faces and gestures, 2007. pp. 168-182.

16. R. Gonzalez and R. Woods. Digital Image Processing. Prentice Hall, second edition, 1992.

17. D. W. Jacobs, P. N. Belhumeur, and R. Barsi, “Comparing images under variable illumination,” In Proceedings, IEEE conference on computer vision and pattern recognition, 1998. pp. 610-617.

18. S. Du, and R. Ward, “Wavelet based illumination normalization for face recognition,” In Proceedings of international conference on image processing, vol. 2, 2005, pp. 954-957.

19. Y. Gao, and M.K.H. Leung, ,“Face Recognition using line edge map,” IEEE Trans.Pattern Anal. Machine Intell., vol.24, no.6, 2002 pp. 764-779.

20. R. Gross, V. Brajovic, “ An image preprocessing algorithm for illumination invariant face recognition”. In: Kittler, J., Nixon, M.S. (eds.) AVBPA 2003 LNCS, vol. 2688, 2003.pp. 10–18. Springer, Heidelberg

21. S. Li, R. Chu, S. Liao, and L. Zhang, “Illumination Invariant Face Recognition using Near-Infrared Images,” In proc. On IEEE Tranc. On Pattern Analysis and Machine Intelligence, vol. 29, no. 4, 2007, pp. 627-639.

22. V. Štruc, N. Pavešic, “Photometric normalization techniques for illumination invariance”, in: Y.J. Zhang (Ed), Advances in Face Image Analysis: Techniques and Technologies, IGI Global, 2011. pp. 279-300.

23. V. Štruc, N. Pavešic, “Gabor-based kernel-partial-least-squares discrimination features for face recognition”, Informatica (Vilnius), vol. 20, no. 1, pp. 115-138, 2009.

24. S. Ferdinando, H. Andy, “Parameterisation of a Stochastic Model for Human Face Identification.” Proceedings of 2nd IEEE Workshop on Applications of Computer Vision, Sarasota FL, December 1994.

25. A. S. Georghiades, and P.N. Belhumeur, and D.J. Kriegman, "From Few to Many: Illumination Cone Models for Face Recognition under Variable Lighting and Pose", "IEEE Trans. Pattern Anal. Mach. Intelligence", volume 23, number 6, pages "643-660",2001

26. K. Kyungnam “Face Recognition using Principle Component Analysis” Department of Computer Science University of Maryland, College Park MD 20742, USA

27. Definition of Nearest Neighbor on wikipedia. Retrived on 4th September 2012 http://www.wikipedia.com/nearest%20neighbor.

28. X. Tan, and B. Triggs, “Enhanced local texture feature sets for face recognition under difficult lighting conditions,” In Proceedings of the IEEE international workshop on analysis and modeling of faces and gestures, pp. 168-182. 2009.

29. C. M. Bishop, Pattern Recognition and Machine Learning, vol. 4, no. 4. Springer, 2006, p. 738.

30. U. K Jaliya et al, “A Comparative Study of PCA & LDA Human Face Recognition Methods”, National Conference on Recent Trends in Engineering & Technology, B.V.M. Engineering College, V.V.Nagar,Gujarat,India. 2011.

31. Martin, Alberto, and Sabri Tosunoglu. "Image processing techniques for machine vision." Miami, Florida 2000.

32. R. Jafri and H. R. Arabnia, “A Survey of Face Recognition Techniques,” Journal of Information Processing Systems, vol. 5, no. 2, pp. 41–68, Jun. 2009.

33. X. Zou, J. Kittler, and K. Messer, “Illumination Invariant Face Recognition: A Survey,” 2007 First IEEE International Conference on Biometrics: Theory, Applications, and Systems, pp. 1–8, Sep. 2007.

34. Ruiz-del-Solar, J., Quinteros, J., Comparing Pre-Processing Approaches for Illumination Invariant Face Recognition, Technical Report UCH-DIE-VISION-2006-04, Dept. Elect. Eng., Universidad de Chile. Available (August 2006) in: http://vision.die.uchile.cl/

35. P. Hallinan, “A low-dimensional representation of human faces for arbitrary lighting conditions,” In Proc. IEEE Conf. on Comp. Vision and Patt. Recog. pp 995–999, 1994.

36. A.U. Batur and M. H. III. Hayes. Linear subspaces for illumination robust face recognition. In Proc. IEEE conf. CVPR, 2001.

37. R.Basri and D. Jacobs. Lambertian reflectance and linear subspaces. In Proc. IEEE ICCV, 2001.

38. R.Basri and D. Jacobs. Lambertian reflectance and linear subspaces. IEEE Trans. PAMI, 25(2):218–233, 2003.

39. R. Ramamoorthi and P. Hanrahan. On the relationship between radiance and irradiance: Determining the illumination from images of a convex lambertain object. Journal of Optical Society of American,18 (10):2448–2459, 2001.

40. A. S. Georghiades, P. N. Belhumeur, and D. J. Kriegman. From fewto many: Illumination cone models for face recognition under variable lighting and pose. IEEE Trans. Pattern Anal. Mach. Intelligence, 23(6):643–660, 2001.

41. L. Zhang and D. Samaras. Face recognition under variable lighting using harmonic image exemplars. In Proc. IEEE Computer Society conf. CVPR, 2003.

42. L. Zhang and D. Samaras. Face recognition from a single training image under arbitrary unknown lighting using spherical harmonics. IEEE Trans. PAMI, 29, 2006.

43. S. Wei and S. Lai. Robust face recognition under lighting variations. In Proc. ICPR, pp1051-4651 2004.

44. C. Yang, S. Lai, and L. Chang. Robust face matching under different lighting conditions. EURASIP Journal on App. Sig. Proc., vol.2, pp 49 - 152 2004.

45. T. Sim and T. Kanade. Combining models and exemplars for face recognition: An illumination example. In Proc. Workshop on Models versus Exemplars in Computer Vision, CVPR, 2001.

46. T. Ojala, M. Pietikainen, D. Harwood, "A comparative study of texture measures with classification based on features distributions", Pattern Recognition, vol. 29, No. 1, 1996, pp51-59.

47. S. Shan, W., Gao, B., Cao, D., Zhao, "Illumination normalization for robust face recognition against varying lighting conditions", In: Proc. IEEE Workshop on AMFG, 2003, pp157-164.

48. V. Štruc, J. Žibert, and N. Pavešic, "Histogram remapping as a preprocessing step for robust face recognition", WSEAS transactions on information science and applications, vol. 6, no. 3, pp. 520-529, 2009."

49. C. A. Poynton . "Digital Video and HDTV: Algorithms and Interfaces". Morgan Kaufmann. 2003, pp. 260, 630. ISBN 1-55860-792-7. April, 18. 2013 [April, 20 2013].

50. B.C., Becker, E.G Ortiz. "Evaluation of Face Recognition Techniques for Application to Facebook," in Proceedings of the 8th IEEE International Automatic Face and Gesture Recognition Conference, 2008.

51. W. Gao, S. Member, B. Cao, S. Shan, X. Chen, D. Zhou, X. Zhang, and D. Zhao, "The CAS-PEAL Large-Scale Chinese Face Database and Baseline Evaluations," vol. 38, no. 1, 2008, pp. 149–161.

52. K. C. Lee, J. Ho, and D. Kriegman."Nine points of lights: Acquiring subspaces for face recognition under variable lighting". In Proc. IEEE conf. CVPR, 2001.

53. A. Shashua. "On photometric issue in 3d visual recognition from a single 2d image". IJCV, 1997.

54. S. Zhou, G. Aggarwal, R. Chellappa, and D. Jacobs. “Appearance characterization of linear lambertian objects, generalized photometric stereo and illumination-invariant face recognition”. IEEE Trans. PAMI, 29(2):230–245, 2007.

55. K. Lee, J. Ho, and D. Kriegman, “Acquiring linear subspaces for face recognition under variable lighting,” IEEE Trans. Pattern Analysis & Machine Intelligence, vol. 27, no. 5, pp. 684–698,2005.

56. X. Xie and K. Lam. An efficient illumintion normalization method for face recognition. Pattern Recognition Letters, 27,pp 609–617, 2005.

57. V. Blanz and T. Vetter. Face recognition based on fitting a 3d morphable model. IEEE Trans. PAMI, 25(9):1063–1073, 2003.

58. C. Zhang, and F. Cohen, “3-D face structure extraction and recognition from images using 3-D morphing and distance mapping,” IEEE Trans. Image Process., vol.11, no11, pp. 1249-1259. 2002.

59. MATLAB Support - Documentation and Helpdesk. Retrieved 22 June 2012, from http://www.mathworks.com/access/helpdesk/help/techdoc/matlab.shtml

APPENDICES

APPENDIX A: GAMMA CORRECTION

% The function performs gamma correction on the input image X

% PROTOTYPE Y=gamma_correction(X, in_interval, out_interval, gamma);

% USAGE EXAMPLE(S)

% Example 1:

% X=imread('sample_image.bmp');

% Y=gamma_correction(X, [0 1], [0 1], 0.2);

% figure,imshow(X);

% figure,imshow((Y),[]);

% INPUTS:

% X - a grey-scale image of arbitrary size

% in_interval - a two component vector defining the input interval from which gamma correction is % applied; this term is used to enable a linear shift of the input values, e.g., [0 1]. If the value of the % interval is given with empty brackets, i.e., [], then no adjustment is performed.

% out_interval - a two component vector defining the output interval of the gamma correction; when

% gamma - the gamma value; i.e., the image is transformed using: X.^gamma

% OUTPUTS:

% Y - a gamma corrected grey-scale image with its dynamic range adjusted to span the % interval defined by the parameter "out_interval",

% The function was tested with Matlab ver. 7.5.0.342 (R2007b) and Matlab ver. 7.11.0.584 (R2010b).

% ABOUT

% Created: 19.8.2009 Last Update: 26.1.2012 Revision: 2.1

% WHEN PUBLISHING A PAPER AS A RESULT OF RESEARCH CONDUCTED BY USING THIS CODE OR ANY PART OF IT, MAKE A REFERENCE TO THE FOLLOWING PUBLICATIONS:

% 1.Struc V., Parvic, N.:Photometric normalization techniques for illumination invariance, in: Y.J. % Zhang (Ed), Advances in Face Image Analysis: Techniques and Technologies, IGI Global, pp. 279- % 300, 2011.

% (BibTex available from: http://luks.fe.uni-lj.si/sl/osebje/vitomir/pub/IGI.bib) 2. struc, V., Paveوic, % N.: Gabor-based kernel-partial-least-squares discrimination features for face recognition, % Informatica (Vilnius), vol. 20, no. 1, pp. 115-138, 2009. (BibTex available from: http://luks.fe.uni- % lj.si/sl/osebje/vitomir/pub/InforVI.bib)

```
% January 2012

function Y=gamma_correction(X, in_interval, out_interval, gamma);

%% default return value

Y=[];

%% Parameter check

if nargin~=4

   disp('Error: The function takes exactly four arguments.');

   return;

end

%% Init. operations

X=double(X);   [a,b]=size(X);

%% Map to input interval

if ~isempty(in_interval)

    if length(in_interval)==2  X=adjust_range(X,in_interval);

    else  disp('Error: Input interval needs to be a two-component vector.'); return; end

end

%% Do gamma correction

X=X.^gamma;

%% Map to output interval

if ~isempty(out_interval)

    if length(out_interval)==2

        Y=adjust_range(X,out_interval);

    else

       disp('Error: Output interval needs to be a two-component vector.');

       return; end

end
```

HISTOGRAM REMAPPING USING NORMAL AND LOG-NORMAL DISTRIBUTION

```
%FITT_DISTRIBUTION: The function fitts a predefined distribution to the histogram of an image.
% PROTOTYPE       Yy=fitt_distribution(X,distr,param);
% USAGE EXAMPLE(S)
%    Example 1:
%      X=imread('sample_image.bmp');
%      Y=fitt_distribution(X);
%      figure,imshow(X);
%      figure,imshow(Y,[]);
%    Example 2:                              %    Example 3:
%      X=imread('sample_image.bmp');         %      X=imread('sample_image.bmp');
%      Y=fitt_distribution(X,2);             %      Y=fitt_distribution(X,2,[0 0.2]);
%      figure,imshow(X);                     %      figure,imshow(X);
%      figure,imshow(Y,[]);                  %      figure,imshow(Y,[]);
% GENERAL DESCRIPTION
% INPUTS:
% X            - a grey-scale image of arbitrary size
% distr        - a scalar value determining the target distribution, valid options are
%              1   - the target distribution is normal (default)       2   - the target distribution is lognormal
%              3   - the target distribution is exponential
% param        - a vector or scalar determining the parameters of the target distribution,
%             for the implemented target distributions:
%             normal      - [mean,std], e.g. [0,1] (default)      lognormal   - [mean,std], e.g. [0,0.2] (default)
%             exponential - [labmda], e.g. [1] (default)
% OUTPUTS:
% Y            - a grey-scale image with a remapped histogram
function Yy=fitt_distribution(X,distr,param);
%% Default return parameter Yy=[];
```

```
%% Parameter check

if nargin ==1    distr = 1; param = [0 1];

elseif nargin == 2  if distr == 1       param = [0 1];

                          elseif distr == 2  param = [0 0.2];

                          elseif distr == 3  param = 1;

 else               param=1;

 end

elseif nargin >3        disp('Error! Wrong number of input parameters.')

return;   end

if distr == 1 || distr == 2      [a,b]=size(param);

 if a ~=1 || b~=2       disp('Error: The normal and lognormal distributions require two parameters in the
parameter vector (size: 1 x 2).')

 return;   end

elseif distr == 3       [a,b]=size(param);

  if a ~=1 || b ~= 1

      disp('Error: The exponential distribution requires one and only one parameter.')

      return;  end

else  disp('Error: Wrong value for parameter "distr".')

   return end

%% Init. operations

[a,b]=size(X);  X=normalize8(X);  n=a*b;

%% Fitt distribution

if distr == 1

   R=rank_normalization(X(:), 'two','descend')';

   left = (n+0.5-(R))/n;

   Yy=norminv(left,param(1,1),param(1,2));

elseif distr ==2

   R=rank_normalization(X(:), 'two','descend')';

   left = (n+0.5-(R))/n;

   Yy=icdf('Lognormal',left,param(1,1),param(1,2));

elseif distr ==3
```

```
    R=rank_normalization(X(:), 'two','ascend')';
    left = (n+0.5-(R))/n;
    Yy=icdf('exp',left,param(1,1));
    Yy = 255-normalize8(Yy);
end
Yy = normalize8(reshape(Yy,[a,b]));
```

APPENDIX C: ANISOTROPIC SMOOTHING

```
% The function applies the anisotropic smoothing normalization technique to an image
% PROTOTYPE          [R,L] = anisotropic_smoothing(X,param,normalize);
% USAGE EXAMPLE(S)
%    Example     1:                          %    Example 2:
%      X=imread('sample_image.bmp');         %      X=imread('sample_image.bmp');
%      Y = anisotropic_smoothing(X);         %      Y = anisotropic_smoothing(X,20);
%      figure,imshow(X);                     %      figure,imshow(X);
%      figure,imshow((Y),[]);                %      figure,imshow((Y),[]);
% REFERENCES
% This function is an implementation of the anisotropic smoothing technique
% described in:  R. Gross, and V. Brajovic, "An Image Preprocessing Algorithm for Illumination
% Invariant Face Recognition," in: Proc. of the 4th International Conference on  Audio- and Video-
% Based Biometric Personal Authentication, AVPBA'03, July    % 2003, pp. 10-18.
% INPUTS:
% X                - a grey-scale image of arbitrary size
% param            - a scalar value controling the relative importance of the smoothness
%     constraint, in the papers on diffusion processes this parameter is usually denoted
%      as "lambda", default value "param=7"
% normalize        - a parameter controlling the post-processing procedure:
%         0 - no normalization      1 - perform basic normalization (truncation of histograms ends and
%        of histograms ends and normalization to the 8-bit interval) - default
% OUTPUTS:
% R      - a grey-scale image processed with anisotropic smoothing (the reflectance)
```

```
% L        - the estimated luminance function


function [R,L] = anisotropic_smoothing(X,param,normalize);
%% Init
R=[];   L=[];
%% Parameter checking
if nargin == 1 param = 7; normalize=1;
   elseif nargin == 2   if isempty(param) param = 7;
end
    normalize=1;
 ``````````````````````````````elseif nargin == 3;
 if isempty(param)  param = 7; end
if ~(normalize==1 || normalize==0)
 disp('Error: The third parameter can only be 0 or 1.'); return; end
elseif nargin >3    disp('Error! Wrong number of input parameters.')  return;
end
%% Init. operations
X = padarray(normalize8(X),[3,3],'symmetric','both');
[a,b]=size(X);   X=normalize8(X,0);    X=double(X+0.001);
im = zeros(a+2,b+2);    im(2:end-1,2:end-1)=X;
%% Compute contrast
pw1x = zeros(a+2,b+2);   pe1x = zeros(a+2,b+2);
ps1x = zeros(a+2,b+2);   pn1x = zeros(a+2,b+2);
%this we could do more elegantly but wont
[rows,cols]=size(im);
for i=1:rows
    for j=1:cols
        if (i>1) & (i<rows) & (j>1) & (j<cols)
            %get pixels
            E=im(i,j+1);           A=im(i,j);   S=im(i+1,j);   W=im(i,j-1);     N=im(i-1,j);
```
```

```
            %Michelson's contrast inverse
            pw1x(i,j) = (A-W)/(abs(A+W)+0.001);
            pe1x(i,j) = (A-E)/(abs(A+E)+0.001);
            pn1x(i,j) = (A-N)/(abs(A+N)+0.001);
            ps1x(i,j) = (A-S)/(abs(A+S)+0.001);
           end end end
end  pw1 = pw1x(2:end-1,2:end-1);       pe1 = pe1x(2:end-1,2:end-1);
ps1 = ps1x(2:end-1,2:end-1);                 pn1 = pn1x(2:end-1,2:end-1);
pw1=normalize8(pw1,0)+0.001;            pe1=normalize8(pe1,0)+0.001;
ps1=normalize8(ps1,0)+0.001;              pn1=normalize8(pn1,0)+0.001;
%% Define the contrast
I = zeros(a*b,1);                pw11 = zeros(a*b,1);
pe11 = zeros(a*b,1);        pn11 = zeros(a*b,1);
ps11 = zeros(a*b,1);        ps11 = zeros(a*b,1);        ps11 = zeros(a*b,1);
counter=1;
for i=1:a
   for j=1:b
      I(counter,1) = X(i,j);             pw11(counter,1) = pw1(i,j);
      ps11(counter,1) = ps1(i,j);     pe11(counter,1) = pe1(i,j);
      pn11(counter,1) = pn1(i,j);    counter=counter+1;
      end
end
clear  pw1 ps1 pe1 pn1 pw1x ps1x pe1x pn1x
%% Construction of sparse matrix S - in diagonal blocks of axb
x_index = zeros(1,3*(a-1)*a*b+a*b);   y_index = zeros(1,3*(a-1)*a*b+a*b);
s_value = zeros(1,3*(a-1)*a*b+a*b);
cont=1; for p=1:a
   %for main-diagonal block
   small_diag = zeros(a,b);      block_num = p;
   for i=1:a
```

```
        for j=1:b

            param_location=(block_num-1)*b+j;
k=(1+param*(pw11(param_location,1)+ps11(param_location,1)+pe11(param_location,1)+pn11(param
_location,1)));

            if j==1 & j==i

                small_diag(i,j) = k;

                small_diag(i,j+1) = -param*pe11(param_location,1);

            elseif j~=1 & j~=b & j==i

                small_diag(i,j) = k;

                small_diag(i,j+1) = -param*pe11(param_location,1);

                small_diag(i,j-1) = -param*pw11(param_location,1);

            elseif j==b & j==i

                small_diag(i,j) = k;

                small_diag(i,j-1) = -param*pw11(param_location,1);

            end end  end

    %the above-main-diagonal block

    if block_num>1   above_diag = zeros(a,b);

         for i=1:a

            for j=1:b    param_location = (block_num-1)*b+j;

                if j==i

                    above_diag(i,j) = -param*(ps11(param_location,1));

                end         end         end         end

    %the below-main-diagonal block

    if block_num>1

        below_diag = zeros(a,b);

        for i=1:a

            for j=1:b

                param_location = (block_num-2)*b+j;

                if j==i

                    below_diag(i,j) = -param*(pn11(param_location,1));

                end         end         end         end
```

```
    if block_num==1
        [ind_y,ind_x]=meshgrid(((p-1)*b+1):(p*b),((p-1)*a+1):(p*a));
        leng = numel(ind_x);
        x_index(1,(cont-1)*leng+1:cont*leng) = ind_x(:)';
        y_index(1,(cont-1)*leng+1:cont*leng) = ind_y(:)';
        s_value(1,(cont-1)*leng+1:cont*leng) = small_diag(:)';
        cont=cont+1;
    else
        %main diagonal
        [ind_y,ind_x]=meshgrid((p-1)*b+1:p*b,(p-1)*a+1:p*a);
        leng = numel(ind_x);
        x_index(1,(cont-1)*leng+1:cont*leng) = ind_x(:)';
        y_index(1,(cont-1)*leng+1:cont*leng) = ind_y(:)';
        s_value(1,(cont-1)*leng+1:cont*leng) = small_diag(:)';
        cont=cont+1;
        %above diagonal
        [ind_y,ind_x]=meshgrid((p-2)*b+1:(p-1)*b,(p-1)*a+1:p*a);
        x_index(1,(cont-1)*leng+1:cont*leng) = ind_x(:)';
        y_index(1,(cont-1)*leng+1:cont*leng) = ind_y(:)';
        s_value(1,(cont-1)*leng+1:cont*leng) = above_diag(:)';
        cont=cont+1;
        %below diagonal
        [ind_y,ind_x]=meshgrid((p-1)*b+1:(p)*b,(p-2)*a+1:(p-1)*a);
        x_index(1,(cont-1)*leng+1:cont*leng) = ind_x(:)';
        y_index(1,(cont-1)*leng+1:cont*leng) = ind_y(:)';
        s_value(1,(cont-1)*leng+1:cont*leng) =  below_diag(:)';
        cont=cont+1;
    end     end
%% Construct sparse system and solve it using matlabs internal functions
S=sparse(y_index, x_index, s_value, a*b,a*b);
```

x=S\I;

%% Reshape result

tmp = reshape(x,[a b]); L = tmp'; L=L(4:end-3,4:end-3); tmp = X./tmp';

R=tmp(4:end-3,4:end-3);

%% Do some final post-processing (or not)

if normalize ~= 0   R = normalize8(histtruncate(R,0.4,0.4));   L=normalize8(L); end

APPENDIX D: DCT-BASED NORMALIZATION ALGORITHM

% The function applies the DCT-based normalization algorithm to an image.

% PROTOTYPE

% Y=DCT_normalization(X,numb,normalize);

% USAGE EXAMPLE

%   Example :

%   X=imread('sample_image.bmp');   %   Y=DCT_normalization(X);

%   figure,imshow(X);   %   figure,imshow((Y),[]);

% INPUTS:

% X   - a grey-scale image of arbitrary size

% numb   - a scalar value determining the number of DCT coefficients to

%   set to zero, default "numb=50"

% normalize   - a parameter controlling the post-processing procedure:

%   0 - no normalization   1 - perform basic normalization (truncation

%   of histograms ends and normalization to the 8-bit interval) - default

%   OUTPUTS:

% Y - a grey-scale image processed with the DCT-based normalization technique

function Y=DCT_normalization(X,numb, normalize);

%% Parameter checking

n if isempty(numb)ormalize = 1;Y=[];%dummy if nargin == 1 numb = 50; elseif nargin == 2 numb = 50; end

normalize = 1;

elseif nargin == 3

if isempty(numb)

```
        numb = 50;    end
    if ~(normalize==1 || normalize==0)
        disp('Error: The third parameter can only be 0 or 1.');
        return;     end
elseif nargin > 3
    disp('Eror: Wrong number of input parameters.')    return; end
[a,b]=size(X);
if numb > a*b
    disp('Error! The number of DCT coeffcients to discard cannot be larger than the numbr of pixels in the image.')
    return; end
%% Init. operations
X=normalize8(X);
[a,b]=size(X); M=a; N=b; coors = do_zigzag(X);
%% Transform to logarithm and frequency domains
X=log(X+1);
X=normalize8(X);
means= mean(X(:))+10; %we chose a mean near the true mean (the value +10 can be changed)
Dc = dct2(X);
%% apply the normalization
c_11=log(means)*sqrt(M*N);
Dc(1,1)=c_11;
for i=2:numb+1    ky = coors(1,i);
    kx = coors(2,i);  Dc(ky,kx) = 0;
end
Y=(idct2(Dc));
%% Do some post-processing (or not)
if normalize ~=0
    Y=normalize8(histtruncate(normalize8(Y),0.2,0.2)); end
%% This function produces the zigzag coordinates
function output = do_zigzag(X);
```

```
%init operations
h = 1; v = 1; vmin = 1; hmin = 1; vmax = size(X, 1); hmax = size(X, 2);
i = 1;
output = zeros(2, vmax * hmax);
%do the zigzag
while ((v <= vmax) & (h <= hmax))
    if (mod(h + v, 2) == 0)
        if (v == vmin)
            output(:,i) = [v;h];
            if (h == hmax)      v = v + 1;
                  else              h = h + 1;
            end;            i = i + 1;
        elseif ((h == hmax) & (v < vmax))
            output(:,i) = [v;h];              v = v + 1;  i = i + 1;
        elseif ((v > vmin) & (h < hmax))
            output(:,i) = [v;h];
            v = v - 1;
            h = h + 1;
            i = i + 1;  end
    else
       if ((v == vmax) & (h <= hmax))       output(:,i) = [v;h];              h = h + 1;
            i = i + 1;
       elseif (h == hmin)     output(:,i) = [v;h];        if (v == vmax)
             h = h + 1;    else            v = v + 1;
       end;
            i = i + 1;
       elseif ((v < vmax) & (h > hmin))     output(:,i) = [v;h];
                    v = v + 1;           h = h - 1;
 i = i + 1; end  end
    if ((v == vmax) & (h == hmax))
```

```
        output(:,i) = [v;h];
        break  end
end
```

APPENDIX E: NEAREST NEIGHBOR CLASSIFICATION

```
% This value now comes from fbInit
topX = fbgCountTopX;
% we "recognize" an image by assigning it the identity of the closest matching training face.
testlen = size(testWeights,2);  trainlen = size(trainWeights,2);
index = zeros(testlen,topX);  resultIds = zeros(testlen,topX);
resultDist=zeros(testlen,topX);x2 = sum(testWeights.^2)'; y2 = sum(trainWeights.^2);
for i = 1:testlen
   % Using Euclidean distance, I add this part.
   %z = testWeights(:,i)'*trainWeights;
   %z = repmat(x2(i),1,trainlen) + y2 - 2*z;
   %[C, index(i)] = min(z);
   %Using cosine distance
   %strcmp(dist,'cos')==1
   norm_x = norm(testWeights(:,i));
   norm_y = norm(trainWeights);
   z = - (testWeights(:,i)'*trainWeights)/(norm_x*norm_y);
   for j = 1:topX
               [best, index(i, j)] = min(z);
               resultIds(i, j) = fbgTrainIds(index(i, j));
               resultDist(i, j) = best;
               z(index(i, j)) = Inf; % Remove best
               % Keep searching until we have a new person because we the first
               % two hits might be the same person (wrong code)
               if j > 1
                       while sum(resultIds(i, 1:j-1) == resultIds(i, j)) > 0
                               [best, index(i, j)] = min(z);
```

```
                        resultIds(i, j) = fbgTrainIds(index(i, j));
                        resultDist(i, j) = best;
                        z(index(i, j)) = Inf; % Remove best
                end   end  end  end
% Store the number of correct
if topX > 1
    resultMatrix = (resultIds == repmat(fbgTestIds, 1, topX));
    results = max(resultMatrix, [], 2);
else
    resultMatrix = (resultIds == fbgTestIds);
    results = max(resultMatrix, [], 2);
end
correct = find(results == 1);
fbgAccuracy = 100* length(correct) / size(fbgTestIds,1);
% Only do this if we are not also
if fbgMakeAccuracy > 0 && topX == 1
            t = fminsearch(@(t) testThreshold(t, correct, resultDist, fbgMakeAccuracy, results), mean(resultDist));
            [accuracy, left, keepIndex] = testThreshold(t, correct, resultDist, 0, results);
            [accuracy, left, t / 1e4];
            fbgAccuracy = accuracy*100;
            fprintf('By thresholding, we have eliminated %0.1f%% of the faces.\n', 100*(1-left));
% Should remove the ignored faces here so HTML output is correct
        resultIds = resultIds(keepIndex, :);  index = index(keepIndex, :);
         resultDist = resultDist(keepIndex, :); resultMatrix = resultMatrix(keepIndex, :);
        testFiles = fbgTestFiles(keepIndex);
else   testFiles = fbgTestFiles;   end
% If we want we can generate the number of correct results
if fbgGenHTMLResults  genHTMLResults;  end
```