8-2015

Face Recognition Under Varying Illuminations

Mohammadreza Faraji
Utah State University

FACE RECOGNITION UNDER VARYING ILLUMINATIONS

by

Mohammadreza Faraji

A dissertation submitted in partial fulfillment
of the requirements for the degree

of

DOCTOR OF PHILOSOPHY

in

Computer Science

Approved:

Dr. Xiaojun Qi Major Professor	Dr. Nicholas Flann Committee Member
Dr. Vladimir Kulyukin Committee Member	Dr. Minghui Jiang Committee Member
Dr. Adele Cutler Committee Member	Dr. Mark R. McLellan Vice President for Research and Dean of the School of Graduate Studies

UTAH STATE UNIVERSITY
Logan, Utah

2015

Abstract

Face Recognition Under Varying Illuminations

by

Mohammadreza Faraji, Doctor of Philosophy

Utah State University, 2015

Major Professor: Dr. Xiaojun Qi
Department: Computer Science

This dissertation proposes four effective methods to produce illumination-invariant features for images with various levels of illuminations. The first proposed method presents a logarithmic fractal dimension (LFD) method by employing the log function and the fractal analysis to produce illumination-invariant representations. The LFD method is also an effective edge enhancer technique to extract and enhance facial features such as eyes, eyebrows, nose, and mouth. The second method proposes a modified version of local directional patterns (LDP), eight local directional patterns (ELDP), which uses Kirsch compass masks to compute the edge responses of a pixel's neighborhood. Then, the ELDP code scheme uses all the directional numbers to produce an illumination-invariant image. The third method, called adaptive homomorphic eight local directional patterns (AH-ELDP), first, uses adaptive homomorphic filtering to reduce the influence of illumination from an input face image. Secondly, it applies an interpolative enhancement function to stretch the filtered image. Finally, it applies the EDLP code scheme on the enhanced filtered image to create an illumination-invariant representation. The final method, complete ELDP (CELDP), seamlessly combines adaptive homomorphic filtering, simplified logarithmic fractal dimension, and complete ELDP to produce illumination-invariant representations. CELDP considers the relations among magnitudes as well as directions of all eight directional edge responses

to achieve robustness against illumination variations and noise. The extensive experimental results on four face databases and one video face database show that the final proposed CELDP method consistently achieves the best face recognition accuracy on all databases compared with six recent state-of-the-art methods and the three proposed methods. The AH-ELDP method achieves the second best face recognition on all image databases.

(101 pages)

Public Abstract

Face Recognition Under Varying Illuminations

by

Mohammadreza Faraji, Doctor of Philosophy

Utah State University, 2015

Major Professor: Dr. Xiaojun Qi
Department: Computer Science

Face recognition under illumination is really challenging. This dissertation proposes four effective methods to produce illumination-invariant features for images with various levels of illuminations. The proposed methods are called logarithmic fractal dimension (LFD), eight local directional patterns (ELDP), adaptive homomorphic eight local directional patterns (AH-ELDP), and complete eight local directional patterns (CELDP), respectively.

LFD, employing the log function and the fractal analysis (FA), produces a logarithmic fractal dimension (LFD) image that is illumination-invariant. The proposed FA feature-based method is an effective edge enhancer technique to extract and enhance facial features such as eyes, eyebrows, nose, and mouth.

The proposed ELDP code scheme uses Kirsch compass masks to compute the edge responses of a pixel's neighborhood. It then uses all the directional numbers to produce an illumination-invariant image.

AH-ELDP first uses adaptive homomorphic filtering to reduce the influence of illumination from an input face image. It then applies an interpolative enhancement function to stretch the filtered image. Finally, it produces eight directional edge images using Kirsch compass masks and uses all the directional information to create an illumination-insensitive representation.

CELDP seamlessly combines adaptive homomorphic filtering, simplified logarithmic fractal dimension, and complete eight local directional patterns to produce illumination-invariant representations.

Our extensive experiments on Yale B, extended Yale B, CMU-PIE, and AR face databases show the proposed methods outperform several state-of-the-art methods, when using one image per subject for training.

We also evaluate the ability of each method to verify and discriminate face images by plotting receiver operating characteristic (ROC) curves which plot true positive rates (TPR) against the false positive rates (FPR).

In addition, we conduct an experiment on the Honda UCSD video face database to simulate real face recognition systems which include face detection, landmark localization, face normalization, and face matching steps. This experiment, also, verifies that our proposed methods outperform other state-of-the-art methods.

To my parents, my brothers and sisters, and my nephews and nieces.

Acknowledgments

I would like to express my deepest gratitude to my supervisor **Dr. Xiaojun Qi**, for her patient guidance, encouragement, and advice throughout my time as her student. Her continued support and invaluable advice were key motivations thoughout my doctoral program. I would like to extend my appreciation to my committee members, **Dr. Nicholas Flann**, **Dr. Vladimir Kulyukin**, **Dr. Minghui Jiang**, and **Dr. Adele Cutler**, for their invaluable advice and feedback which improved the quality of this research. I would also like to thank my dearest friends and my labmates for their scientific and moral support. Finally, none of this would ever have transpired without the unselfish love and support of my precious family. My parents, my brothers and sisters, and my nephews and nieces are unshakable sources of support, understanding, and love. I have been fortunate to have the love and affection of a most wonderful family, especially my parents, who have always encouraged me to follow my dreams. Their constant support, patient, love, and care and unswerving faith have gotten me where I am. This dissertation is dedicated to them.

Mohammadreza Faraji

Contents

List of Tables

List of Figures

Acronyms

AH-ELDP	adaptive homomorphic eight local directional pattern
CELDP	complete eight local directional pattern
ELDP	eight local directional pattern
EnLDP	enhanced local directional pattern
FA	fractal analysis
FD	fractal dimension
FN	false negative
FP	false positive
FPR	false positive rate
LBP	local binary pattern
LDP	local directional pattern
LDN	local directional number pattern
LFD	logarithmic fractal dimension
NN	nearest neighbor
ROC	receiver operating characteristic
SRC	sparse representation-based classification
TN	true negative
TP	true positive
TPR	true positive rate
1NN	one nearest neighbor

Chapter 1

Face Recognition under Varying Illuminations

1.1 Face recognition

Body and behavioral characteristics such as fingerprints, iris, finger/palm, voice, signature, and face are often called biometrics [1]. Today, any application that requires security and access control can be integrated with biometric systems and technologies, which makes the biometric field very attractive [1]. Fig. 1.1 [1] shows the distribution of the most popular biometrics from a commercial point of view. Clearly, fingerprints are the most popular biometrics in the last decade with 52% of the distribution. However, some biometrics have shortcomings. For example, iris recognition is extremely accurate, but implementing an iris recognition system is expensive. Fingerprint recognition systems are reliable and non-intrusive, but need individuals' collaboration. Face recognition is a good compromise between reliability and social acceptance. It balances security and privacy well and can be employed in places with a large concourse of unaware or non-collaborative visitors [1].

Fig. 1.2 shows a framework for a typical face recognition system [2] which includes four main stages: Face Detection, Landmark Localization, Face Normalization, and Face Matching. These stages are briefly explained below.

Face Detection: In the first stage, faces are detected using face detectors such as SNoW-based (sparse network of windows) face detector [3], Viola-Jones detector [4], and skin-color-pixel-based Viola-Jones face detector [5]. The SNoW-based face detector is a sparse network of linear units (functions) over a common predefined or incrementally learned feature space. Viola-Jones detector, the most popular detector, is a robust real-time face detector to distinguish faces from non-faces. The detector has four main stages including Haar-like Feature Selection, Integral Image Creation, Adaboost Training algorithm, and

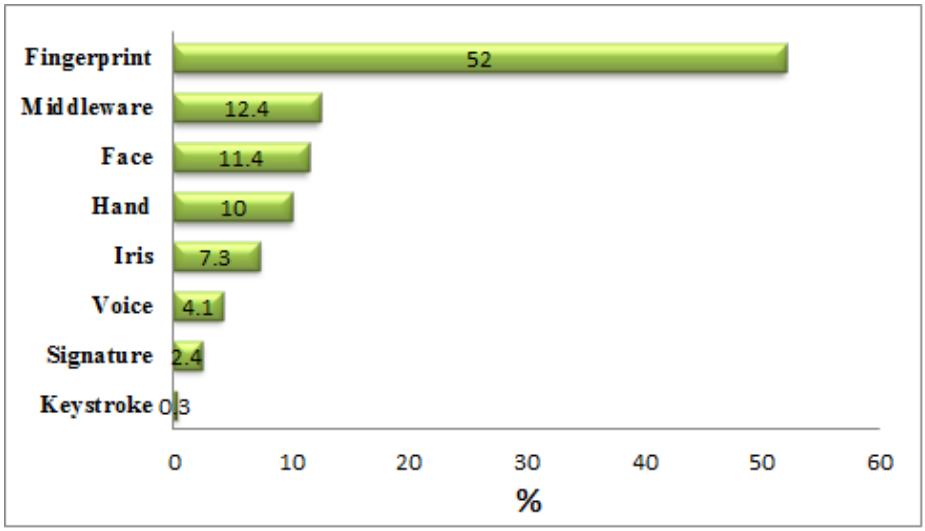

Fig. 1.1: The distribution of the most popular biometrics in terms of percentage.

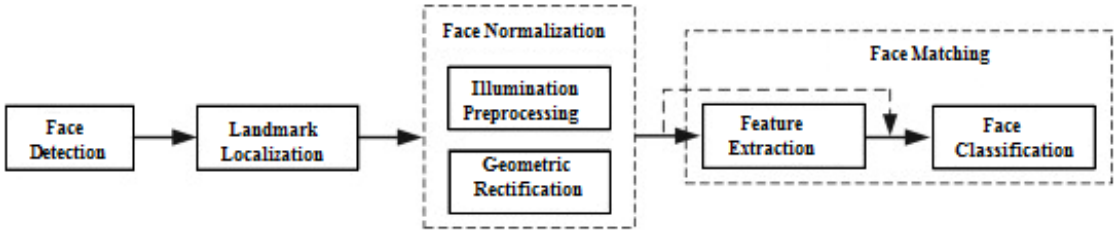

Fig. 1.2: Typical framework for illumination-invariant face recognition [2].

Cascaded Classifiers. Haar-like features are reminiscent of Haar basis functions [6] which use the idea of Haar wavelets. They consider adjacent rectangular regions at a specific location in a detection window which moves over the input image. Haar-like features sum up the intensity values in each region and calculate the difference between these sums to categorize subsections such as eye and cheek of a face. The Integral Image (also known as the summed area table) efficiently computes any scale or location of these features in a constant time. Since the total number of extracted Haar-like features is very large, a simple modification of the AdaBoost procedure [7] is used to exclude a large majority of the Haar-like features and focus on a small set of critical features. Viola-Jones detector, finally, employs a fast cascaded classifier which focuses attention on promising regions of

the image (i.e., where an object might occur) to distinguish faces from non-faces. The skin-color-pixel-based Viola-Jones face detector combines a hybrid skin color model and the Viola-Jones face detector. The hybrid skin color model, called RGB-CbCrCg, classifies skin and non-skin pixels. The skin region is first extracted using a set of bounding rules and then optimized by employing the multi-objective differential evolution method. The segmented face regions are finally identified using the Viola-Jones algorithm.

Landmark Localization: In the second stage, respective facial landmarks are localized for each detected face. These landmarks usually are locations of eyes, eyebrows, nose, and mouth. Fig. 1.3 shows regions in a face which contain facial key point features or landmarks. In the last decade, various facial landmark localization techniques have been proposed. Some of recently proposed techniques include Boosted Regression with Markov Networks (*BoRMaN*) [8], Local Evidence Aggregation for Regression Based Facial Point Detection (*LEAR*) [9], Discriminative Response Map Fitting (*DRMF*) [10], and *Chehra v.1* (meaning "face" in Hindi) [11].

LEAR detects 20 facial landmarks (Fig. 1.4(a)). It can use every location in a point's neighbourhood to predict where the target point is relative to that location. *LEAR* employs a stochastic process to obtain a precise prediction of the target location. It combines different target predictions into a function by aggregation of Gaussian distributions. The maximum of this function corresponds to the target. This aggregation of Gaussian distributions cancels out the effect of incorrect estimates to generate correct estimates since incorrect estimates produce more uncorrelated predictions. *LEAR* then finds facial points in a few iterations. Finally, it uses Markov networks to confirm whether a set of predicted facial points adhere to shape statistics. If they do not follow shape statistics, Markov networks suggest new candidate locations to continue the search for facial points.

Chehra, however, locates 49 facial landmarks (Fig. 1.4(b)). It is an incremental formulation for a cascade linear regression face alignment framework, which can be used as tracking software as well as landmark localization software. This framework is a type of discriminative deformable face alignment framework and is a static generic model built

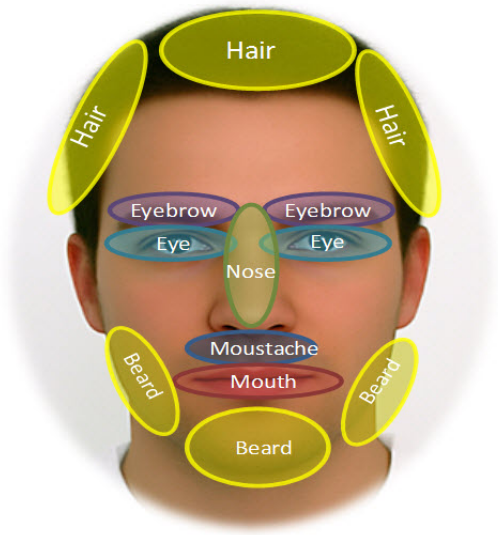

Fig. 1.3**:** Regions of interest in a face which include facial landmarks in a face (image source: http://www.andrew.cmu.edu/user/thihoanl/face.jpg).

completely from off-line training data. It uses an iteratively learned cascade of regression functions to directly map the textual features to shape. *Chehra* incrementally updates the cascade of linear regressions such that all levels in a cascade can be trained (and updated) independently using only the statistics of the previous level, instead of the previous iterations.

Face Normalization: In the third stage, detected faces are geometrically normalized and corrected by facial landmarks. First, *geometric rectification* is applied to normalize the pose. Pose normalization then uses facial landmarks to estimate the frontal view for each face [12]. To this end, the angle between the line passing the locations of the external corners of eyes and the horizontal axis can be used to compute the head's roll value and rotate the face accordingly. Then, the possibly rotated landmarks are further used to determine whether the face is the left/right/frontal view. In case the face is non-frontal, a frontal view can be estimated by stretching pixels. Face images can be automatically cropped using facial landmarks, since landmarks exactly indicate the locations of key facial points in the image. The *illumination preprocessing* method is finally applied on the cropped images to obtain illumination-insensitive face images. In this stage, if face images contain facial

Fig. 1.4**:** Facial landmark localization results by (a) *LEAR*, and (b) *Chehra*.

expressions such as smile, anger, and scream, Gabor feature-based expression normalization can be employed to extract facial features that are robust to expression variations [13–15].

Face Matching: In the final stage, an identification process is used on the normalized faces to recognize the identity of the faces. This stage includes *feature extraction* and *face classification*. In *feature extraction*, each face is converted to a one-dimensional vector to be used for classification. Each face might also be divided into a regular grid of cells to extract histograms [16]. Dimension reduction methods can also be used to reduce the dimension of the data before a classifier is employed to verify the identity of input face images. Representative dimension reduction methods include principal component analysis (PCA) [17–19], independent component analysis (ICA) [18, 20], kernel principal component analysis (KPCA) [21, 22], Fisher's linear discriminant analysis (FDA) [23], kernel Fisher's linear discriminant analysis (KFDA) [24, 25], and singular value decomposition (SVD). For example, PCA computes a subset of principal components in a set of training faces. It then projects face images into the space of these principal components to compute feature vectors. Finally, the *face classification* stage uses the extracted features to classify and verify a new unknown face image with respect to the image database. Some of the popular classification methods used in face recognition systems are k nearest neighbors [26, 27], neural-networks [28, 29], and sparse representation classifier (SRC) [30, 31].

For instance, the k nearest neighbor method uses a majority voting algorithm to classify a face image. It assigns the input face image to the most common subject among its k nearest

Fig. 1.5**:** Face images from the Yale B face database displaying illumination variations. Images are from the same subject [32].

neighbors. When $k = 1$, the 1 nearest neighbor method simply assigns the input face image to its nearest neighbor face image (i.e., subject) in the database. A distance measure (e.g., l_2 norm) is used to compute the similarity between the input face image and face images in the database. SRC finds a sparse representation of the test face image in terms of the training set as a whole by solving a simple convex program, up to some sparse error due to occlusion. The main idea of SRC is that the sparse nonzero coefficients should concentrate on the training samples which actually have the same identity as the test image.

1.2 Illumination-invariant methods

The performance of a face recognition system is significantly affected by five key factors including illumination variations, pose changes, facial expressions, age variations, and occlusions [1,34]. Figs. 1.5, 1.6, 1.7, 1.8, and 1.9 individually show samples of face images with the above five key factors illustrated. Illumination variations such as shadows, underexposure, and overexposure in face images, as one of the crucial problems among these factors, have attracted much attention in the face recognition community in the last decade [2]. As a result, to deal with illumination variations, various methods have been proposed which can be categorized into the following three groups, namely, gray-level transformation methods, gradient or edge extraction methods, and face reflection field estimation methods [2].

Gray-level transformation methods redistribute the intensities in a face image with

Fig. 1.6: Face images from the FEI face database displaying pose variations [33].

Fig. 1.7: Face images from the AT&T (formerly ORL) face database displaying expression variations [38].

Fig. 1.8: Face images from the FG-NET aging database displaying age variations. Each row shows images of the same subject [39, 40].

Fig. 1.9: Face images from the AR face database displaying occlusions [32].

a linear or non-linear transformation function to correct the uneven illumination to some extent [2]. Representative methods include histogram equalization (HE) [35], local histogram specification (LHS) [36], and gamma intensity correction (GIC) [37]. HE adjusts and flattens the contrast of the image using the image's histogram. LHS uses the ideal face images (i.e., captured under normal lighting conditions) to learn the local histograms and local histogram statistics. It then removes both the low and high frequency parts of illumination on face images, as well as enhancing face features lying in the low frequency part. GIC maps the image $I(x, y)$ to the image $G(x, y)$ using $\alpha I(x, y)^{1/\gamma}$, where α is a gray-stretch parameter and γ is the Gamma coefficient.

Gradient or edge extraction methods compute gradients of a face image as an illumination-insensitive representation since gradients are more stable than the pixel intensities under different lighting conditions [2]. Some examples of such methods include directional gray-scale derivative (DGD) [41], local binary patterns (LBP) [42] and its modified version local ternary patterns (LTP) [16], local directional patterns (LDP) [43], enhanced LDP (EnLDP) [44], local directional number patterns (LDN) [45], directional pattern of phase congruency (DPPC) [46], and discriminant face descriptor (DFD) [47]. DGD extracts gray-level gradients in horizontal and vertical directions which are robust to absolute intensity variations but are still greatly affected by severe shadows. LBP and LTP take a local neighborhood around each pixel and threshold the neighborhood pixels based on the value of the central pixel. LDP, EnLDP, and LDN use eight Kirsch compass masks $(M_0, M_1, ..., M_7)$ to produce eight directional edge images which are used to produce illumination-invariant representations. All the eight Kirsch masks as shown in Fig. 1.10 can be produced by rotating the first Kirsch mask (M_0) 45° apart in eight directions [48]. Fig. 1.11 illustrates code schemes for the LBP, LDP, EnLDP, and LDN approaches. DPPC replaces the intensity value required for calculating the traditional LDP with the phase congruency (PC) value at the corresponding pixel. The DFD method is an improvement of the LBP feature descriptor. It is involved with three-step feature extraction to maximize the appearance difference from different persons and minimize the difference from the same person.

$$\underset{M_0\,(0^\circ)}{\begin{bmatrix}-3 & -3 & 5\\ -3 & 0 & 5\\ -3 & -3 & 5\end{bmatrix}}\quad \underset{M_1\,(45^\circ)}{\begin{bmatrix}-3 & 5 & 5\\ -3 & 0 & 5\\ -3 & -3 & -3\end{bmatrix}}\quad \underset{M_2\,(90^\circ)}{\begin{bmatrix}5 & 5 & 5\\ -3 & 0 & -3\\ -3 & -3 & -3\end{bmatrix}}\quad \underset{M_3\,(135^\circ)}{\begin{bmatrix}5 & 5 & -3\\ 5 & 0 & -3\\ -3 & -3 & -3\end{bmatrix}}$$

$$\underset{M_4\,(180^\circ)}{\begin{bmatrix}5 & -3 & -3\\ 5 & 0 & -3\\ 5 & -3 & -3\end{bmatrix}}\quad \underset{M_5\,(225^\circ)}{\begin{bmatrix}-3 & -3 & -3\\ 5 & 0 & -3\\ 5 & 5 & -3\end{bmatrix}}\quad \underset{M_6\,(270^\circ)}{\begin{bmatrix}-3 & -3 & -3\\ -3 & 0 & -3\\ 5 & 5 & 5\end{bmatrix}}\quad \underset{M_7\,(315^\circ)}{\begin{bmatrix}-3 & -3 & -3\\ -3 & 0 & 5\\ -3 & 5 & 5\end{bmatrix}}$$

Fig. 1.10**:** Kirsch compass masks in eight directions.

Reflectance field estimation methods estimate the face reflectance field, which is illumination-invariant, from a face image. They usually apply the reflectance-illumination model [2] as their face imaging model. Self-quotient image (SQI) [49,50], single-light-region & single-dark-region SQI [51], logarithmic total variation (LTV) [52,53], logarithmic wavelet transform (LWT) [54], generic intrinsic illumination subspace [55], local normalization [56], adaptive smoothing (AdaS) [57], logarithmic discrete cosine transform (LDCT) [58], large & small logarithmic discrete cosine transform (L&S Log-DCT) [59], Gradientface [60], Weberface [61], and generalized Weberface (GWF) [62] are representative methods. SQI, which is a ratio image between a given test image and its smoothed version, implicitly indicates that each preprocessed image is illumination-invariant [63]. The single-light-region & single-dark-region SQI method separately processes bright and dark areas in the image by SQI without changing the essential characteristics of the face image. The LTV model improves SQI by utilizing the total variation (TV) model to factorize an image. It enhances the SQI's edge-preserving ability and lessens its parameter selection problem. LWT uses wavelet-based denoising techniques to reduce illuminations, while generating a multiscale facial structure. The generic intrinsic illumination subspace approach estimates the illuminance component using a bootstrap set. The approach assumes that the illumination images are from a specific ideal class and share the same generic intrinsic illumination subspace. The local normalization method treats a face image as a combination of a sequence of small and flat facets whose illumination is modeled by a multiplicative term and an additive term. Then, it applies a normalization technique to the image point by point which normalizes the intensity values within each facet to be of zero mean and unit variance. The

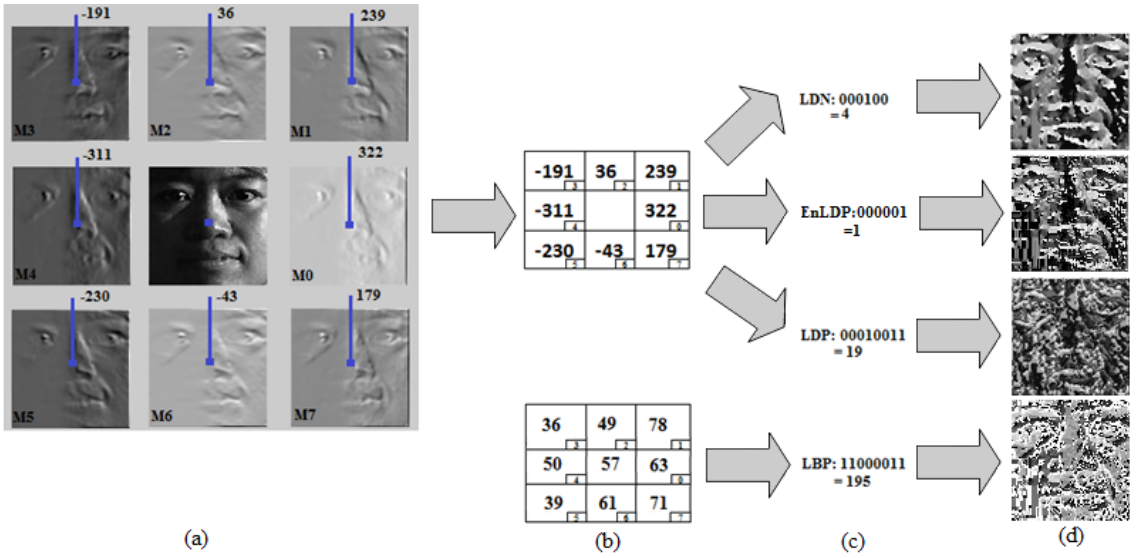

Fig. 1.11**:** Illustration of code schemes LBP, LDP, EnLDP, and LDN approaches. (a) Edge response images produced by convolving the original face image and Kirsch compass masks together with their response values at their specific same locations marked in blue dots. (b) Top: The LDN, EnLDP, and LDP window generated by the directional numbers at the same specified location of each response image. Bottom: The LBP window at the same specified location of the original image (the center pixel with an intensity value of 57). (c) The LDN, EnLDP, LDP, and LBP codes at the same specified location. (d) The final LDN, EnLDP, LDP, and LBP images generated by applying the same procedure to all locations of the original image.

AdaS method estimates the illumination by smoothing the input image using an iterative method. LDCT employs the discrete cosine transform (DCT) to compensate for illumination variations in the logarithmic domain. Considering illumination variations mainly lie in the low-frequency band, it truncates an appropriate number of DCT coefficients to minimize illumination variations. L&S Log-DCT decomposes the face image into large- and small-scale features. It then removes illumination variations on the large-scale features by setting some of the low-frequency DCT coefficients to be zero in the logarithmic domain. A minor correction is made on the small-scale features. Finally, L&S Log-DCT reconstructs an illumination-invariant image by combining the processed large- and small-scale features. The Gradientface method creates a ratio image between the y-gradient and the x-gradient of a given image. The Weberface method, inspired by Weber's law and based on the Weber local descriptor (WLD) [64], creates a ratio image between the local intensity variation and the background. The Gradientface and Weberface methods are proven to be illumination-

insensitive. The GWF method is a generalized multi-scale version of the Weberface method. It also considers an inner and outer ground for each pixel and assigns different weights for them to develop a weighted GWF (wGWF). Huang and Li [65] combine three distinct methods (i.e., homomorphic filtering, ratio image generation, and anisotropic smoothing) to preprocess each face image and then apply an illumination compensation method on the preprocessed image for the face recognition task.

Some researchers also classify illumination-invariant methods into three categories: illumination modeling, preprocessing and normalization, and illumination-invariant feature extraction [63].

The first category includes statistical or physical models in which the model parameters are extracted by low-dimensional linear subspaces [66]. However, the main limitations and shortcomings of this category of approaches are constructing a linear subspace and requiring several sample images for training [63].

Preprocessing and normalization approaches improve illumination conditions of images without using any face model or surface information. Since they are simple and feasible, they can be employed before other face recognition methods [2,67]. Representative methods are HE [35], LHS [36], and GIC [37].

The main goal of the third category is to extract illumination-invariant features. Most of the recently proposed methods belong to this category. Representative methods include SQI [49,50], single-light-region & single-dark-region SQI [51], LTV [52,53], LWT [54], generic intrinsic illumination subspace [55], local normalization [56], AdaS [57], LDCT [58], L&S Log-DCT [59], Gradientface [60], Weberface [61], GWF [62], LBP [42], LTP [16], LDP [43], EnLDP [44], LDN [45], DPPC [46], and local phase quantisation and multi-resolution local binary pattern fusion [68].

In this dissertation, we focus on face recognition under varying illumination. We present four different methods to produce illumination-invariant features. Our extensive experiments show each proposed method outperforms several state-of-the-art methods.

1.3 Contributions

This dissertation presents a few contributions which will be explained in detail in each chapter. The most significant contributions of this research are as follows:

1. Proposing methods which operate in the gradient domain to be robust against illumination variations.

2. Proposing logarithmic fractal dimensions (LFD) to transfer face images to the fractal dimension (FD) domain and produce illumination-invariant representations which enhance facial features such as eyes, eyebrows, nose, and mouth while keeping noise at a low level.

3. Proposing eight local directional patterns (ELDP) to extract more edge information by utilizing all eight directional edge images compared to LDP approaches.

4. Proposing adaptive homomorphic eight local directional patterns (AH-ELDP) to incorporate ELDP with an adaptive filter and an enhancement function to automatically attenuate the low-frequency (i.e., illuminance) component of each original face image and effectively enhance the contrast of the filtered image, respectively.

5. Incorporating LFD images with AH-ELDP to produce complete ELDP (CELDP) which considers the magnitude of edge images as well as all the eight directions.

As the contributions point out, we propose four evolving methods (i.e., LFD, ELDP, AH-ELDP, and CELDP) to produce illumination-insensitive representations. All these methods are individual systems and can be used to produce illumination-invariant images. However, AH-ELDP uses ELDP as the final stage to obtain the AH-ELDP image. The CELDP method used in our final system simplifies the LFD method by significantly reducing the running time without compromising the performance and combines the simplified LFD method with the adaptive filter and ELDP to obtain illumination-invariant images. The CELDP method considers not only all eight directional edge images but also the magnitude of these edge images.

1.4 Evaluation

We evaluate the proposed methods by conducting experiments on four publicly available image databases with large illumination variations. We also use a video face database to evaluate our methods. We implement all the stages of the practical face recognition system including face detection, landmark localization, face normalization, and face matching as shown in Fig. 1.2 to recognize faces in the video face database. In addition, we use receiver operating characteristic (ROC) curves to illustrate the ability of each method to verify and discriminate face images. In the following, we explain face databases and discuss ROC curves as they will be used in this research for evaluation purposes.

1.4.1 Face databases

We evaluate the proposed methods by conducting experiments on four publicly available image databases (i.e., Yale B, extended Yale B, CMU-PIE, and AR [32,69,70]) as well as the Honda UCSD video database [71]. These databases have different illumination conditions which help us to evaluate our methods in different illumination and light changes. We use the recognition accuracy, which is computed as the percentage of the number of correctly recognized faces to the number of all probe (testing) images, as the performance measure. In the following, we review these face databases.

Yale B face database: The Yale B database contains 5760 grayscale face images (480×640 pixels) of 10 subjects under nine poses and 64 illumination conditions (i.e., 576 images per subject). We use frontal face images in our experiment, so each subject has 64 images. All images are cropped and resized to 100×100 pixels. Fig. 1.5 shows eight sample images for a subject from this database. These images are categorized into six subsets based on the angle between the light source directions and the central camera axis: S0 (0°, 6 images per subject), S1 (1° to 12°, 8 images per subject), S2 (13° to 25°, 10 images per subject), S3 (26° to 50°, 12 images per subject), S4 (51° to 77°, 10 images per subject), and S5 (above 78°, 18 images per subject). S0 contains six images per subject, which are shown in the first column of Fig. 1.12 for one subject. These six images, respectively, have -35, -20, 0, 20, 45, and 90 degrees of elevation of the light source. Positive elevation implies the light source

is above the horizon, while negative elevation implies the light source is below the horizon. We conduct six experiments to evaluate the performance of the proposed methods. In the ith experiment, we use the ith image from each subject in S0 as the reference (training) image and the remaining five images from each subject in S0 (50 images in total) and all the images in each of the other five subsets as probe (testing) images. In each experiment, the remaining five probe images for each subject in S0 are formed a new subset S0' with 50 images. Then, for each experiment, we compute the recognition accuracy for each of the six subsets (i.e., S0', S1, S2, S3, S4, and S5). Next, the average face recognition accuracy of each subset across all the six experiments is computed and, considered as the recognition accuracy for the respective subset. We also report the standard deviation of the recognition accuracy across all the six experiments for each subset. Finally, we compute the average and standard deviation of the face recognition accuracy across all the six subsets.

The Weberface and Gradientface methods compute the face recognition accuracy of one training session, wherein the first image per subject from S0 is chosen for training. However, we conduct more thorough experiments by using each of six images per subject from S0 for training and computing the average face recognition accuracy using the remaining images outside of each of six training sessions.

Extended Yale B face database: The extended Yale B database contains grayscale face images (480×640 pixels) of 38 subjects under nine poses. We use frontal face images in our experiment. Each subject has 64 images (2414 images out of 2432 images are used since 18 images are corrupted). All images are cropped and resized to 100×100 pixels. Similar to the Yale B face database, these frontal face images are categorized into six subsets based on the angle between the light source directions and the central camera axis: S0 (0°, 228

Fig. 1.12**:** Face images in S0 of the Yale B face database.

images), S1 (1° to 12°, 301 images), S2 (13° to 25°, 380 images), S3 (26° to 50°, 449 images), S4 (51° to 77°, 380 images), and S5 (above 78°, 676 images). S0 contains six images per subject, which are similar to S0 in the Yale B database and have -35, -20, 0, 20, 45, and 90 degrees of elevation of the light source. We conduct six experiments using the same experimental settings for the Yale B database. Since the extended Yale B face database also contains the Yale B face database, images in Figs. 1.5 and 1.12 are also sample images from this database.

CMU-PIE face database: The CMU-PIE database, collected at Carnegie Mellon University in 2000, contains 41,368 grayscale images (486×640 pixels) from 68 subjects. They are captured under 13 different poses, 43 different illuminations, and 4 different expressions. Frontal images from the illumination subset (C27) are used in our experiment. This subset contains 21 images per subject. All images are cropped and resized to 100×100 pixels. Fig. 1.13 shows all 21 images for a subject from this database. We conduct 21 experiments to evaluate the performance of the proposed methods. In the ith experiment, we use the ith image from each subject as the reference (training) image and all the other 20 images as the probe (testing) images.

AR face database: The AR database, collected at Ohio State University, contains over 4,000 color images corresponding to 126 people's faces (70 men and 56 women). They are frontal view faces with different facial expressions, illumination conditions, and occlusions (sun glasses and scarf) [32]. Each person participated in two sessions, separated by a two-week (14 days) time period. The same pictures were taken in both sessions. We use the cropped version of the database used in [72]. This version contains 100 subjects (50 men and 50 women) of the size of 165×120 pixels. However, we still crop the faces to be similar to the other databases that we used. We only use images with illumination conditions and neutral expressions. They include two images with neutral expressions, two with light from the left, two with light from the right, and two images with light from both sides for each subject. Since the original size is small, all images are resized to 64×64 pixels. This also helps us to evaluate the robustness of parameters in our methods with different

Fig. 1.13**:** Illustration of 21 sample images for a subject from the CMU-PIE database.

sized images. Fig. 1.14 shows all eight images for a subject from the AR database. We conduct eight experiments to evaluate the performance of the proposed methods. In the ith experiment, we use the ith image from each subject as the reference (training) image and all the other seven images as the probe (testing) images.

Honda UCSD video database: The Honda UCSD video database is a standard video database for evaluating face tracking/recognition algorithms [71]. Each video sequence is recorded in an indoor environment at 15 frames per second, and each lasts for at least 15 seconds. The resolution of each video sequence is 640×480. Each individual is recorded in at least two video sequences. All the video sequences contain significant 2-D (in-plane) and 3-D (out-of-plane) head rotations. In each video, the person rotates and turns his/her head in his/her own preferred order and speed. Typically in about 15 seconds, the individual is able to provide a wide range of different poses. However, some of these sequences contain difficult events such as partial occlusion, face partly leaving the field of view, and large scale changes. The database contains two datasets corresponding to the two video sequences. We use training and testing subsets from the first dataset, which respectively contain 20 videos for training and 39 videos for testing. The training videos belong to 20 human subjects, while the testing videos contain 19 individuals of these subjects. We exclude the subject that is not in the testing set in our experiment.

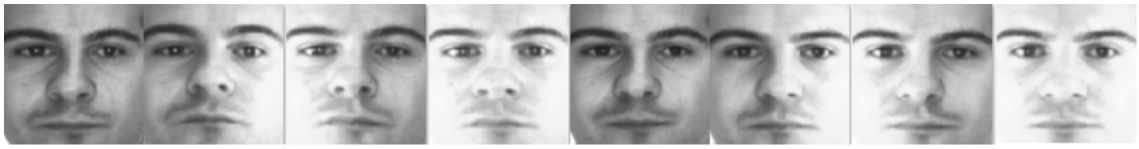

Fig. 1.14**:** Eight images with light changes and neutral expression from a subject in the AR face database.

1.4.2 Receiver Operating Characteristic (ROC) Curve

The ROC curve is a graphical plot to illustrate the verification and discrimination ability of a binary classifier. It plots the true positive rate (TPR) against the false positive rate (FPR) of the classifier. A classifier or method with higher TPR and lower FPR indicates a better verification and discrimination ability. Since we use the same classifier for all methods, the ROC curve can illustrate the ability of each method to verify and discriminate face images. Here, a positive match indicates an input image for a subject is determined to be the same subject as the reference (training) image.

1.5 Outline

The dissertation is organized as follows based on the main contributions of the research:

Chapter 2 proposes a novel fractal analysis (FA) feature-based preprocessing method, called Logarithmic Fractal Dimension (LFD), to generate an illumination-invariant representation for a given face image.

Chapter 3 proposes a modified version of Local Directional Patterns (LDP), called Eight Local Directional Patterns (ELDP), to produce an illumination-insensitive representation from an input face image.

Chapter 4 proposes a three-step illumination-invariant face recognition method. This method derives a descriptor called Adaptive Homomorphic Eight Local Directional Patterns (AH-ELDP) to produce an illumination-insensitive representation.

Chapter 5 proposes a complete eight local directional patterns (CELDP) method, which effectively incorporates the proposed LFD and AH-ELDP methods to produce robust illumination-insensitive representations.

Chapter 6 evaluates the aforementioned proposed methods by conducting experiments on four publicly available image face databases with large illumination variations as well as a video face database. It, also, compares them with several recently proposed state-of-the-art methods to illustrate the effectiveness of the proposed methods.

Chapter 7 draws the conclusion of the dissertation.

Chapter 2

Face Recognition under Varying Illuminations with Logarithmic Fractal Analysis

2.1 Introduction

Fractal analysis (FA), as a type of texture analysis, has been recently used in medical imaging and image processing [73–76]. For texture analysis of fractal features, image intensities are transformed to the fractal dimension (FD) domain [74]. The FD transform is considered as an edge enhancement and preprocessing algorithm that does not increase noise [73, 74]. Specifically, Al-Kadi and Watson [74] enhance edges in respective images using FA to differentiate between aggressive and nonaggressive malignant lung tumors. Kim et al. [75] apply FA to detect and predict glaucomatous progression. Zlatintsi and Maragos [76] use multiscale FA to quantify the multiscale complexity and fragmentation of different states of the music waveform.

In this chapter, we propose to apply a novel FA feature-based preprocessing method to generate an illumination-invariant representation for a given face image. To the best of our knowledge, this is the first attempt to apply FA in the face recognition task to achieve illumination-insensitive representation. To this end, we first perform a log-based transformation to partially reduce the illumination effect and make the image brighter. This log function expands the values of dark pixels and compresses brighter pixels in the image. As a result, pixel values are spread more uniformly. We then transfer the scaled image to a Logarithmic FD (LFD) image using the Differential Box-Counting (DBC) algorithm [74, 77, 78]. The proposed method makes the following contributions:

1. Using a simple and efficient log function to expand dark pixels and compress bright pixels for partial illumination reduction.

2. Transforming face images to the FD domain to produce the illumination-invariant representation.

3. Enhancing facial features such as eyes, eyebrows, nose, and mouth while keeping noise at a low level.

2.2 Methodology

2.2.1 FA Feature-Based Method

The FD transform is an effective edge enhancer technique that keeps the noise level low [73,74]. Since edge magnitudes are largely insensitive to illumination variations [45], we apply FA to produce corresponding FD images of given face images to achieve illumination-insensitive representation.

Fractals are defined as a geometrical set whose Hausdorff-Besicovitch dimension strictly exceeds the topological dimension. They describe non-Euclidean structures that show self-similarity at different scales [74]. Most of biological and natural features tend to have a FD [74]. We use the DBC algorithm, a popular computationally efficient method for dealing with large images, to quickly transfer face images to FD images [74, 77–79].

In order to transfer the face image Im of size $M \times N$ to the image FD in the FD domain, we first compute a sequence of two-dimensional matrices Mat's to overlay the image Im at each pixel (x, y) as follows:

$$Mat_{d-1}(x,y) = (\text{floor}[\frac{max(W_d(x,y)) - min(W_d(x,y))}{d}] + 1)(\frac{d_{max}}{d})^2 \tag{2.1}$$

where d is the scaling factor with a minimum value of 2 and a maximum value of d_{max} that represents how much a specific structure of pixels is self-similar to its surroundings, $W_d(x, y)$ is a $d \times d$ window or neighborhood block whose center is positioned at the location (x, y) of an image. When d is an odd number, the center is in the middle of the neighborhood block. When d is an even number, the center is to the left and above the middle of the neighborhood block. Fig. 2.1 shows the center position for the $d \times d$ neighborhood block

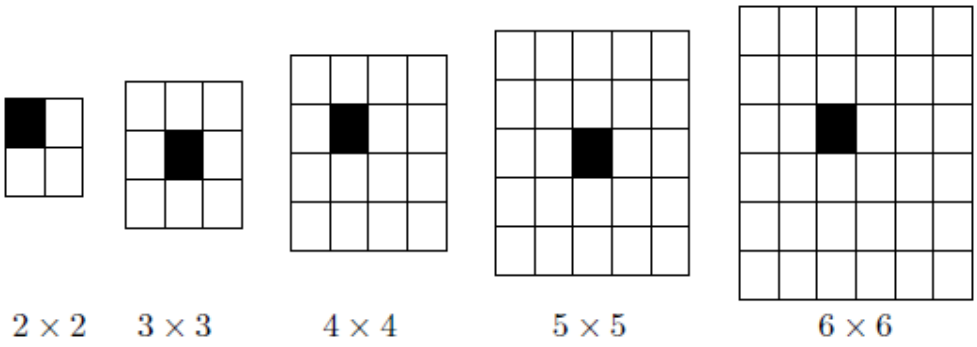

Fig. 2.1**:** Illustration of the center position (shown in black) for the $d \times d$ neighborhood block when $d = 2, 3, 4, 5, 6$.

when $d = 2, 3, 4, 5, 6$. *max* and *min* operations obtain the highest and lowest intensity values in the image block covered by the $d \times d$ neighborhood block centering at the location (x, y) of an image, respectively. Specifically, for each scaling factor d, the center of a sliding neighborhood block of size $d \times d$ is positioned at each pixel in an image. The maximum intensity value and the minimum intensity value of the image portion covered by the sliding neighborhood block are used to compute the new value at the corresponding pixel location following Eq. (2.1).

We then convert each two-dimensional matrix Mat_{d-1} of size $M \times N$ to a one-dimensional column vector of size $(M \times N) \times 1$ and concatenate the converted $(d_{max} - 1)$ column vectors to obtain the two-dimensional matrix V, which has $(M \times N)$ rows and $(d_{max} - 1)$ columns. For each row in the matrix V, it indicates the new values computed by Eq. (2.1) at a particular location in an image across scales from 2 to d_{max}. For example, the first row in V contains the new values at location $(1, 1)$ of an image across scales from 2 to d_{max}. The last row in V contains the new values at location (M, N) of an image across scales from 2 to d_{max}. Here, we use the one-dimensional vector $v_{(x,y)}$ to denote the new values at location (x, y) of an image across scales from 2 to d_{max} with $v_{(x,y),d}$ indicating the new value at location (x, y) for the scaling factor of d.

Finally, for each pixel (x, y), we generate the fractal slope by the linear regression line of the one-dimensional vector $v_{(x,y)}$ and the scaling factor d $(d = 2, ..., d_{max})$ to represent the FD value at (x, y) (i.e., $FD(x, y)$). To this end, we first apply the log function on all

the elements of V and the respective scaling factor d to compress the dynamic range of the images [74]. The FD value at (x, y) is then computed as the fractal slope of the least square linear regression line by:

$$FD(x, y) = \frac{S_v(x, y)}{S} \tag{2.2}$$

where S and S_v are the sums of squares as follows:

$$S = \sum_{d=1}^{d_{max}-1} (log(d))^2 - \frac{(\sum_{d=1}^{d_{max}-1} log(d))^2}{d_{max} - 1} \tag{2.3}$$

$$S_v(x, y) = \sum_{d=1}^{d_{max}-1} log(d)(log(v_{(x,y)),d})) - \frac{(\sum_{d=1}^{d_{max}-1} log(d))(\sum_{d=1}^{d_{max}-1} log(v_{(x,y),d}))}{d_{max} - 1} \tag{2.4}$$

The transformation process to convert face images to FD images is illustrated in steps 2 and 3 of Fig. 2.2.

2.2.2 Implementation

This subsection illustrates how to implement the proposed method. First, we perform a log-based transformation to partially reduce the illumination effect and make the image brighter. This log function expands the values of dark pixels and compresses the values of bright pixels. Next, we transfer the scaled image to the FD domain using the FA feature-based method introduced in the previous subsection. The final image is called LFD image. The entire process to compute the LFD image is illustrated in Fig. 2.2 and the algorithmic view of the proposed method is summarized in Algorithm 2.1.

Fig. 2.3 shows two face images along with their corresponding Log and LFD images. It clearly demonstrates that the log function reduces the illuminance to some extent. Furthermore, LFD images enhance the important features of faces such as eyes, eyebrows, nose, mouth, and the shape of the face in general. Comparing LFD images with their original images verifies that the proposed method produces illumination-insensitive features. Fig.

2.4(a) shows 21 images for a subject from the CMU-PIE database and Fig. 2.4(b) shows their corresponding LFD images.

Algorithm 2.1 The algorithmic view of the proposed LFD method.

Input: Original face image $OriIm$.

Output: The fractal dimension image LFD.

1. Perform the log transformation on the original image $OriIm$
$$Im = log(OriIm)$$
2. For $d = 2, 3, ..., d_{max}$
   a) Update the size for the neighborhood block W to $d \times d$.
   b) Compute the 2D Mat_{d-1} using Eq. (2.1).
3. Convert all $d_{max} - 1$ two-dimensional matrices Mat's of size $M \times N$ to the two-dimensional matrix V of size $(M \times N) \times (d_{max} - 1)$.
4. Initialize the fractal dimension image LFD with the same size of the original face image $OriIm$, as all 0's.
5. Compute S using Eq. (2.3).
6. For each pixel at location (x, y) of the LFD image
   a) Compute S_v using Eq. (2.4).
   b) Set the value of $LFD(x, y)$ using Eq. (2.2).

2.3 Parameter d_{max}

The proposed FA feature-based method has only one parameter d_{max}. It is the maximum value for the scaling factor which represents how much a specific structure of pixels is self-similar to its surroundings. Different values for d_{max} can lead to different accuracy. Fig. 2.5 lists the recognition accuracy for CMU-PIE and Yale B databases using values ranging from 2 to 20 for d_{max}. It clearly shows both databases have similar trends towards d_{max}. Specifically, accuracy for both databases increases gradually until d_{max} value is almost 9 and then accuracy does not significantly change for d_{max} values more than 9. This observation indicates different face databases would have similar behavior regarding d_{max} since CMU-PIE and Yale B databases contain different illumination conditions. Therefore, we recommend to use a d_{max} value between 9 and 12 for the face recognition task to achieve a decent compromise between computational time and good face recognition accuracy. We

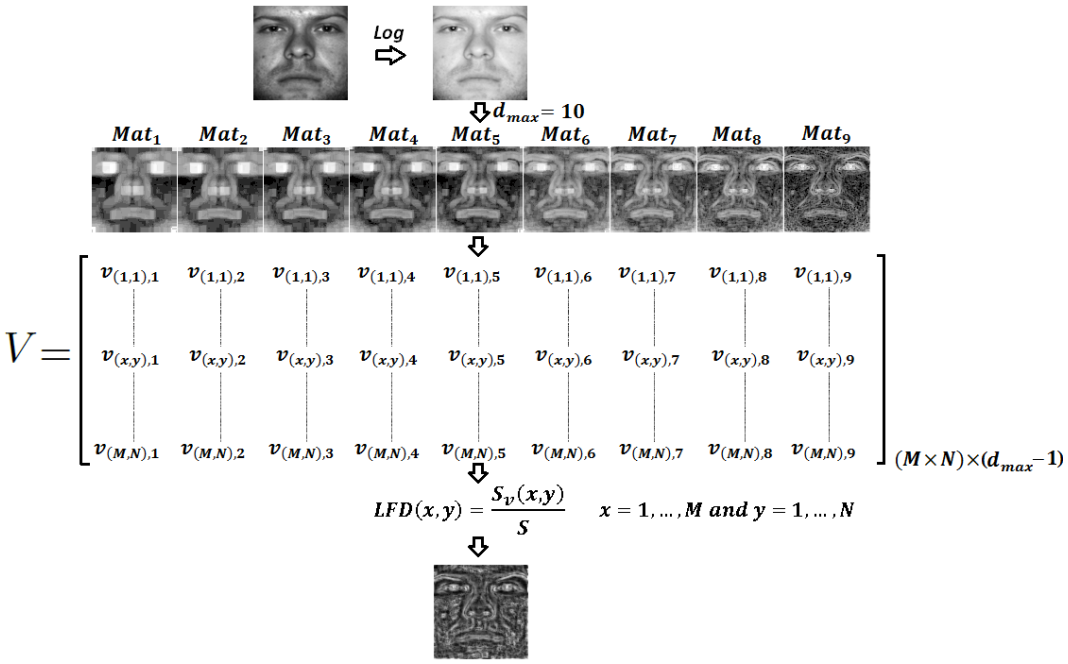

Fig. 2.2**:** Illustration of the proposed LFD process: 1) The face image is scaled by the log function. 2) For each d, the matrix Mat_{d-1} is computed using the DBC algorithm. 3) Mat's are converted to V and the LFD image is obtained using Eq. 2.2.

set d_{max} to be 10 for both face databases in our experiment.

2.4 Summary

We propose a FA feature-based method to produce illumination-invariant features from face images with illumination variations. Our contributions are:

- Applying an effective and simple log transformation to produce partially illumination reduced face images.
- Applying the FA feature-based method to produce the illumination-invariant face representations (i.e., LFD images) while enhancing facial features such as eyes, eyebrows, nose, and mouth and keeping noise at a low level.

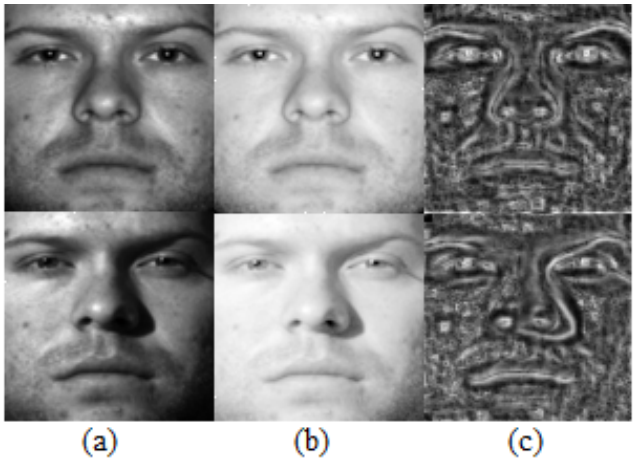

Fig. 2.3**:** Results of the proposed FA feature-based method. (a) original face images; (b) scaled images after the log operation; and (c) LFD images.

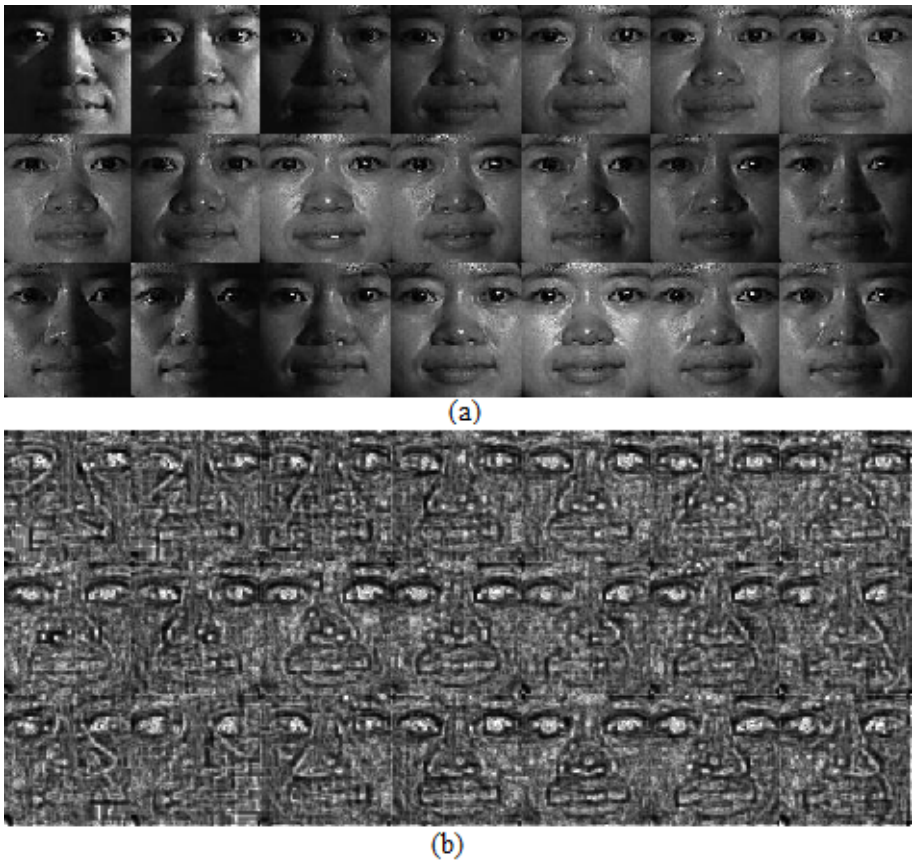

Fig. 2.4**:** Illustration of sample images and their LFD images. (a) 21 samples from the CMU-PIE database(b) Corresponding LFD images.

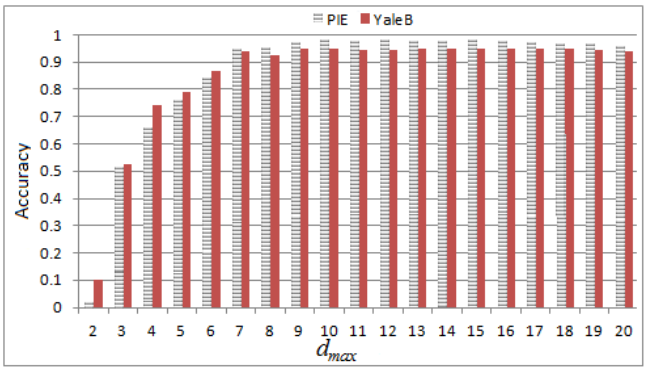

Fig. 2.5**:** Comparison of the recognition accuracy of the proposed LFD method with different d_{max} values ranging from 2 to 20 for CMU-PIE and Yale B images.

Chapter 3

Face Recognition Under Illumination Variations Based on Eight Local Directional Patterns

3.1 Introduction

The LBP, LTP, LDP, EnLDP, and LDN methods produce illumination-invariant representations. The LBP and LTP methods summarize local gray-level structures. They take a local neighborhood of P pixels in a circle of radius R around each pixel and subtract the value of the central pixel from them to obtain eight values. Usually, for face recognition applications, P and R are set to be 8 and 2, respectively. LBP converts the resulting eight values to the values of 1's if they are positive and 0's if they are negative and then transfers the 8-bit binary pattern to the corresponding decimal value. LTP uses a user-specified positive threshold to convert the resulting eight values to 1's, -1's, or 0's. If they are greater than the threshold, convert them to 1's; if they are less than the negative of the threshold, convert them to -1's; otherwise, convert them to 0's. Finally, it splits the ternary pattern into two binary patterns. The final LBP and LTP images are illumination-invariant.

LDP, EnLDP, and LDN operate in the gradient domain to produce illumination-invariant representations. They could be also considered as LBP approaches. However, they use edge directional information instead of intensity changes, since edge responses are insensitive to lighting variations [45]. Specifically, they use Kirsch compass masks (Fig. 1.10) to compute the edge responses and produce eight directional edge images (or eight directional numbers) for a face image by the convolution of the original face image $OriIm(x, y)$ and each of eight masks M_i $(0 \leq i \leq 7)$. LDP considers an 8-bit binary code for each pixel and assigns 1's to the three bits corresponding to the three prominent numbers and 0's to the other five bits. EnLDP and LDN consider a 6-bit binary code for each pixel. In EnLDP,

the first three bits code the position of the top positive directional number and the next three bits code the position of the second top positive directional number. In LDN, the first three bits code the position of the top positive directional number and the next three bits code the position of the top negative directional number. The generated codes are then converted to their corresponding decimal values to respectively produce LDP, EnLDP, and LDN images, which are illumination-invariant.

This chapter proposes an illumination-invariant face recognition method, called Eight Local Directional Patterns (ELDP). This new ELDP method uses all the eight directional numbers instead of the three prominent directional numbers as used in LDP or the top two positive directional numbers as used in EnLDP or the top positive and negative directional numbers as used in LDN to encode more local structural information and intensity changes of a face image. The proposed ELDP makes the following contributions:

1. Operating in the gradient domain to be robust against illumination variations and noise.

2. Extracting more edge information by utilizing all eight directional edge images.

3. Considering relations among eight directional numbers to use valuable structural information.

3.2 Methodology

To produce the illumination-invariant representation, we first use Kirsch compass masks to compute the edge responses (i.e., eight directional edge images). Each of these eight edge images represents the edge significance in its respective direction [45]. These edge responses have different importance levels, but all of them are significant. As a result, we use all the eight directional numbers of each pixel to assign an associated 8-bit binary code. If the directional number of its edge image is positive, we set the value of the respective bit to 1. Otherwise, we set the value of the respective bit to 0. It should be noted that the ordering of the directional numbers is not important as long as the same order is employed for all images. Here, we use the conventional order used in LBP approaches in our system.

Finally, we compute the corresponding decimal value for the binary code and consider it as the pixel's ELDP value. Fig. 3.1 illustrates the detailed steps to compute the ELDP value at one location. The same procedure can be applied to all locations in a face image to produce the final ELDP image as shown in Fig. 3.1(d).

ELDP is similar to LBP in one perspective since both transform the 8-bit binary code to a decimal value. However, they use different mechanisms to generate the 8-bit code. Specifically, ELDP uses the positive or negative edge response information of the eight directional edge images to generate the 8-bit code. LBP uses the sparse points compared with the center pixel in each neighborhood of the original image as the threshold (i.e., few number of intensities in a neighborhood) to generate the 8-bit code. As a result, LBP discards most of the information in the neighborhood, makes the method sensitive to noise, and limits the accuracy of the method [45]. ELDP avoids the above shortcomings of LBP by using more edge information from the entire neighborhood and is therefore superior to LBP.

The proposed ELDP is also similar to LDP, EnLDP and LDN methods [43,45] from the perspective of the use of Kirsch compass masks to compute the edge responses. However, LDP, EnLDP, and LDN respectively use the three prominent directional numbers, the first and second top positive directional numbers, and the top positive and negative directional numbers to generate their corresponding binary codes. ELDP uses all eight directional numbers to generate its 8-bit binary code. Since these eight directional numbers provide the "panorama view" of gradient directions in a chosen neighborhood, ELDP offers the following advantages:

- It is robust against illumination variations and noise by operating in the gradient domain [45].
- It extracts more edge information by utilizing all eight directional edge images.
- It indicates valuable structural information from the neighborhood by considering the relations among eight directional edge responses.

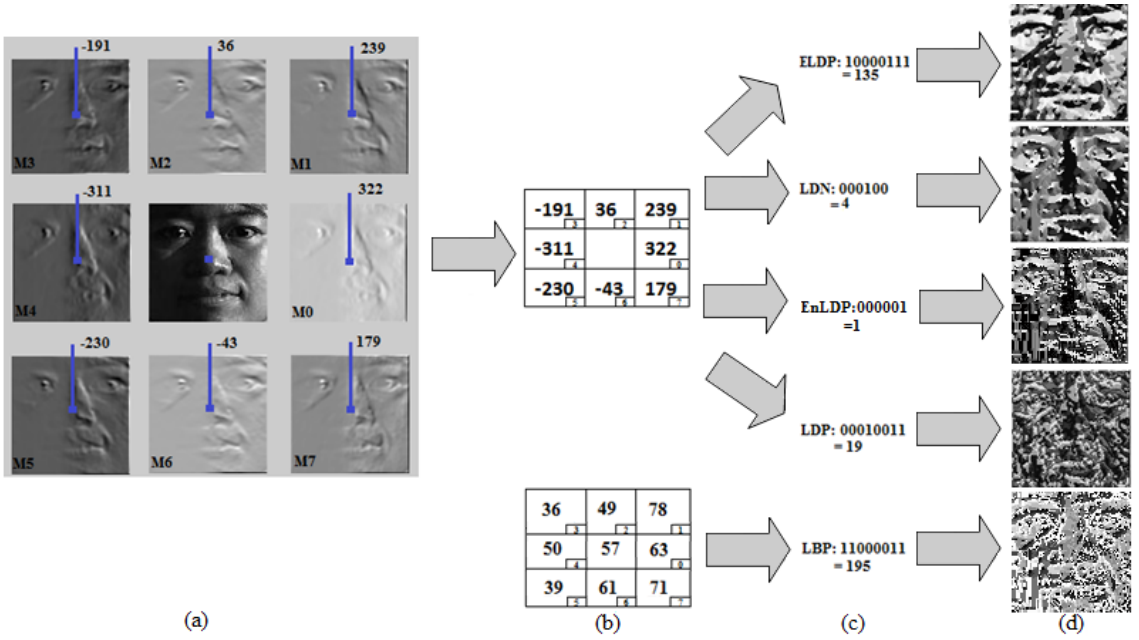

Fig. 3.1**:** Illustration of the proposed ELDP code scheme. (a) Edge response images produced by convolving the original face image and Kirsch compass masks together with their response values at their specific same locations marked in blue dots. (b) Top: The ELDP, LDN, EnLDP, and LDP window generated by the directional numbers at the same specified location of each response image. Bottom: The LBP window at the same specified location of the original image (the center pixel with an intensity value of 57). (c) The ELDP, LDN, EnLDP, LDP, and LBP codes at the same specified location. (d) The final ELDP, LDN, EnLDP, LDP, and LBP images generated by applying the same procedure to all locations of the original image.

- It is simple and easy to compute.

3.3 Implementation

Similar to other work, we first smooth the face image with a Gaussian filter to reduce the side-effect of shadow boundaries and make the gradient computation more robust to noise [60, 61]. We then implement the ELDP code scheme. The algorithmic view of the proposed method is summarized in Algorithm 3.1.

Fig. 3.2 shows sample original images from the CMU-PIE face database and their corresponding ELDP images. It clearly shows the ELDP code scheme reduces the effect of illumination and produces illumination-insensitive images since all ELDP images look alike as shown in Fig. 3.2(b).

Algorithm 3.1 The algorithmic view of the proposed ELDP.

Input: Original face image $OriIm$.

Output: The ELDP image

1. Obtain the smoothed image Im by computing the convolution of the original Image $OriIm$ and the Gaussian filter as

$$Im = OriIm * G(x, y, \sigma)$$

where * is the convolution operator and

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} exp(-\frac{x^2+y^2}{2\sigma^2})$$

is the Guassian kernel function with standard deviation σ.

2. Compute eight edge directional images by convolving the smoothed image Im with the Kirsch compass masks

$$\mathbb{I}_i = Im * M_i; i = 0, 1, ..., 7$$

where M_i is the ith Kirsch compass mask as shown in Fig. 1.10.

3. Initialize the output image ELDP with the same size of the original face image $OriIm$, as all 0's.

4. For each pixel at location (x, y) of the ELDP image

   a) Assign an 8-bit binary code B by

$$B_i = \begin{cases} 1 & \mathbb{I}_i(x, y) > 0 \\ 0 & otherwise \end{cases}$$

   b) Convert the binary code B to its respective decimal value.

   c) Assign the decimal value to ELDP(x, y).

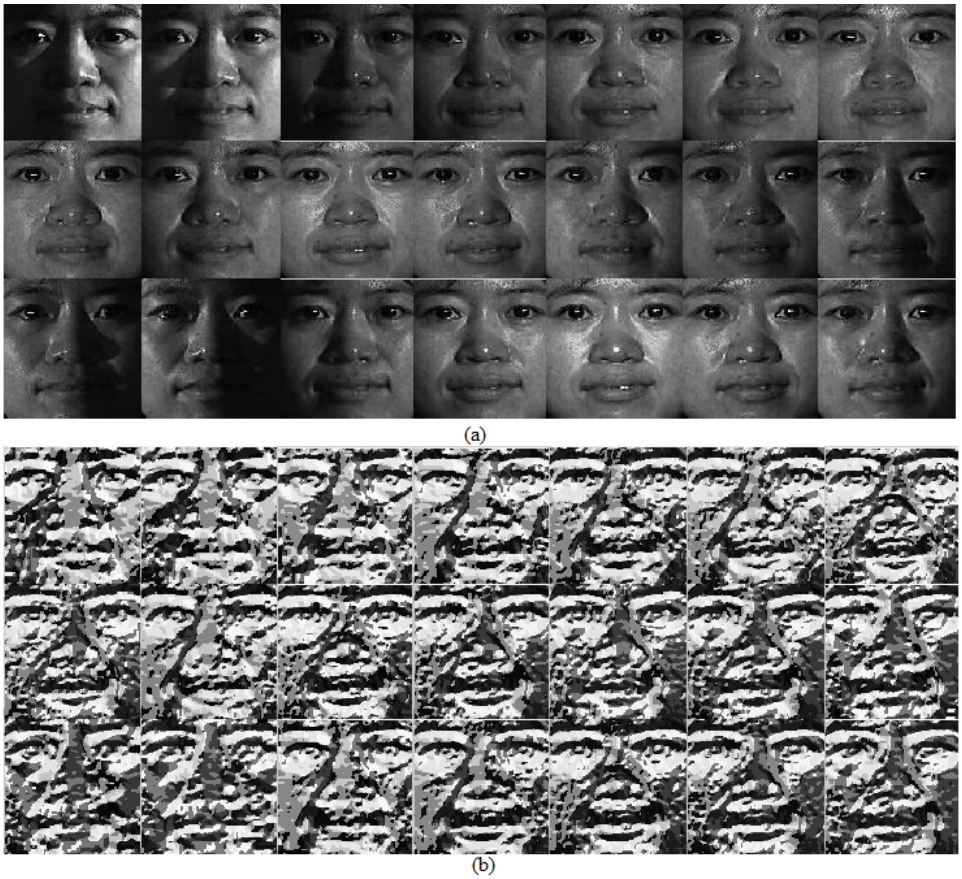

Fig. 3.2**:** Illustration of sample images and their ELDP images. (a) 21 samples from the CMU-PIE database (b) Corresponding ELDP images.

3.4 Summary

We propose a new illumination-invariant face descriptor, ELDP, to preprocess face images with illumination variations. The proposed ELDP method uses Kirsch compass masks to compute eight directional edge responses in the gradient domain, which is robust against illumination changes and noise. It extracts more edge information by utilizing all eight directional edge images. It also indicates valuable structural information from the neighborhood by considering the relations among eight directional edge responses.

Chapter 4

Face Recognition under Varying Illuminations Based on Adaptive Homomorphic Eight Local Directional Patterns

4.1 Introduction

This chapter proposes a three-step illumination-invariant face recognition method. The new method derives a descriptor called Adaptive Homomorphic Eight Local Directional Patterns (AH-ELDP) to produce an illumination-insensitive representation. As a result, we name it as the AH-ELDP method. First, the proposed method employs an adaptive homomorphic filter to reduce the influence of illumination. Then, it uses the interpolation function to enhance the image features. Finally, it applies all the eight directional edge numbers to produce illumination-invariant features to recognize faces.

The contributions of the proposed AH-ELDP method are:

1. Adjusting the homomorphic filter parameter c to automatically attenuate the low-frequency (i.e., illuminance) component of each original face image.

2. Employing an interpolation function to effectively enhance the contrast of the filtered image.

3. Presenting a new gradient-based descriptor by considering the relations among all eight directional edge responses to achieve robustness against illumination variations and noise [45] and represent more valuable structural information from the neighborhood.

4.2 Methodology

The proposed AH-ELDP method consists of three steps. The first step is to use homomorphic filtering to reduce the illumination effects. The second step is to apply an interpolative enhancement function to stretch the contrast of the filtered image. The last step employs the ELDP method, explained in Chapter 3, to produce the illumination-invariant representation of a face image.

4.2.1 Homomorphic Filtering

Based on the Lambertian-reflectance model, a face image I is represented as follows:

$$I(x, y) = R(x, y)L(x, y) \tag{4.1}$$

where $I(x, y)$, $R(x, y)$, and $L(x, y)$ are respectively the pixel intensity, reflectance, and illuminance at position (x, y) of a face image [61, 65, 80]. Specifically, R is considered as a high-frequency signal closely corresponding to texture information of a face because reflectance indicates the contrast arrangement of skin, eyebrows, eyes, and lips of a face. L is considered as a low-frequency signal because the illumination values of neighboring pixels in a face image are similar to each other [65].

In order to reduce the influence of illumination, we use the homomorphic filter $H(u, v)$ as a Gaussian highpass filter [80]. This filter adjusts the image intensity by strengthening the high-frequency signal and attenuating the low-frequency signal [65]. However, the Fourier transform of the two functions, $R(x, y)$ and $L(x, y)$, in Eq. (4.1) is not separable. So, we use the logarithmic operation to separate these functions as follows:

$$ln(I(x, y)) = ln(R(x, y)) + ln(L(x, y)) \tag{4.2}$$

We then apply the Fourier transform F on the logarithmic results by:

$$\begin{aligned} F(ln(I(x,y))) &= F(ln(R(x,y))) + F(ln(L(x,y))) \\ \Leftrightarrow F_I(u,v) &= F_R(u,v) + F_L(u,v) \end{aligned} \tag{4.3}$$

where $F_I(u,v)$, $F_R(u,v)$, and $F_L(u,v)$ are the Fourier transform of $ln(I(x,y))$, $ln(R(x,y))$, and $ln(L(x,y))$, respectively. Now, we multiply the homomorphic filter, $H(u,v)$, on both sides of Eq. (4.3) to yield the following:

$$\begin{aligned} S(u,v) &= H(u,v)F_I(u,v) \\ &= H(u,v)F_R(u,v) + H(u,v)F_L(u,v) \end{aligned} \tag{4.4}$$

where S is the filtered result in the frequency domain. Subsequently, we let F^{-1} denote the inverse Fourier transform and compute $I_{F^{-1}}(x,y)$ as the inverse Fourier transform of $S(u,v)$ as follows:

$$\begin{aligned} I_{F^{-1}}(x,y) &= F^{-1}(H(u,v)F_R(u,v)) + F^{-1}(H(u,v)F_L(u,v)) \\ &= R_{F^{-1}}(x,y) + L_{F^{-1}}(x,y) \end{aligned} \tag{4.5}$$

where $R_{F^{-1}}(x,y) = F^{-1}(H(u,v)F_R(u,v))$, and $L_{F^{-1}}(x,y) = F^{-1}(H(u,v)F_L(u,v))$.

Finally, we use the exponential operation to yield the desired filtered image $I_f(x,y)$ as follows:

$$\begin{aligned} I_f(x,y) &= exp^{I_{F^{-1}}(x,y)} = exp^{R_{F^{-1}}(x,y)} exp^{L_{F^{-1}}(x,y)} \\ &= R_f(x,y)L_f(x,y) \end{aligned} \tag{4.6}$$

where $R_f(x,y) = exp^{R_{F^{-1}}(x,y)}$ and $L_f(x,y) = exp^{L_{F^{-1}}(x,y)}$ are illuminance and reflectance of the filtered image at position (x,y), respectively.

Similar to [65], we use the modified Gaussian highpass filter discussed in [80] to define $H(u, v)$ by:

$$H(u, v) = (\gamma_H - \gamma_L)[1 - exp^{-cD^2(u,v)/D_0^2}] + \gamma_L \tag{4.7}$$

where $D(u, v)$ is the distance from (u, v) to the origin of the centered Fourier transform, D_0 is the cutoff distance measured from the origin, $\gamma_L < 1$ and $\gamma_H > 1$ are the parameters of the filter, and c is a constant to control the sharpness of the slope of the filter function as it transitions between γ_H and γ_L [80]. The homomorphic filtering using the aforementioned process is summarized in Fig. 4.1. We finally normalize the filtered image I_f so that the normalized filtered image f falls in the range of [0, 1]. Fig. 4.2(a) and (b) show two sample images and their corresponding homomorphic filtered images, respectively. Fig. 4.2(b) clearly indicates that the homomorphic filtering attenuates the low-frequency component that is the illuminance of an image.

4.2.2 Image Enhancement

After reducing the illumination influence, we enhance features of the filtered face image. To this end, we adopt the idea of the interval type-2 fuzzy sets [81] to enhance the contrast of the filtered image by making the darker area in the image brighter and the brighter area in the image darker. We then set the lower and upper values for each pixel that is currently in the range of [0, 1] as follows:

$$\mu_{Lower}(x, y) = f^2(x, y) \tag{4.8}$$

and

$$\mu_{Upper}(x, y) = f^{0.5}(x, y) \tag{4.9}$$

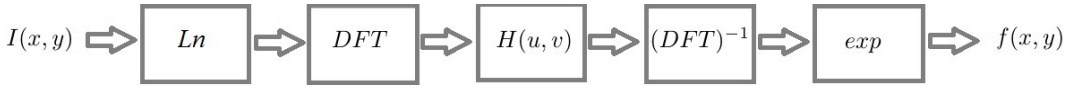

Fig. 4.1: Homomorphic filtering approach to partially reduce the illumination of a face image.

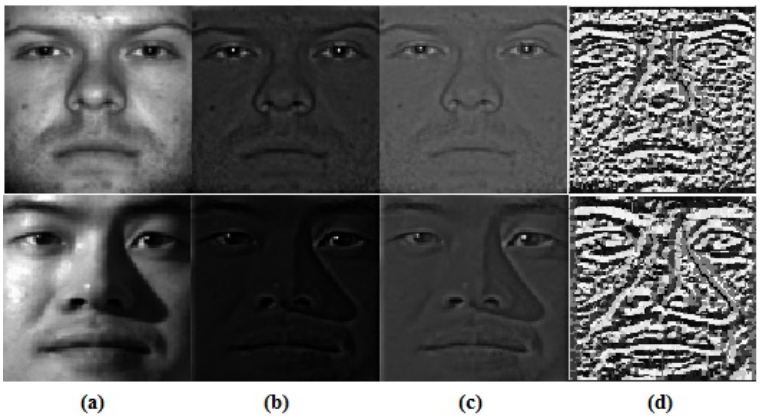

Fig. 4.2: Illustration of the intermediate results obtained by the three steps of the proposed AH-ELDP method. (a) Original face images; (b) Homomorphic filtered images; (c) Enhanced filtered images; and (d) AH-ELDP images after extracting eight local directional patterns.

where $\mu_{Lower}(x,y)$ and $\mu_{Upper}(x,y)$ are the lower and upper values of the normalized filtered pixel value at position (x,y), respectively. The enhancement function is then defined by:

$$\begin{aligned} g(x,y) &= (\mu_{Lower}(x,y) \times f_{mean}(x,y)) \\ &+(\mu_{Upper}(x,y) \times (1 - f_{mean}(x,y))) \end{aligned} \tag{4.10}$$

where $f_{mean}(x,y)$ ($0 \leq f_{mean}(x,y) \leq 1$) and $g(x,y)$ are respectively the local mean and the enhanced value at position (x,y). From the mathematical perspective, Eq. (4.10) is actually the linear interpolation between the two end points (i.e., the lower and upper values). $g(x,y)$ is close to the lower value when $f_{mean}(x,y)$ is close to 1 (i.e., when the pixel is in a bright neighborhood). Similarly, $g(x,y)$ is close to the upper value when $f_{mean}(x,y)$ is close to 0 (i.e., when the pixel is in a dark neighborhood). Therefore, this interpolation function makes pixels in a dark neighborhood brighter and pixels in a brighter neighborhood darker. In other words, it enhances the details of the filtered image in dark and bright neighborhoods. Fig. 4.2(c) shows the results of this interpolative enhancement operation on two homomorphic filtered images.

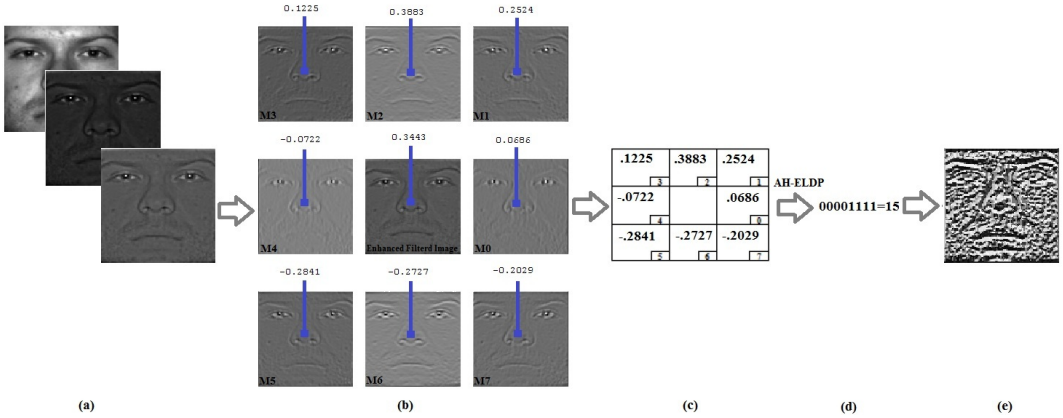

Fig. 4.3**:** Illustration of the proposed AH-ELDP method. (a) An original face image along with its corresponding homomorphic filtered and enhanced filtered images. (b) Edge response images produced by computing the convolution of the enhanced filtered image and Kirsch compass masks together with the directional numbers at their specific same positions marked in blue dots. (c) The 3×3 AH-ELDP window generated by the directional number at the same specified position of each response image. (d) The AH-ELDP code at the same specified position. (e) The final AH-ELDP image generated by applying the same procedure to all positions of the original image.

4.2.3 Eight Local Directional Patterns

This subsection presents a modified version of the LDP method to produce illumination-invariant features since enhanced filtered images still contain part of the illumination information. As we explained earlier, LDP-based methods (i.e, LDP, EnLDP, and LDN) operate in the gradient domain to produce illumination-invariant features. LDP, EnLDP, and LDN use the three prominent directional numbers, the two top positive directional numbers, and the top positive and negative directional numbers of the eight directional numbers at each position to generate illumination-invariant representations, respectively.

However, each edge directional image represents the edge significance in its respective direction and each directional number provides the gradient direction in a chosen neighborhood. Therefore, all the edge responses are significant. The proposed AH-ELDP method uses all the eight directional numbers, instead of only few of them, to provide more valuable structural information of the neighborhood. We assign an 8-bit binary code to each pixel. If the directional number of its edge directional image is positive, we set the value of the respective bit to 1's. Otherwise, we set the value of the respective bit to 0's. Finally, the binary code is converted to its corresponding decimal value which is considered as the pixel's AH-ELDP value. Fig. 4.3 illustrates the detailed steps to compute the AH-ELDP

value at one position. The same procedure can be applied to all positions to obtain the final AH-ELDP image as shown in Fig. 4.3(e). The algorithmic view of the proposed AH-ELDP method is summarized in Algorithm 4.1. Fig. 4.4 shows sample original images from the CMU-PIE face database and their corresponding AH-ELDP images. It clearly shows the AH-ELDP method reduces the effect of illumination and produces illumination-insensitive images since all AH-ELDP images look alike as shown in Fig. 4.4(b).

Algorithm 4.1 The algorithmic view of the proposed AH-ELDP.

Input: Original face image $OriIm$.

Output: The AH-ELDP image

1. Compute the value of the parameter c using Eq. (4.11).
2. Apply the homomorphic filtering approach shown in Fig. 4.1 on $OriIm$ to obtain the normalized filtered image f.
3. Apply Eq. (4.10) on f to obtain the enhanced filtered image g.
4. Compute eight edge directional images by convolving g with the Kirsch compass masks
$$\mathbb{I}_i = g * M_i; i = 0, 1, ..., 7$$
where M_i is the ith Kirsch compass mask as shown in Fig. 1.10.
5. Initialize the output image AH-ELDP with the same size of the original face image $OriIm$, as all 0's.
6. For each pixel at location (x, y) of the AH-ELDP image
   a) Assign an 8-bit binary code B by
$$B_i = \begin{cases} 1 & \mathbb{I}_i(x, y) > 0 \\ 0 & otherwise \end{cases}$$
   b) Convert the binary code B to its respective decimal value.
   c) Assign the decimal value to AH-ELDP(x, y).

The three steps are combined to produce illumination-insensitive representations for face images under varying illuminations. The adaptive homomorphic (AH) filter is a high-pass filter which is used to attenuate the low frequency signal (i.e., illumination component) and strengthening the high frequency signal (i.e., texture information). This filter has been

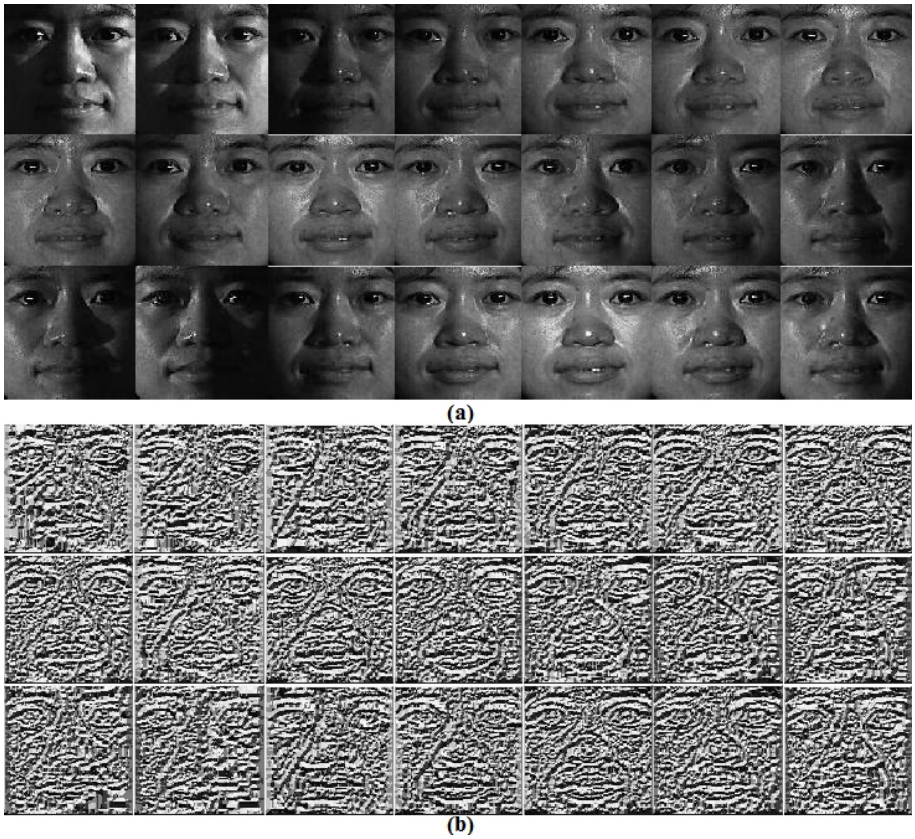

Fig. 4.4**:** Illustration of sample images of the CMU-PIE database and their AH-ELDP images. (a) 21 samples for a subject (b) Corresponding AH-ELDP images.

explained in [80] in more detail and used in [65] to reduce illumination in face images. We designed a method to compute the adaptive c value to automatically adjust the homomorphic filter for each face image based on its illumination (low frequency). So the illumination of each face image is reduced based on its own frequency.

The interpolative enhancement function is used to enhance the contrast of the filtered image. It simply uses the linear interpolation to adjust the intensity values by making darker pixels in the image brighter and brighter pixels in the image darker. The degree of contrast enhancement is varied depending on the brightness change in a region. As a result, the facial features of each face image are enhanced.

Finally, ELDP is a modified version of the LDP approach to produce the edge responses which are insensitive to lighting variations [43–45]. ELDP uses all the eight directional numbers, instead of few of them as used in LDP, to provide more valuable structural information

of the neighborhood.

Each of these three steps either enhances face images or reduces and removes illumination. Therefore, the proposed AH-ELDP method deems to be an illumination-invariant method for the face recognition.

4.3 Settings of Parameters

The AH-ELDP method has four parameters which all belong to the homomorphic filter: D_0, γ_L, γ_H, and c. D_0 is the cutoff distance as conventionally defined in the band-pass filter. γ_L and γ_H are the lowest and highest values of the filter and c controls the transitions between γ_L and γ_H. Based on Eq. (4.7), c along with D_0 (i.e., $D_0/\sqrt{c}$) determines the actual cutoff distance. This makes c the most important parameter of the system. Fig. 4.5 shows the image representation of the homomorphic filter for different c values and different D_0, γ_L, and γ_H values. It clearly shows the shape of the filter changes gradually with changing values of c and D_0. However, the shape of the filter does not change much with changing values of γ_L and γ_H. This indicates that c is the most important parameter in our system. On the other hand, every face image has different unknown amount of illuminations that require a different filter to reduce its illumination. Therefore, we produce the adaptive homomorphic filter based on each input face image by adjusting the c parameter.

The homomorphic filter operates in the frequency domain. Frequency is directly related to the rate of changes and intuitively associates with patterns of intensity variations in an image [80]. The origin of the Fourier transform contains the slowest varying frequency component which corresponds to the average gray level intensity of the image. While moving away from the origin, the frequency changes from low to high. Since illumination variations are mainly contained in the low frequency components, components around the origin contain most illumination changes. To this end, we consider a square window of components around the origin of Fourier transform and use a proportion of some of these low frequency components to determine the adaptive parameter c for the homomorphic filter. This proportion approximately indicates the slope of changes in the low frequency.

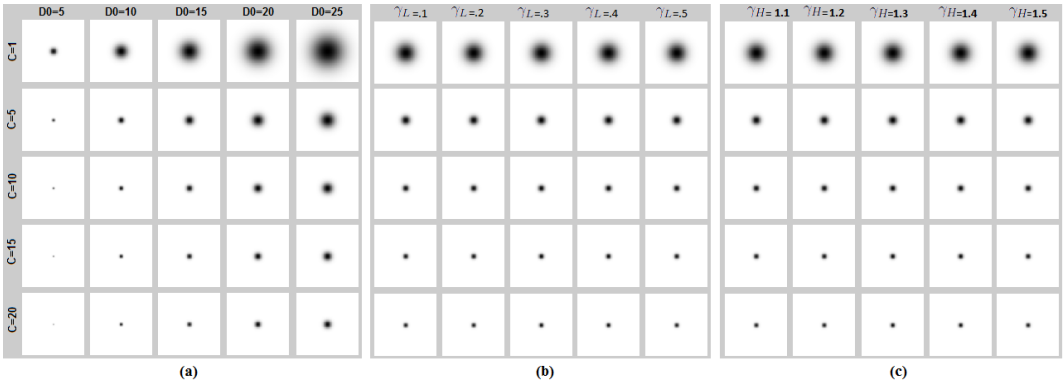

Fig. 4.5**:** The image representation of the homomorphic filter for five different c values versus (a) five different D_0 values with the empirically determined values of γ_L=.5 and γ_H=1.1, (b) five different γ_L values with the empirically determined values of γ_H=1.1 and D_0=15, and (c) five different γ_H values with the empirically determined values of γ_L=.5 and D_0=15.

Therefore, we define the parameter c as follows:

$$c = \frac{Mag_1}{Mag_2} \tag{4.11}$$

where Mag_i indicates the ith largest magnitude value within the square window. We use the magnitude to compute c since it measures the strength of the frequency components. The origin of the Fourier transform is excluded because it only represents the average intensity of the image. In our experiment, we empirically set the length of the square window to be 15% of the smallest dimension of the image. For example, we use a window of size 15×15 centering at the origin of Fourier transform for a face image of size 100×100. The other parameters are empirically set to be: γ_H=1.1, γ_L=0.5, and D_0=15. D_0 is the cutoff distance and is set to be the length of the window.

We also investigate the effects of different γ_L and γ_H on the face recognition performance when adaptive c is used. CMU-PIE, Yale B, and extended Yale B face databases have different charactritics and features. Therefore, a good performance on one of them for a method does not guarantee obtaining a good performance on the other one for the same method. Also, parameters of each method might need to be different for each database. The results of different γ_L and γ_H values on the three databases are summarized in Tables 4.1 through 4.3. Our experiments in Tables 4.1, 4.2, and 4.3 verify the choices of γ_L and γ_H do

not affect the performace of AH-ELDP too much. The values in these tables are the average face recognition accuracy of respective experiments yielded by the AH-ELDP method using seven different values for each γ_L and γ_H and the adaptive c values. In our experiments, we choose γ_L=.5 and γ_H=1.1 to have a good balance among three face databases.

We also conduct some experiments to evaluate the influence of the adaptive parameter c. Fig. 4.6 shows the performance of AH-ELDP for CMU-PIE, Yale B, and extended Yale B databases when a fixed value of c is used. It clearly demonstrates the different trends for three face databases. For example, the performance of AH-ELDP on the CMU-PIE database slightly improves when the value of c increases. On the contrary, the performance of AH-ELDP on the Yale B and the extended Yale B database improves when the value of c decreases. Therefore, the adaptive c introduced in Eq. (4.11) adjusts the homomorphic filter based on the illumination (low frequency) of each face image. Comparing the performace of AH-ELDP using the adaptive c (refer to Tables 4.1, 4.2, and 4.3) with the performance of AH-ELDP using the fixed c values (refer to Fig. 4.6), we can clearly see that the AH-ELDP method achieves a balanced performance using the adaptive c on all three databases.

We also conduct eight experiments on each of the three publicly available databases to investigate the influence of D_0 on the face recognition accuracy. Fig. 4.7 shows the face recognition accuracy for CMU-PIE, Yale B, and extended Yale B databases using different D_0 values including 1, 5, 10, 15, 20, 25, 30, and 35, respectively. It clearly confirms that the value of 15 is a good choice for D_0 for face images of 100×100 pixels.

4.4 Summary

We propose a three-step illumination-invariant face recognition method, called AH-ELDP, to preprocess face images with illumination variations. The proposed AH-ELDP method offers the following advantages:

- Using an adaptive c to adjust the homomorphic filter for each face image to reduce the illumination influence using the slope of changes in the low frequency components of the image.

Table 4.1: Recognition accuracy (%) of AH-ELDP with different γ_L and γ_H values and the adaptive c values and the empirically determined D_0 (i.e., 15) for the CUM-PIE face images.

γ_H/γ_L	**.1**	**.2**	**.3**	**.4**	**.5**	**.6**	**.7**
1.1	99.32	99.38	99.41	99.45	99.45	99.46	99.51
1.2	99.29	99.34	99.41	99.45	99.45	99.42	99.46
1.3	99.32	99.33	99.39	99.40	99.41	99.44	99.44
1.4	99.34	99.33	99.38	99.38	99.38	99.40	99.42
1.5	99.32	99.32	99.37	99.38	99.39	99.38	99.38
1.6	99.32	99.33	99.36	99.37	99.39	99.39	99.37
1.7	99.32	99.33	99.35	99.36	99.36	99.36	99.37

Table 4.2: Recognition accuracy (%) of AH-ELDP with different γ_L and γ_H values and the adaptive c values and the empirically determined D_0 (i.e., 15) for the Yale B face images.

γ_H/γ_L	**.1**	**.2**	**.3**	**.4**	**.5**	**.6**	**.7**
1.1	96.51	96.69	96.53	96.45	96.67	96.27	95.93
1.2	96.30	96.43	96.38	96.24	96.35	96.40	95.95
1.3	96.01	96.19	96.24	96.08	96.19	96.11	96.01
1.4	95.93	96.01	95.87	95.90	95.98	96.01	95.95
1.5	95.79	95.85	95.74	95.66	95.87	95.87	95.69
1.6	95.74	95.74	95.56	95.53	95.50	95.66	95.45
1.7	95.74	95.58	95.58	95.34	95.45	95.42	95.37

- Applying an interpolative enhancement function to stretch the contrast of the filtered image.

- Employing Kirsch compass masks to compute edge responses and considers the relations among eight directional numbers to obtain the AH-ELDP image which is robust against illumination variations and represents more valuable structural information from the neighborhood.

Table 4.3: Recognition accuracy (%) of AH-ELDP with different γ_L and γ_H values and the adaptive c values and the empirically determined D_0 (i.e., 15) for the extended Yale B face images.

γ_H/γ_L	**.1**	**.2**	**.3**	**.4**	**.5**	**.6**	**.7**
1.1	85.02	85.23	85.53	85.57	85.77	85.70	85.44
1.2	84.79	85.07	85.21	85.32	85.54	85.58	85.47
1.3	84.63	84.87	85.08	85.19	85.44	85.34	85.33
1.4	84.60	84.78	84.90	84.94	85.05	85.13	85.14
1.5	84.51	84.63	84.68	84.76	84.86	84.85	84.83
1.6	84.39	84.55	84.52	84.60	84.65	84.69	84.72
1.7	84.36	84.39	84.49	84.46	84.59	84.51	84.51

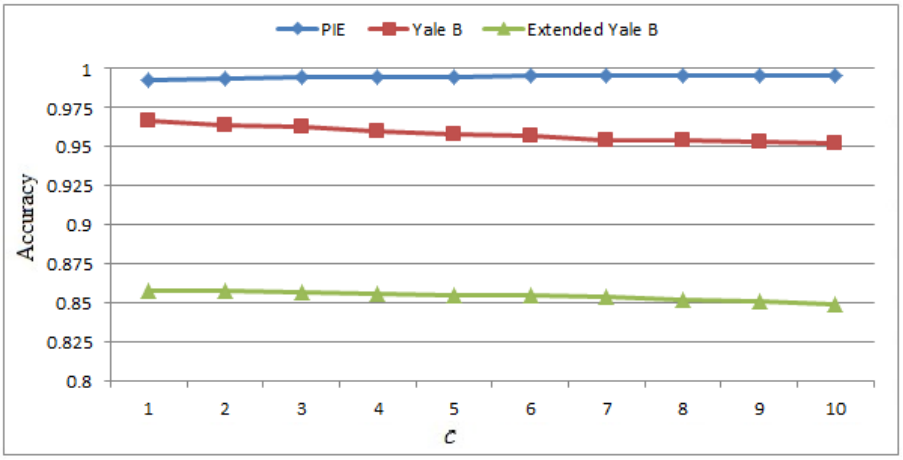

Fig. 4.6**:** Comparison of the recognition accuracy of the AH-ELDP method with different fixed values for c and the empirically determined values γ_L=.5, γ_H=1.1, and D_0 =15 for CMU-PIE, Yale B, and extended Yale B images.

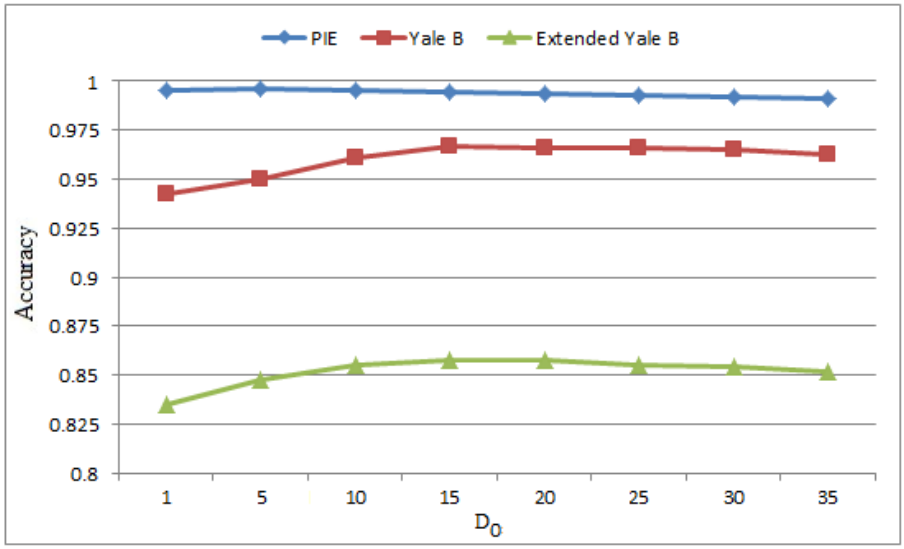

Fig. 4.7**:** Comparison of the recognition accuracy of the AH-ELDP method with different D_0 values for the adaptive c and the empirically determined values γ_L=.5 and γ_H=1.1 for CMU-PIE, Yale B, and extended Yale B images.

Chapter 5

Face Recognition under Varying Illuminations Using Logarithmic Fractal Dimension-based Complete Eight Local Directional Patterns

5.1 Introduction

The LDP-based approaches such as LDP, EnLDP, LDN, and the proposed ELDP method operate in the gradient domain to produce illumination-invariant representations. They use directional information to produce edge responses which are insensitive to lighting variations [45]. The proposed AH-ELDP method combines three steps to produce illumination-invariant images. It first applies an adaptive homomorphic filter on the original face image in the frequency domain to attenuate the low frequency signal (i.e., illuminance) and strengthen the high frequency signal (i.e., texture information) of the image. An adaptive c is used to adjust the homomorphic filter for each face image based on the slope of changes in the low frequency components of the image and reduce the influence of its illumination. Next, AH-ELDP applies an interpolative enhancement function to stretch the contrast of the filtered image. Finally, similar to the ELDP method, it uses Kirsch compass masks to compute edge responses and considers the relations among eight directional numbers to obtain the AH-ELDP image.

On the other hand, the fractal analysis (FA) as a type of texture analysis has been recently used in medical imaging [73–75] and signal processing [76]. The proposed LFD method performs a log function on a face image to expand dark pixels and compress bright pixels for partial illumination reduction. It then transfers the face image to the FD domain to produce the illumination-invariant representation by using the DBC algorithm.

In this chapter, we propose a new method which combines adaptive homomorphic

filtering, simplified logarithmic fractal dimension, and complete eight local directional patterns to produce illumination-invariant representations. The proposed method effectively encodes face images with various levels of illuminations. To this end, we first filter images using the adaptive homomorphic filter to partially reduce the illumination. We then use the simplified logarithmic fractal dimension transformation as an effective edge enhancer technique to enhance facial features and remove illumination to some point. We finally produce eight edge directional images using Kirsch compass masks and propose a complete ELDP (CELDP) which considers both directions and magnitudes of the edge responses to obtain illumination-invariant representations.

The contributions of the proposed method are:

1. Using the adaptive homomorphic filter to attenuate the low-frequency (i.e., illuminance) component of each face image.

2. Transforming filtered face images to the LFD domain to enhance facial features such as eyes, eyebrows, nose, and mouth while keeping noise at a low level.

3. Employing a gradient-based descriptor which considers the relations among directions and magnitudes of all eight directional edge responses to achieve robustness against illumination variations and noise and represent more valuable structural information from the neighborhood.

5.2 Methodology

The proposed method consists of three steps. The first step is to use the adaptive homomorphic filter to partially reduce the illumination. The second step is to apply the simplified LFD method to enhance facial features. The last step employs a complete ELDP (CELDP), which uses both direction and magnitude of edge responses to produce the illumination-invariant representation. Fig. 5.1 illustrates the framework of the proposed method to produce final matching scores for an input face image.

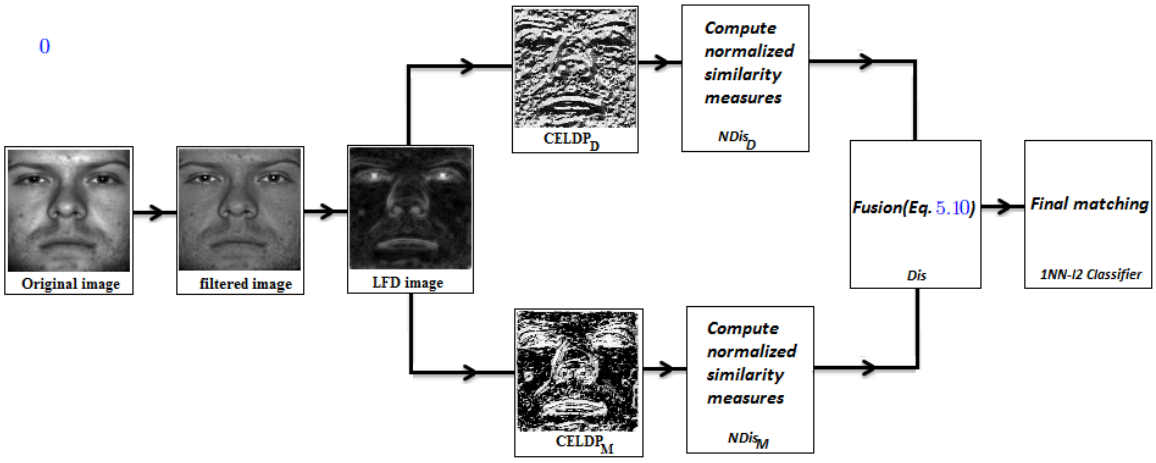

Fig. 5.1: The framework of the proposed method.

5.2.1 Adaptive homomorphic filter

In the first step, we apply the adaptive homomorphic filter on the face image. The adaptive homomorphic filter explained in the sub-section 4.2.1 is used to partially reduce the illumination. A face image I uses the Lambertian-reflectance model (i.e., $I(x,y) = R(x,y)L(x,y)$, where $I(x,y)$, $R(x,y)$, and $L(x,y)$ are respectively the pixel intensity, reflectance, and illuminance at position (x,y)) to employ the homomorphic filter $H(u,v)$ as a Gaussian high-pass filter which strengthens the high-frequency signal and attenuates the low-frequency signal [65, 80].

The process of applying the filter is represented in Fig. 5.2. It indicates that the logarithmic function is used to separate the R and L functions so that the operation can be performed in the frequency domain. The Fourier transform result is then multiplied with the adaptive homomorphic filter to reduce the illuminance. The filtered image is obtained by the exponential function of the inverse Fourier transform.

The homomorphic filter $H(u,v)$ is represented as follows:

$$H(u,v) = (\gamma_H - \gamma_L)[1 - exp^{-cD^2(u,v)/D_0^2}] + \gamma_L \quad (i.e.,\ Eq.\ 4.7) \tag{5.1}$$

where $D(u,v)$ is the distance from (u,v) to the origin of the centered Fourier transform, D_0 is the cutoff distance measured from the origin, $\gamma_L < 1$ and $\gamma_H > 1$ are the parameters of

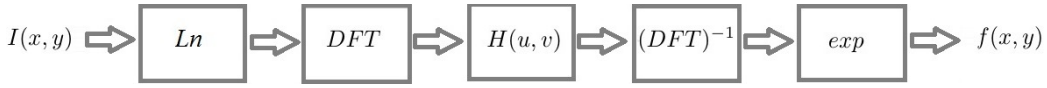

Fig. 5.2**:** Homomorphic filtering approach to partially reduce the illuminance of a face image (Fig. 4.1).

the filter, and c is a constant to control the sharpness of the slope of the filter function as it transitions between γ_H and γ_L. Since c along with D_0 (i.e., $D_0/\sqrt{c}$) determines the actual cutoff distance, it is considered as the most important parameter of the filter. As proposed in the chapter 4, the c parameter is computed by:

$$c = \frac{Mag_1}{Mag_2} \quad (i.e.,\ Eq.\ 4.11) \tag{5.2}$$

where Mag_i indicates the ith largest magnitude value (i.e., the strength of the frequency components) at non-zero frequency locations within a square window centering at the origin of the Fourier transform.

5.2.2 Logarithmic fractal dimension

In the second step, we perform the LFD method on top of the filtered image. The LFD method presented in Chapter 2, first, applies a log function on the filtered image to expand the values of dark pixels and compress the values of bright pixels. Then, it transfers the obtained image to the FD domain using the DBC algorithm. In the following, we shortly present the LFD method without referring readers to Chapter 2.

LFD computes a sequence of two-dimensional matrices Mat's to overlay the filtered image f of size $M \times N$ at each pixel (x, y) as follows:

$$Mat_{d-1}(x,y) = (\text{floor}[\frac{max(W_d(x,y)) - min(W_d(x,y))}{d}] + 1)(\frac{d_{max}}{d})^2 \quad (i.e.,\ Eq.\ 2.1) \tag{5.3}$$

where d is the scaling factor with a minimum value of 2 and maximum value d_{max} that represents how much a specific structure of pixels is self-similar to its surroundings. $W_d(x, y)$

is a $d \times d$ neighborhood block whose center is at location (x, y) of an image. When d is an odd number, the center is in the middle of the neighborhood block. When d is an even number, the center is to the left and above the middle of the neighborhood block. *max* and *min* operations obtain the highest and lowest intensity values in the image block covered by the $d \times d$ neighborhood block centering at the location (x, y) of an image, respectively. Specifically, for each scaling factor d, the center of a sliding neighborhood block of size $d \times d$ is positioned at each pixel in an image. The maximum intensity value and the minimum intensity value of the image portion covered by the sliding neighborhood block are used to compute the new value at the corresponding pixel location following Eq. (5.3).

Each two-dimensional matrix Mat_{d-1} of size $M \times N$ is then converted to a one-dimensional column vector of size $(M \times N) \times 1$ and the converted $(d_{max} - 1)$ column vectors are concatenated to obtain the two-dimensional matrix V, which has $(M \times N)$ rows and $(d_{max} - 1)$ columns. Each row in the matrix V contains the values computed by Eq. (5.3) at a particular location in an image across scales from 2 to d_{max}. For example, $v_{(x,y)}$ denotes the obtained values at location (x, y) of an image across scales from 2 to d_{max}. A log function is then applied on all the elements of V and the respective scaling factor d. The LFD image is finally computed as follows:

$$LFD(x, y) = \frac{S_v(x, y)}{S} \quad (i.e.,\ Eq.\ 2.2) \tag{5.4}$$

where S and S_v are the sums of squares as follows:

$$S = \sum_{d=1}^{d_{max}-1} (log(d))^2 - \frac{(\sum_{d=1}^{d_{max}-1} log(d))^2}{d_{max} - 1} \quad (i.e.,\ Eq.\ 2.3) \tag{5.5}$$

$$S_v(x, y) = \sum_{d=1}^{d_{max}-1} log(d)(log(v_{(x,y)),d})) - \frac{(\sum_{d=1}^{d_{max}-1} log(d))(\sum_{d=1}^{d_{max}-1} log(v_{(x,y),d}))}{d_{max} - 1} \quad (i.e.,\ Eq.\ 2.4) \tag{5.6}$$

The original LFD method (Chapter 2) sets d_{max} to be 10 and recommends to set d_{max} to be between 9 and 12 for face images. However, we observe that the structure

of pixels in a face image changes slowly across scales. Therefore, skipping each adjacent scale does not lose much information and we propose to set $d = 2, 4, 6, ..., d_{max}$. In this way, the number of overlay two-dimensional matrices Mat (i.e., the column dimension of the matrix V) decreases to a half (i.e., $d_{max}/2$) and these targeted scales are involved in the computation in Eq. (5.5) and Eq. (5.6). Obviously, this reduces computational time to transfer an image to its FD image by a half and therefore reduces the run time of the method significantly.

5.2.3 Complete ELDP (CELDP)

The LFD image obtained from the adaptive homomorphic filtered image is fed into the ELDP process. To this end, we compute eight edge directional images using Kirsch compass masks to produce the ELDP image. Given a pixel (x, y) in the LFD image, if $\mathbb{I}_i(x, y)$ shows the ith (i=0,1,...,7) directional number computed by the ith Kirsch compass mask M_i, the respective ELDP value at the pixel (x, y) is defined as follows:

$$ELDP(x, y) = \sum_{i=0}^{i=7} D(\mathbb{I}_i(x, y))(2)^i \tag{5.7}$$

where

$$D(z) = \begin{cases} 1 & z > 0 \\ 0 & \text{otherwise} \end{cases}$$

Fig. 5.3 shows the process of producing an ELDP value of a pixel using its eight directional numbers. Their directions (i.e., positive and negative) are used to obtain the ELDP image, while their respective magnitude are discarded. We notice that a directional number with a higher value is expected to contain more information about the texture changes in its respective direction. Therefore, we propose to consider and integrate the influence of the magnitudes of the directional numbers in the ELDP process. To this end, we adopt the idea of the completed LBP (CLBP) method [82], which considers magnitudes of difference between the neighborhood pixels and the central pixel as well as their corresponding signs,

to design the proposed CELDP. We define CELDP as the composition of direction component (CELDP$_D$) and magnitude component (CELDP$_M$), which respectively are defined as follows:

$$CELDP_D(x, y) = ELDP(x, y) \tag{5.8}$$

$$CELDP_M(x, y) = \sum_{i=0}^{i=7} M(abs(\mathbb{I}_i(x, y)), m)(2)^i \tag{5.9}$$

where

$$M(z, m) = \begin{cases} 1 & z > m \\ 0 & \text{otherwise} \end{cases}$$

and m is the average of absolute values of all the eight edge directional images. Basically, CELDP$_D$ is the proposed ELDP, explained in Chapter 3, and contains the information of the directions of edge responses. CELDP$_M$ has a similar 8-bit binary code structure. However, it thresholds the absolute value of all eight magnitude values for each pixel based on the average of absolute values of all the eight edge images (i.e., m). Fig. 5.4 shows the process of producing an CELDP$_M$ value of a pixel using its eight magnitudes.

Fig. 5.5 illustrates the process of the proposed method to produce the final CELDP$_D$ and CELDP$_M$ images from an input face image. Clearly, both CELDP$_D$ and CELDP$_M$ images contain important features which are illumination-invariant. Fig. 5.6 shows sample original images from the CMU-PIE face database and their corresponding CELDP$_D$ and CELDP$_M$ images. It clearly shows the CELDP method reduces the effect of illumination and produces illumination-insensitive images since all CELDP$_D$ (and CELDP$_M$) images look alike as shown in Fig. 5.6(b) (and Fig. 5.6(c)).

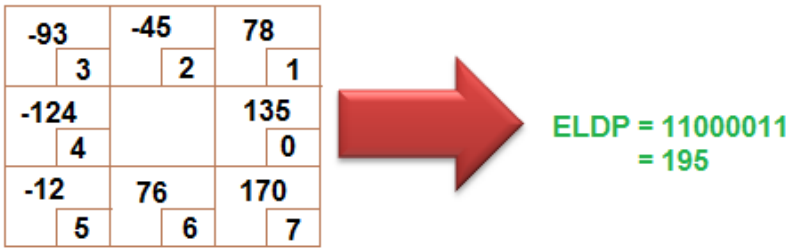

Fig. 5.3**:** ELDP code computation for eight directional numbers computed using eight Kirsch masks for a pixel. This computation is also used for CELDP$_D$.

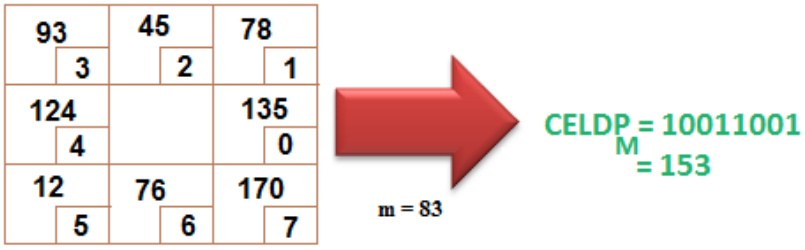

Fig. 5.4**:** CELDP$_M$ code computation for eight magnitudes of directional numbers using the average of absolute values of all the eight edge images as the threshold (i.e., 83).

5.2.4 Classification

Similar to the Weberface, the one nearest neighbor (1NN) method with l_2 norm (1NN-l_2) is used as the classifier. 1NN-l_2 simply assigns an input image to its nearest neighbor image in the database. Therefore, recognition accuracy only shows the effectiveness of preprocessing methods in producing illumination-invariant features. To utilize the CELDP$_D$ and CELDP$_M$ images of each input image for the classification task, we first compute similarity distances Dis_D and Dis_M between the CELDP$_D$ and CELDP$_M$ images of the input image and their corresponding CELDP$_D$ and CELDP$_M$ preprocessed images for all images in the database, respectively. We then independently normalize all distances Dis_D and Dis_M to be in the range of [0, 1]. We denote the normalized distances as $NDis_D$ and $NDis_M$, respectively. The final similarity measure Dis is computed by fusing $NDis_D$ and $NDis_M$ as follows:

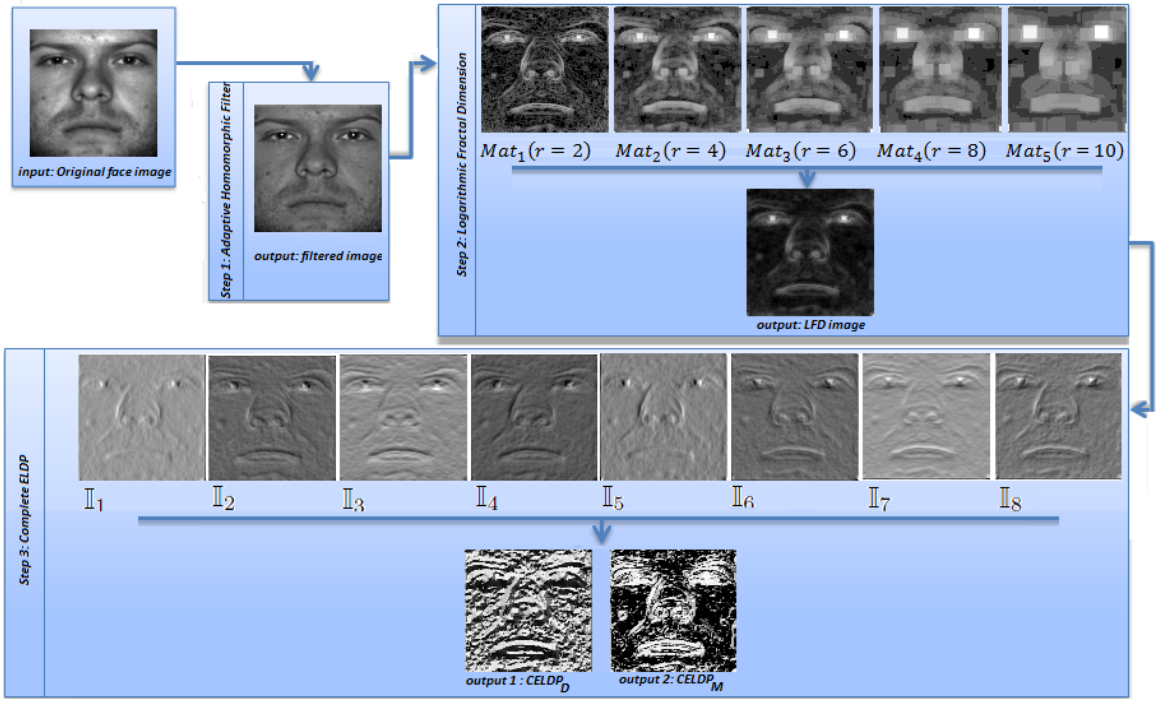

Fig. 5.5**:** Illustration of the three steps of the proposed method.

$$Dis = \alpha \times NDis_D + (1 - \alpha) \times NDis_M \tag{5.10}$$

where α is the weight of the direction component of the edge images at the decision level. We set α to be 0.80, as Guo et al. [82] proved that the local difference error made by using the sign component of the LBP code to reconstruct an image is only 1/4 of the respective error made by the magnitude component of the LBP code. This valid argument is used in our system to compute the final similarity measure as a proportion of similarity measures obtained from CELDP_D and CELDP_M components.

5.3 Summary

This chapter proposes a method to produce illumination-invariant representation for

face images using logarithmic fractal dimension-based complete eight local directional patterns. The proposed CELDP method offers the following advantages:

- Using adaptive homomorphic filter to partially reduce the illumination by attenuating the low-frequency component of the face image.
- Applying the simplified logarithmic fractal dimension operation as an effective edge enhancer technique to enhance facial features such as eyes, eyebrows, nose, and mouth while keeping noise at a low level with a significantly reduced computational cost.
- Considering the relations among directions and magnitudes of all eight directional edge responses to represent more valuable structural information from the neighborhood and achieve robustness against illumination variations and noise.

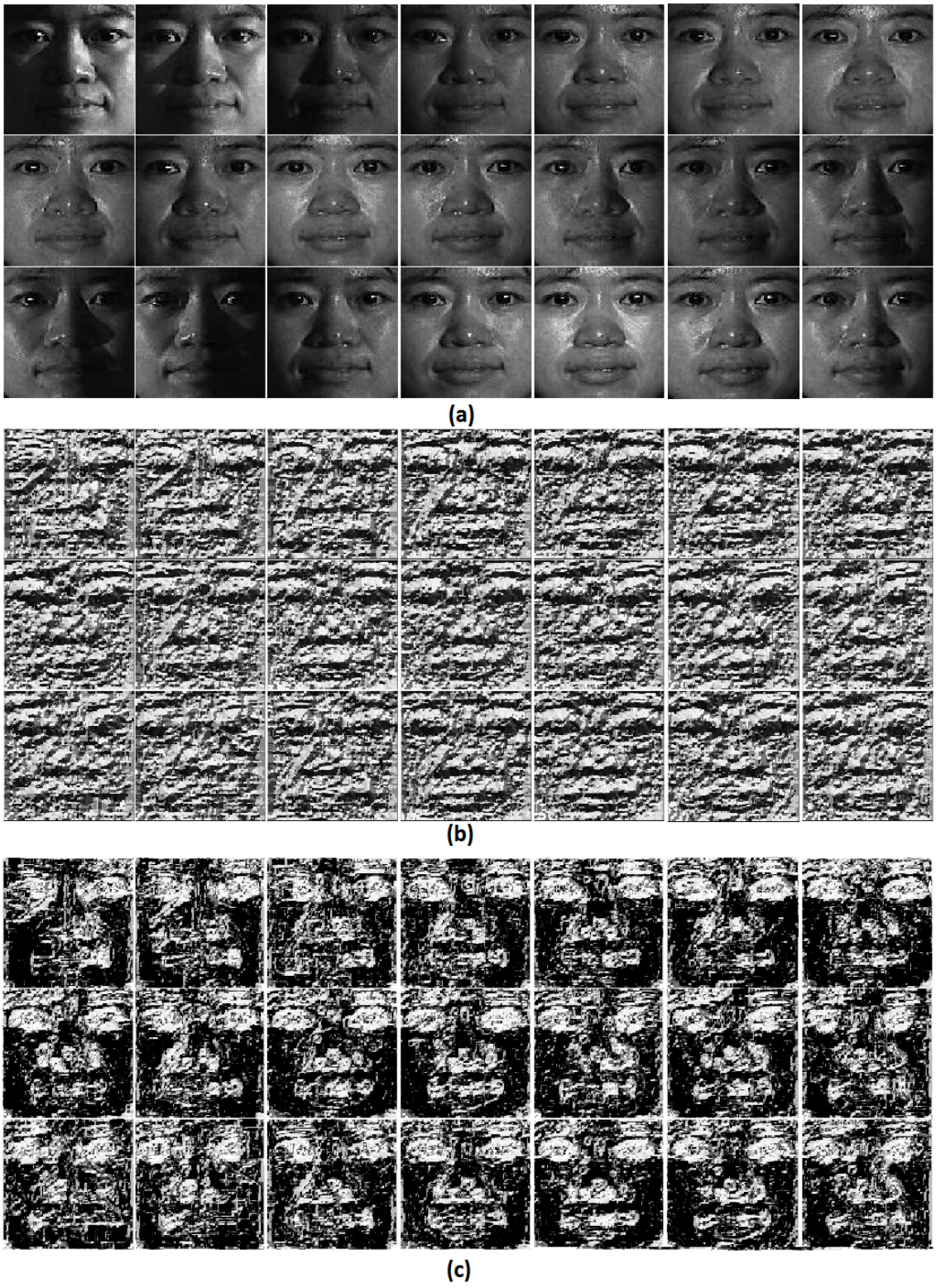

Fig. 5.6**:** Illustration of sample images of the CMU-PIE database and their CELDP images. (a) 21 samples for a subject (b) Corresponding CELDP$_D$ images (c) Corresponding CELDP$_M$ images.

Chapter 6

Experimental results

6.1 Experimental settings

As explained in Chapter 1, we evaluate the proposed methods by conducting experiments on four publicly available image databases with large illumination variations (i.e., Yale B, extended Yale B, CMU-PIE, and AR databases [32, 69, 70]) as well as the Honda UCSD video database [71]. We also illustrate the verification and discrimination ability of our methods using the ROC curves.

We compare the proposed methods with six recent state-of-the-art methods such as Gradientface, Weberface, LBP, LDP, EnLDP, and LDN. Fig. 6.1 demonstrates six original images from the Yale B face database, their corresponding preprocessed images generated by six compared state-of-the-art methods and the LFD, ELDP, AH-ELDP, CELDP$_D$ and CELDP$_M$ images generated by the proposed methods. All of these methods are implemented in MATLAB and their parameters are set as recommended by the respective researchers. For the LBP method, we apply the uniform LBP operator with 8 pixels in a circle of radius of 2. We also use the Gaussian filter with the Standard Deviation (SD) of 0.3 to smooth the image for Gradientface and ELDP methods and use the Gaussian filter with the SD of 1 for Weberface method, since the above methods achieve decent accuracy for all four databases using these values. Also, the parameters of the proposed system are set to be: γ_H=1.1, γ_L=0.5, D_0=15, and d_{max}=10. However, when images are of size 64×64, we set D_0 to be proportion to this size (i.e., D_0=64×0.15=9.6).

6.2 Results on the Yale B face database

As mentioned earlier in Chapter 1, we conduct six experiments for each compared method. In each experiment, we use one of the six face images per subject in S0 for

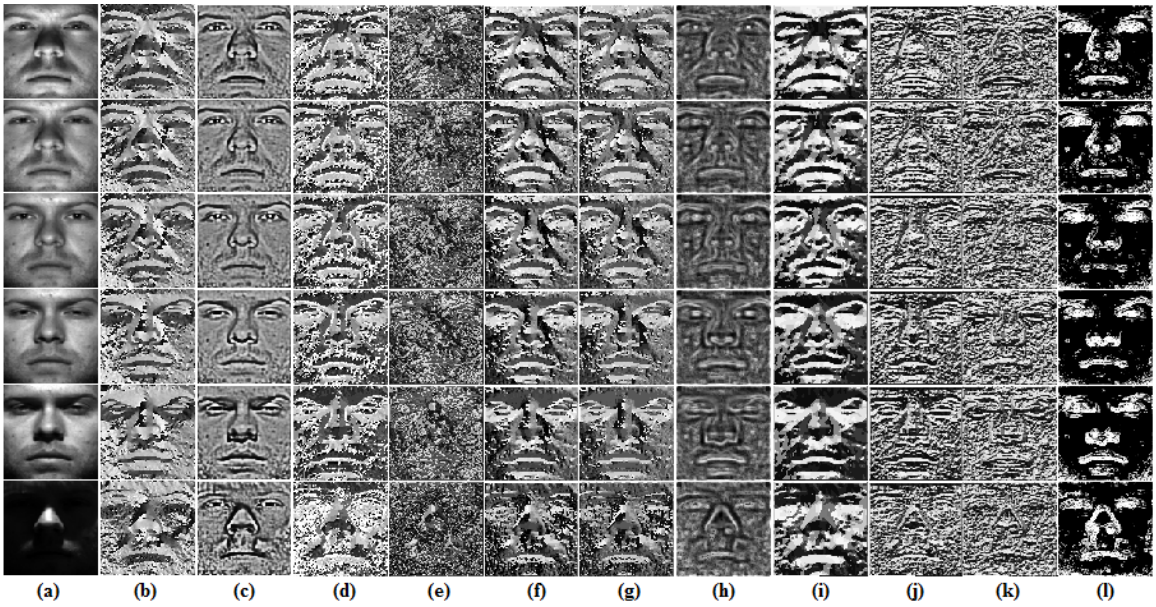

Fig. 6.1**:** Comparison of preprocessed images obtained by different methods. (a) Original face images in S0 of the Yale B face database. Illumination-invariant images preprocessed by the methods of (b) Gradientface, (c) Weberface, (d) LBP, (e) LDP, (f) EnLDP, (g) LDN, (h) LFD, (i) ELDP, (j) AH-ELDP, (k) CELDP$_D$, and (l) CELDP$_M$.

training, and the remaining five images per subject in S0 (50 images in total) and all the images in each of the other five subsets for testing. We then compute the average face recognition accuracy of each subset across all the six experiments. Table 6.1 summarizes the average recognition accuracy and its SD of each subset for the nine state-of-the-art compared methods and the proposed CELDP method. For each experiment, S0' contains the images in S0 excluding the training image. We also include the average accuracy together with its SD for all subsets including S0', S1, S2, S3, S4, and S5 (i.e., 630 testing images in total) in the last column of Table 6.1. The proposed CELDP method outperforms the compared methods with the best face recognition accuracy of 99.47% and the smallest SD of 0.004. It improves the face recognition accuracy of Gradientface, Weberface, LBP, LDP, EnLDP, LDN, LFD, ELDP, and AH-ELDP by 9.18%, 7.52%, 10.92%, 15.69%, 15.73%, 20.72%, 4.56%, 8.85%, and 2.90%, respectively.

6.3 Results on the extended Yale B face database

Using the same experimental settings for the Yale B database, we conduct six experiments for each compared method. Table 6.2 summarizes the average recognition accuracy and its SD of each subset for the nine state-of-the-art compared methods and the proposed CELDP method. For each experiment, S0' contains the images in S0 excluding the training image. We also include the average accuracy and its SD for S0', S1, S2, S3, S4, and S5 (i.e., 2376 testing images in total) in the last column of Table 6.2. The proposed CELDP method outperforms the other nine compared methods with the best face recognition accuracy of 94.55% and the smallest SD of 0.02, while the second highest accuracy of 85.77% is obtained by the proposed AH-ELDP method. It improves the face recognition accuracy of Gradientface, Weberface, LBP, LDP, EnLDP, LDN, LFD, ELDP, and AH-ELDP by 20.05%, 16.27%, 28.74%, 32.83%, 29.01%, 38.09%, 14.04%, 18.41%, and 10.24%, respectively.

6.4 Results on the CMU-PIE face database

Fig. 5.6 shows all 21 images for a subject from this database and their corresponding CELDP$_D$ and CELDP$_M$ images. We conduct 21 experiments to evaluate the performance of our proposed methods. In the ith experiment, we use the image i from each subject as the reference image and all the other 20 images as the test images. Fig. 6.2 shows the recognition accuracy of different methods under each reference set. Obviously, the proposed method outperforms the other methods for all reference images except for the AH-ELDP method which achieves comparable accuracy as the proposed CELDP method. Table 6.3 summarizes the average recognition accuracy and its SD of the ten compared methods for all reference images. The proposed CELDP method achieves the highest average recognition accuracy of 99.53% and the smallest SD of 0.01, while the second highest accuracy of 99.45% is obtained by the AH-ELDP method. Compared with the other nine methods, the proposed CELDP method improves the face recognition accuracy of Gradientface, Weberface, LBP, LDP, EnLDP, LDN, LFD, ELDP, and AH-ELDP by 2.46%, 5.10%, 4.36%, 19.31%, 6.10%, 10.05%, 1.71%, .60% and .08%, respectively. It clearly shows the effectiveness of the proposed method.

Table 6.1: Recognition accuracy (%) and corresponding SD in parentheses for Yale B face images.

Method	S0'	S1	S2	S3	S4	S5	Ave.
Gradientface	83.67 (.10)	95.62(.06)	90.83(.09)	84.86 (.10)	91.00(.09)	95.56(.05)	91.11(.06)
Weberface	87.33(.12)	93.96(.09)	90.17(.12)	89.17(.06)	89.83(.08)	98.33(.01)	92.51(.06)
LBP	83.33(.08)	96.25(.05)	91.83(.09)	85.28(.07)	88.50(.07)	90.93(.05)	89.68(.04)
LDP	81.67(.12)	92.71(.10)	86.67(.17)	87.22(.03)	84.00(.03)	84.07(.08)	85.98(.03)
EnLDP	79.67(.11)	92.92(.10)	87.00(.13)	81.81(.10)	85.33(.12)	87.13(.09)	85.95(.05)
LDN	77.00(.13)	91.25(.13)	84.50(.16)	80.14(.08)	81.33(.11)	80.93(.10)	82.40(.05)
LFD	92.33(.13)	95.83(.10)	93.50(.13)	93.06(.10)	95.67(.06)	97.59(.03)	95.13(.08)
ELDP	83.33(.12)	95.42(.06)	89.83(.10)	84.86(.09)	90.50(.10)	95.46(.05)	91.38(.06)
AH-ELDP	93.67(.06)	99.58(.01)	96.50(.05)	94.31(.06)	96.17(.04)	98.15(.02)	96.67(.03)
CELDP	**99.33**(.01)	**100.00**(00)	**99.50**(.01)	**99.03**(.01)	**99.67**(.01)	**99.44**(.00)	**99.47**(.004)

Table 6.2: Recognition accuracy (%) and corresponding SD in parentheses for extended Yale B face images.

Method	S0'	S1	S2	S3	S4	S5	Ave.
Gradientface	68.25(.15)	86.54(.19)	79.69 (.21)	73.05 (.07)	77.59(.13)	82.17 (.13)	78.76 (.07)
Weberface	71.84(.17)	84.38(.15)	78.55(.21)	77.43(.06)	78.68(.12)	88.24(.05)	81.32(.09)
LBP	63.07(.17)	82.45(.18)	76.32(.24)	67.52(.07)	72.54(.10)	75.17(.11)	73.44(.07)
LDP	67.89(.18)	84.60(.23)	77.10(.25)	73.31(.11)	65.96(.06)	64.32(.11)	71.18(.08)
EnLDP	65.61(.19)	86.21(.19)	77.41(.21)	68.93(.07)	71.58(.13)	71.23(.17)	73.29(.08)
LDN	61.58(.21)	83.17(.24)	74.30(.26)	65.07(.09)	66.27(.10)	64.08(.16)	68.47(.09)
LFD	79.30(.23)	85.88(.24)	84.82(.26)	83.03(.21)	83.33(.17)	81.21(.06)	82.91(.17)
ELDP	70.53(.13)	87.76(.11)	81.14(.14)	72.94(.09)	78.25(.15)	83.70(.13)	79.85(.09)
AH-ELDP	77.46(.10)	88.26(.05)	84.82(.12)	81.89(.06)	86.32(.11)	89.79(.06)	85.77(.05)
CELDP	**92.46**(.06)	**94.07**(.03)	**94.34**(.06)	**93.98**(.03)	**96.05**(.03)	**95.02**(.01)	**94.55**(.02)

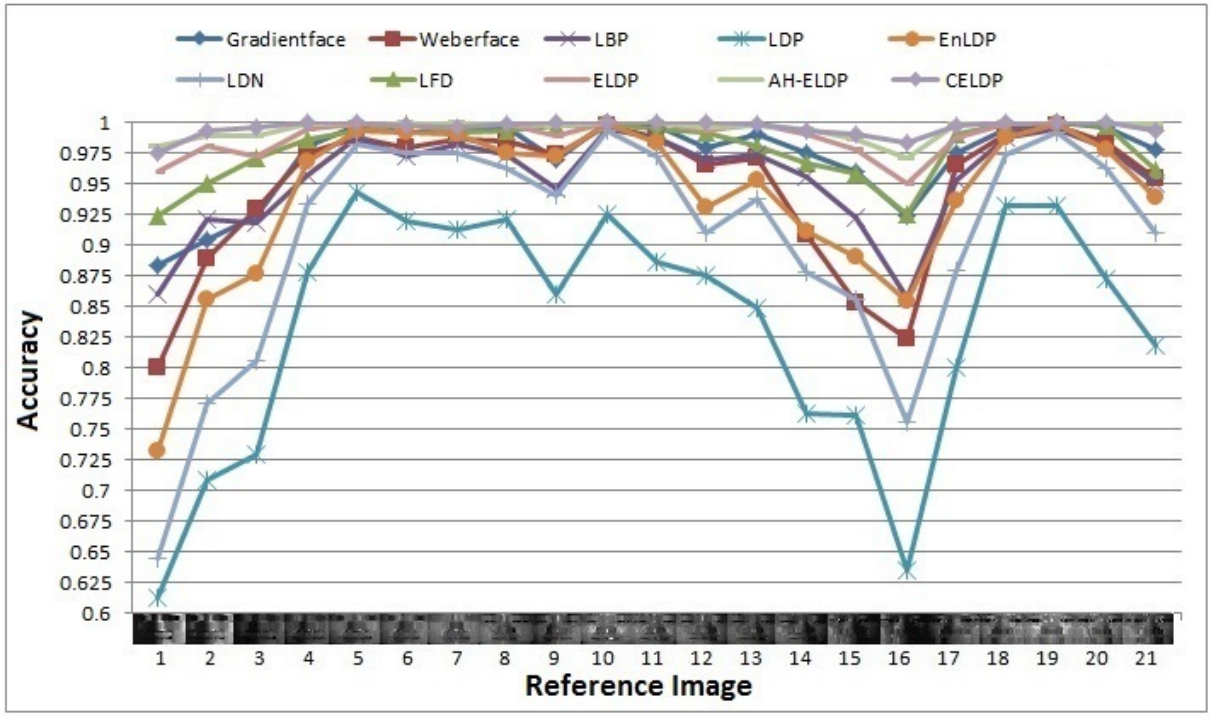

Fig. 6.2**:** Comparison of the recognition accuracy of ten methods for CMU-PIE face images.

Table 6.3: Average recognition accuracy (%) and corresponding SD in parentheses for CMU-PIE face images.

Gradientface	Weberface	LBP	LDP	EnLDP	LDN	LFD	ELDP	AH-ELDP	**CELDP**
97.14(.03)	94.70(.06)	95.37(.04)	83.42(.10)	93.81(.07)	90.44(.09)	97.86(.02)	98.91(.01)	99.45(.01)	**99.53(.01)**

6.5 Results on the AR face database

We conduct eight experiments to evaluate the performance of the proposed methods. In the ith experiment, we use the ith image from each subject as the reference image and all the other seven images as the test images. Table 6.4 summarizes the average recognition accuracy of the ten compared methods for all reference images.

The proposed CELDP method achieves the highest average recognition accuracy of 86.63%, while the second highest accuracy of 85.95% is obtained by the proposed AH-ELDP method. Compared with the other nine methods, the proposed CELDP method improves the face recognition accuracy of Gradientface, Weberface, LBP, LDP, EnLDP, LDN, LFD, ELDP, and AH-ELDP by 19.49%, 27.47%, 14.20%, 12.51%, 1.64%, 4.44%, 1.59%, 1.26% and .08%, respectively.

6.6 Results on the Honda UCSD video database

As explained in Chapter 1, in a typical face recognition system, the face detection step is employed first to detect faces from images/video frames. Facial landmarks such as locations of eyes, eyebrows, nose, and mouth are then localized which can be used for the normalization process. In this research, the Viola-Jones face detector [4] is used to detect faces in each frame. It is a robust real-time face detector to distinguish faces from non-faces. Also, in our experiment, we use the most recently proposed facial landmark localization technique, *Chehra v.1* [11], to localize the landmarks in each face. *Chehra* is a fully-automatic real-time face and eyes landmark detection and tracking software capable of handling faces under uncontrolled natural setting. It determines 49 landmarks for each face. Fig. 6.3 shows a video frame from the Honda database together with the face detection and landmark localization results, where the the detected face is positioned in the green bounding box and its 49 landmarks are marked by "*".

Illumination-invariant methods require the face images to be in the frontal view and the same size. Therefore, a series of preprocessing steps should be applied. Since the bounding box for a detected face contains lots of non-facial features, especially the background, each face needs to be cropped. If all faces are in the frontal view similar to the frontal view face

Table 6.4: Average recognition accuracy (%) and corresponding SD in parentheses for AR face images.

Methods	Accuracy
Gradientface	72.50(.03)
Weberface	67.96(.03)
LBP	75.86(.03)
LDP	77.00(.01)
EnLDP	85.23(.01)
LDN	82.95(.02)
LFD	85.27(.02)
ELDP	85.55(.01)
AH-ELDP	85.95(.01)
CELDP	**86.63(.01)**

shown in Fig. 6.3, we only need to crop and resize all faces. We can use the landmarks, i.e., the location of eyebrows, left and right corners of the left and right eyes (i.e., external corners), and the lower part of mouth, to automatically crop the face. Fig. 6.4 shows the cropped version of the detected face in Fig. 6.3.

However, in the Honda UCSD video database, most of faces are in non-frontal views. Fig. 6.5 (a) shows a typical detected face in the non-frontal view and its landmarks. More operations should be applied to normalize the pose and estimate the frontal view so the processed face can be used in our system as well as other compared systems. Here, we use landmarks for different tasks. To this end, we first compute the roll value for each detected face. Roll is defined as the angle between the horizontal line and the line connecting the external corners of the eyes. It is shown in Fig. 6.5 (b). We then determine whether the face is in the frontal/right/left view. As shown in Fig. 6.5(c), y_1 is computed as the distance from the nose tip to the external corner of the right eye, and y_2 is computed as the distance from the nose tip to the external corner of the left eye. If y_1 is equal to y_2, the face is in the frontal view. If y_1 is greater than y_2, the face is in the right view. Otherwise, it is in the left view.

Next, we rotate the face image using the computed roll value. We also need to rotate the landmarks to their new locations. Fig. 6.5(d) shows the result after rotating the face

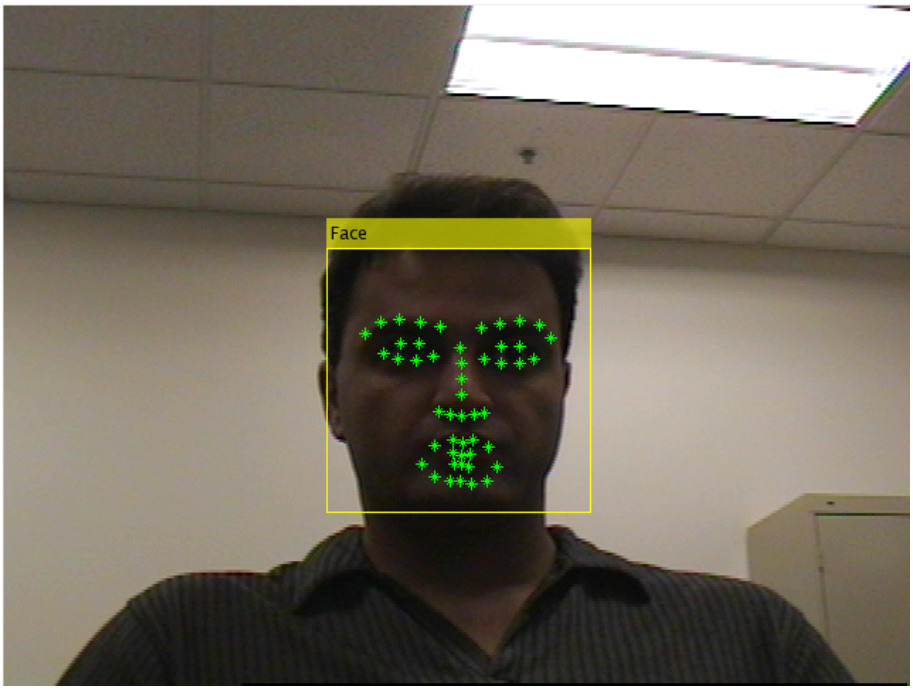

Fig. 6.3**:** A detected face from the Honda database. Each of 49 landmarks extracted by *Chehra v.1* is represented by "*".

and its respective landmarks. In the proposed system, we assume the face is in the left view before we estimate its frontal view. Therefore, if the face is in the right view (i.e., $y_1 > y_2$), we flip the face to obtain its respective left view. Fig. 6.5(f) shows the flipped face for Fig. 6.5(e), which is now in the left view. Then, we appropriately stretch each row from the left side of the face to estimate the frontal view for the half of the face. The stretching operation needs to keep important regions of a face in an appropriate scale with respect to each other. Therefore, we divide the half face to four vertical bands. The boundary of each band is determined using specific landmarks or midpoint between a pair of specific landmarks. For every row in the cropped image, its respective four segments in the four bands are stretched to have an equal size to produce the estimated frontal view of the half face. The bands and the employed landmarks to estimate the frontal view of the half face together with the estimated frontal view are shown in Fig. 6.5(g). Finally, the right view of the face is reconstructed by flipping the estimated frontal left view. The final estimated frontal view of the face is shown in Fig. 6.5(h).

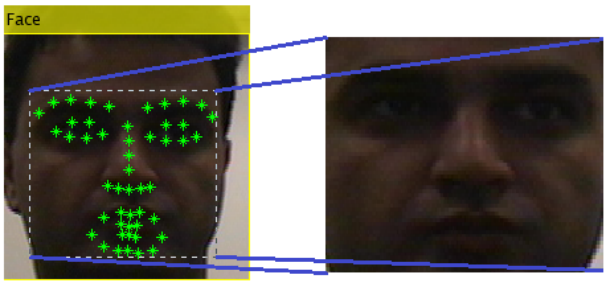

Fig. 6.4**:** Cropped face for the detected face in Fig. 6.3 using the facial landmarks.

After estimating the frontal view for each face, all face images are resized to the size of 64×64. In this experiment, we use every other ten frame to evaluate the performance of compared methods and exclude other frames due to the large number of frames. As a result, there are 42 face images for each subject in this database on average. We can then conduct the experiment on this reduced database. For the Honda UCSD video database, we use five estimated frontal views of each subject from the training subset for training and use estimated frontal views from the testing subset for testing. Since subjects have facial expressions, we apply Gabor filters as a post-processing step to deal with expression variations. We also use sparse representation-based classifier (SRC) [30,31] as the classifier since more than one image are used for training. SRC takes advantage of sparse representation to achieve better performance when each subject has multiple images in training.

Table 6.5 summarizes the recognition accuracy of the ten compared methods. The proposed CELDP method achieves the highest average recognition accuracy of 76.89%, while the second highest accuracy of 75.15% is obtained by the proposed ELDP method. Compared with the other nine methods, the proposed CELDP method improves the face recognition accuracy of Gradientface, Weberface, LBP, LDP, EnLDP, LDN, LFD, ELDP, and AH-ELDP by 7.21%, 3.22%, 3.19%, 27.41%, 5.72%, 9.33%, 2.34%, 2.31% and 7.03%, respectively.

It should be mentioned that the compared methods behave differently on the Yale B, extended Yale B, CMU-PIE, AR, and Honda UCSD databases. For example, when not considering the four proposed methods, the Gradientface method achieves the second best accuracy of 91.11% and 78.76% on the Yale B and Extended Yale B databases, the

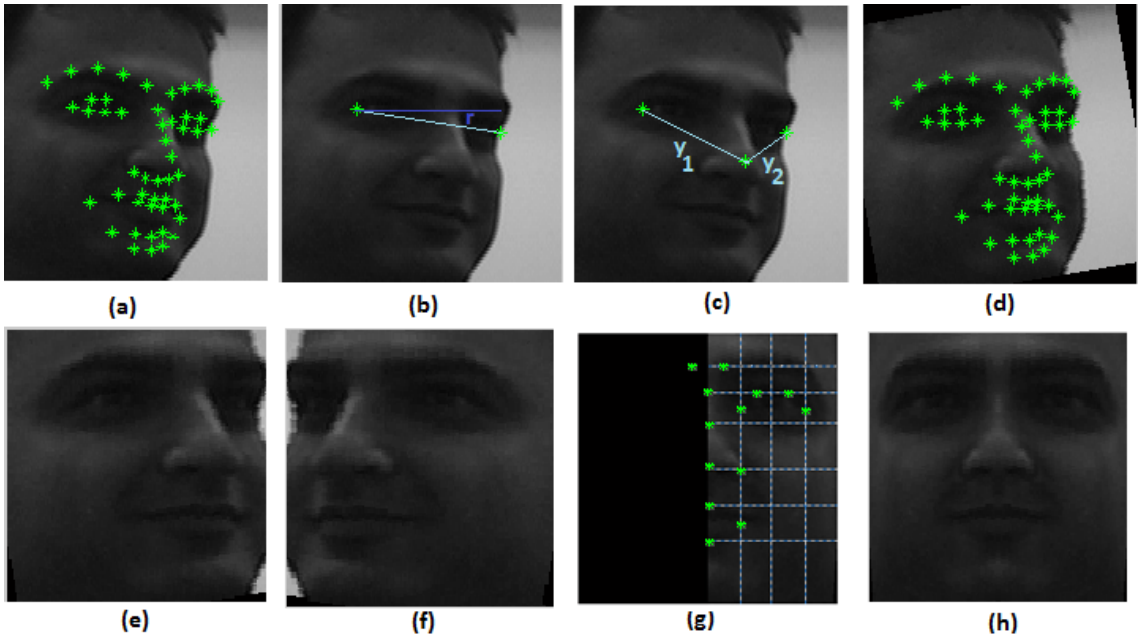

Fig. 6.5**:** Pose normalization process. (a) A detected face and its respective landmarks; (b) Angle between the external corners of eyes and the horizontal axis determines the roll value (here, r=9.52°); (c) distances between the external corners of eyes and the nose tip determine whether the face is in the left view or in the right view(e.g., $y_1 > y_2$ means it is in the right view); (d) the rotated face using the roll value and its respective landmarks which are also rotated to the new locations; (e) cropped face using the landmarks; (f) the cropped face is flipped; (g) the estimated frontal view of the left side of the face; and (h) the right side of the face is reconstructed using the left side of the face.

best accuracy of 97.14% on the CMU-PIE database, the fifth best accuracy of 72.50% on the AR database, and the fourth best accuracy of 71.72% on the Honda UCSD database, respectively. Our proposed methods also behave differently on different databases. For instance, LFD achieves the third best accuracy on Yale B, extended Yale B, and Honda UCSD databases, while it achieves the fourth best accuracy on CMU-PIE and AR databases. Clearly, the proposed CELDP consistently outperforms all the other methods by achieving the highest recognition accuracy on all these databases with different illuminations.

6.7 ROC curves and computational time

We plot ROC curves for the four databases. For Yale B and extended Yale B databases, we use each of the six images in S0 as the reference image and all the rest as probe images and compute the average of respective similarity measures as true and false scores. In a

Table 6.5: Recognition accuracy (%) for Honda UCSD video face images.

Methods	Accuracy
Gradientface	71.72
Weberface	74.49
LBP	74.51
LDP	60.35
EnLDP	72.73
LDN	70.33
LFD	75.13
ELDP	75.15
AH-ELDP	71.84
CELDP	**76.89**

similar setting, we use each of the 21 images for each subject in the CMU-PIE database as the reference image and all the rest as probe images and compute the average of respective similarity measures as true and false scores. Similarly, we use each of the eight images for each subject in the AR database as the reference image and all the rest as probe images and compute the average of respective similarity measures as true and false scores. Appendix explains how to compute the TPR and FPR to plot a ROC curve for a given database with M subjects and K images per subject when one reference image from each subject in the training dataset is used for training and the rest images are used for testing. The reference image in the Appendix changes from 1 to R, where R is 6 for Yale B and extended Yale B databases, 21 for the CMU-PIE database, and 8 for the AR database.

For the Honda UCSD video database, since we use multiple images from each subject for training and conduct one experiment using the remaining images as testing images, we compute the average of similarity measures between each probe image from subject i and training images for the subject i as true scores. We then compute the average of similarity measures between each probe image from subject i and training images for all the subjects excluding the subject i as false scores.

Figs. 6.6, 6.7, 6.8, 6.9, and 6.10 compare the respective ROC curves of ten compared methods for Yale B, extended Yale B, CMU-PIE, AR, and Honda UCSD databases, respectively. Clearly, the proposed CELDP method achieves the best verification and discrim-

ination ability compared with the other state-of-the-art methods since it has the largest area under the ROC curve. The proposed AH-ELDP method achieves the second best verification and discrimination ability on four face image databases.

The proposed methods, also, have a comparable computational time with the compared state-of-the-art methods. Fig. 6.11 shows average computational runtime for a face image of the size of 100×100. The LFD method has the highest computational time, which is also employed in our system. However, we efficiently simplify the method by skipping adjacent scales while using the same d_{max} of 10.

6.8 Summary

This chapter includes our extensive experiments which evaluate the four proposed methods and compare them with six state-of-the-art methods proposed by other researchers. Our extensive experiments show the final proposed CELDP method outperforms other state-of-the-art methods and achieves the best overall face recognition accuracy of 99.39%, 94.35%, 99.53%, and 86.63% on Yale B, Extended Yale B, CMU-PIE, and AR face databases, respectively, when using a simple 1NN-l_2 classifier and one image per subject for training. It also includes experimental results using the Honda UCSD video database. Our CELDP method outperforms the other methods and achieves the best face recognition accuracy of 76.89% when using the SRC classifier and five images per subject for training.

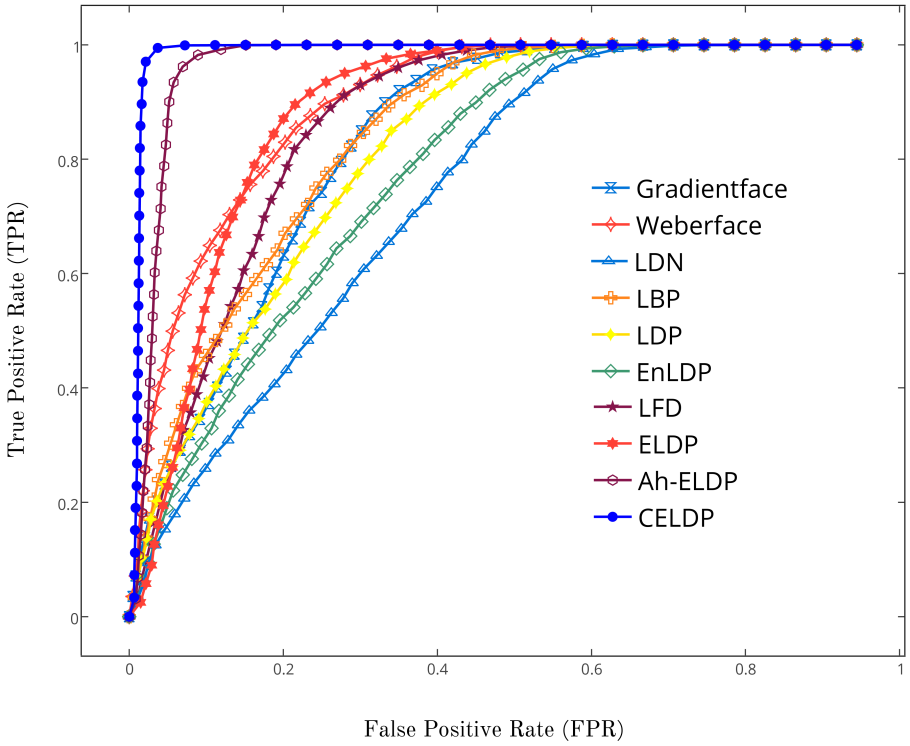

Fig. 6.6**:** Comparison of ROC curves of the ten compared methods for the Yale B database.

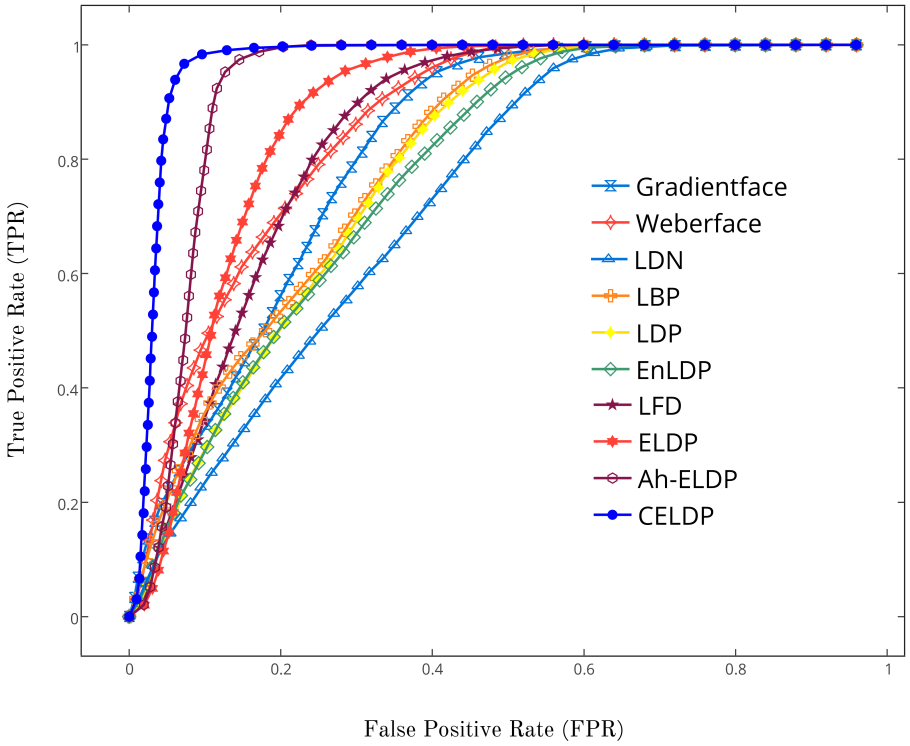

Fig. 6.7**:** Comparison of ROC curves of the ten compared methods for the extended Yale B database.

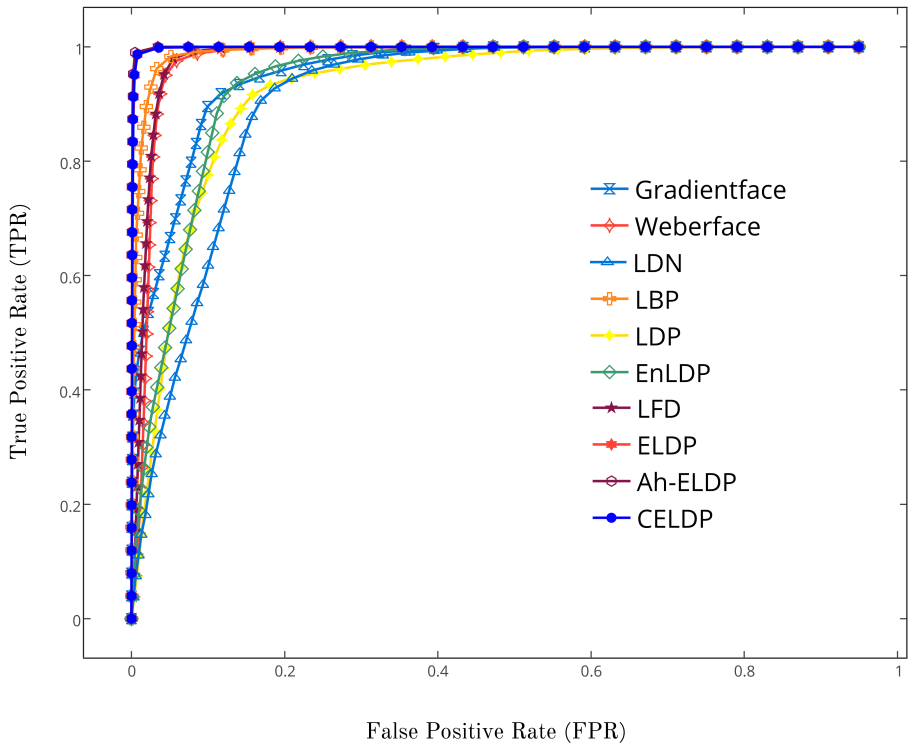

Fig. 6.8**:** Comparison of ROC curves of the ten compared methods for the CMU-PIE database.

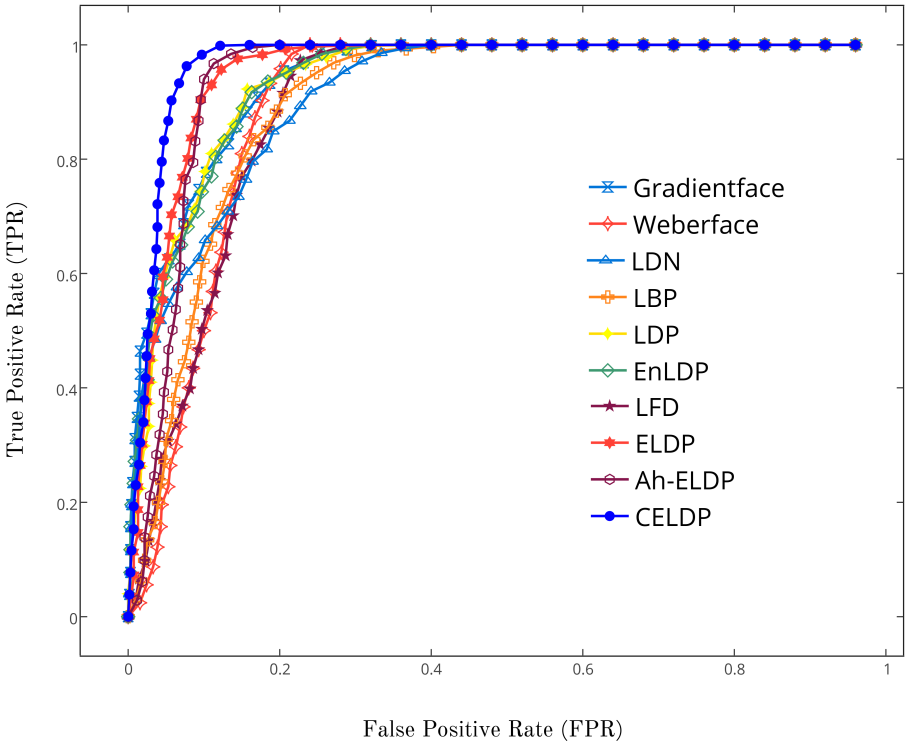

Fig. 6.9**:** Comparison of ROC curves of the ten compared methods for the AR database.

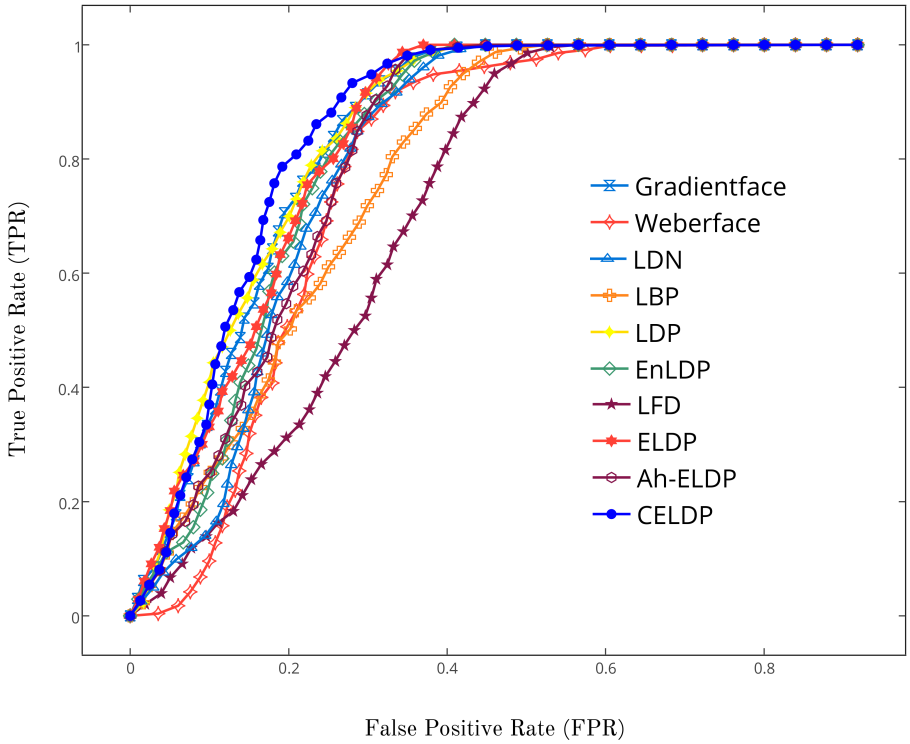

Fig. 6.10: Comparison of ROC curves of the ten compared methods for the Honda UCSD video database.

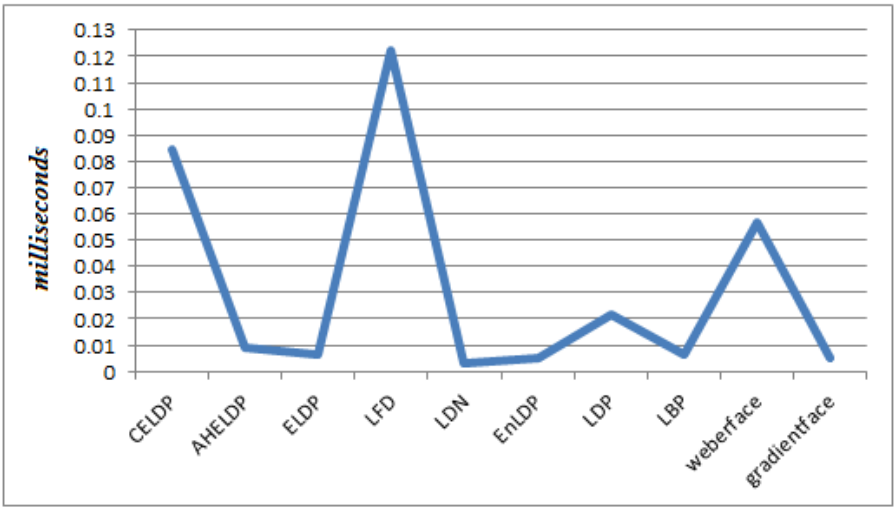

Fig. 6.11: Average computational runtime for a face image (100×100).

Chapter 7

Conclusions

The necessity of security, privacy, and access control has increased the need for face recognition systems. Illumination variations are one of the most crucial factors to affect performance of face recognition systems. As a result, many researchers have studied face recognition under varying illuminations and many illumination-invariant methods have been proposed to produce illumination-invariant representations. This dissertation studies practical face recognition systems with a focus on faces under illumination variations and proposes four evolving methods to produce illumination-insensitive representations. The four proposed methods are logarithmic fractal dimensions (LFD), eight local directional patterns (ELDP), adaptive homomorphic local directional patterns (AH-ELDP), and complete ELDP (CELDP). Major contributions for the proposed methods are summarized as follows:

- The logarithmic fractal dimensions (LFD) method transfers face images to the fractal dimension (FD) domain and produces illumination-invariant representations. The LFD image enhances facial features such as eyes, eyebrows, nose, and mouth while keeping noise at a low level.

- The eight local directional patterns (ELDP) method is a modified version of LDP approaches which extracts more edge information by utilizing all eight directional edge images, while LDP approaches such as LDP, EnLDP, and LDN only use up to three directional numbers from the eight edge images.

- The adaptive homomorphic local directional pattern (AH-ELDP) method, first attenuates the low-frequency (i.e., illuminance) component of each original face image with an adaptive filter. Then, it effectively enhances the contrast of the filtered image using

an interpolative enhancement function. Finally, the enhanced filtered image incorporates the ELDP code scheme to produce an illumination-invariant representation.

- The complete ELDP (CELDP) method employs the adaptive homomorphic filter to partially reduce the illumination of a face image. Then, it applies a simplified LFD operation as an effective edge enhancer technique to enhance facial features. Finally, it considers the relation among magnitudes of all eight directional edge responses as well as the relation among their directions to extract more valuable structural information from the neighborhood and achieve robustness against illumination variations and noise.

All the four proposed methods are individual systems that can be used to produce illumination-invariant representations. However, AH-ELDP employs ELDP as the final stage to obtain the AH-ELDP image. The CELDP method also combines the simplified LFD method with the adaptive homomorphic filter and ELDP to obtain illumination-invariant representation.

Our extensive experiments on four publicly available face databases as well as a standard video database show that the CELDP method outperforms six state-of-the-art methods. Our evaluations using ROC curves also verifies the CELDP and AH-ELDP methods have the best and the second best verification and discrimination ability compared to the other methods considered.

References

[1] A. F. Abate, M. Nappi, D. Riccio, and G. Sabatino, "2d and 3d face recognition: A survey," *Pattern Recognition Letters*, vol. 28, pp. 1885–1906, 2007.

[2] H. Han, S. Shan, X. Chen, and W. Gao, "A comparative study on illumination preprocessing in face recognition," *Pattern Recognition*, vol. 46, no. 6, pp. 1691–1699, 2013.

[3] M. Yang, D. Roth, and N. Ahuja, "A snow-based face detector," in *Advances in Neural Information Processing Systems 12*. MIT Press, 2000, pp. 855–861.

[4] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Computer Vision and Pattern Recognition*, pp. I–511 –I–518, 2001.

[5] G. C. Luh, "Face detection using combination of skin color pixel detection and viola-jones face detector," in *Int. Conf. Machine Learning and Cybernetics*, vol. 1, pp. 364–370, July 2014.

[6] C. Papageorgiou, M. Oren, and T. Poggio, "A general framework for object detection," in *Int. Conf. Computer Vision*, pp. 555–562, Jan. 1998.

[7] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *J. Computer and System Sciences*, vol. 55, no. 1, pp. 119–139, Aug. 1997.

[8] M. Valstar, B. Martinez, X. Binefa, and M. Pantic, "Facial point detection using boosted regression and graph models," in *IEEE Conf. Computer Vision And Pattern Recognition*, pp. 2729–2736, 2010.

[9] B. Martinez, M. Valstar, X. Binefa, and M. Pantic, "Local evidence aggregation for regression-based facial point detection," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 35, no. 5, pp. 1149–1163, May 2013.

[10] A. Asthana, S. Zafeiriou, S. Cheng, and M. Pantic, "Robust discriminative response map fitting with constrained local models," in *Computer Vision and Pattern Recognition*, 2013.

[11] ——, "Incremental face alignment in the wild," in *Computer Vision and Pattern Recognition*, pp. 1859–1866, 2014.

[12] M. De Marsico, M. Nappi, D. Riccio, and H. Wechsler, "Robust face recognition for uncontrolled pose and illumination changes," *IEEE Trans. Systems, Man, and Cybernetics: Systems*, vol. 43, no. 1, pp. 149–163, Jan. 2013.

[13] C. Liu and H. Wechsler, "Gabor feature based classification using the enhanced fisher linear discriminant model for face recognition," *IEEE Tran. on Image Processing*, vol. 11, no. 4, pp. 467–476, April 2002.

[14] V. Štruc and N. Pavešić, "The complete gabor-fisher classifier for robust face recognition," *EURASIP J. Advances in Signal Processing*, vol. 2010, pp. 31:1–31:13, Feb. 2010.

[15] Z. Lai, D. Dai, C. Ren, and K. Huang, "Multiscale logarithm difference edgemaps for face recognition against varying lighting conditions," *IEEE Trans. Image Processing*, vol. 24, no. 6, pp. 1735–1747, June 2015.

[16] X. Tan and B. Triggs, "Enhanced local texture feature sets for face recognition under difficult lighting conditions," *IEEE Trans. Image Processing*, vol. 19, no. 6, pp. 1635–1650, 2010.

[17] M. Turk and A. Pentland, "Eigenfaces for recognition," *J. Cognitive Neuroscience*, vol. 3, no. 1, pp. 71–86, 1991.

[18] B. A. Draper, K. Baek, M. S. Bartlett, and J. Beveridge, "Recognizing faces with {PCA} and {ICA}," *Computer Vision and Image Understanding*, vol. 91, no. 12, pp. 115–137, 2003, special Issue on Face Recognition.

[19] V. Perlibakas, "Face recognition using principal component analysis and wavelet packet decomposition," *Informatica*, vol. 15, no. 2, pp. 243–250, 2004.

[20] M. Bartlett, J. R. Movellan, and T. Sejnowski, "Face recognition by independent component analysis," *IEEE Trans. Neural Networks*, vol. 13, no. 6, pp. 1450–1464, Nov 2002.

[21] M.-H. Yang, N. Ahuja, and D. Kriegman, "Face recognition using kernel eigenfaces," in *Int. Conf. Image Processing*, vol. 1, 2000, pp. 37–40.

[22] K. I. Kim, K. Jung, and H. J. Kim, "Face recognition using kernel principal component analysis," *IEEE Signal Processing Letters*, vol. 9, no. 2, pp. 40–42, Feb. 2002.

[23] D. L. Swets and J. Weng, "Using discriminant eigenfeatures for image retrieval," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 18, no. 8, pp. 831–836, 1996.

[24] Q. Liu, R. Huang, H. Lu, and S. Ma, "Face recognition using kernel-based fisher discriminant analysis," in *Fifth IEEE Int. Conf. Automatic Face and Gesture Recognition*, 2002, pp. 197–201.

[25] Z. Sun, J. Li, and C. Sun, "Kernel inverse fisher discriminant analysis for face recognition," *Neurocomputing*, vol. 134, no. 0, pp. 46–52, 2014.

[26] D. Masip and J. Vitria, "Shared feature extraction for nearest neighbor face recognition," *IEEE Trans. Neural Networks*, vol. 19, no. 4, pp. 586–595, April 2008.

[27] F. Song, Z. Guo, and Q. Chen, "Two-dimensional nearest neighbor classifiers for face recognition," in *Int. Conference on Systems and Informatics*, pp. 2682–2686, May 2012.

[28] S. Lawrence, C. Giles, A. C. Tsoi, and A. Back, "Face recognition: a convolutional neural-network approach," *IEEE Trans. Neural Networks*, vol. 8, no. 1, pp. 98–113, Jan. 1997.

[29] T. H. Le, "Applying artificial neural networks for face recognition," *Advances in Artificial Neural Systems*, vol. 2011, p. 16 pages, Jan. 2011.

[30] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 31, no. 2, pp. 210–227, Feb. 2009.

[31] A. Wagner, J. Wright, A. Ganesh, Z. Zhou, H. Mobahi, and Y. Ma, "Toward a practical face recognition system: robust alignment and illumination by sparse representation," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 34, no. 2, pp. 372–386, Feb. 2012.

[32] A. M. Martinez and R. Benavente, "Pca versus lda," *CVC Technical Report 24*, 1998.

[33] C. E. Thomaz and G. A. Giraldi, "A new ranking method for principal components analysis and its application to face image analysis," *Image and Vision Computing*, vol. 28, no. 6, pp. 902–913, 2010.

[34] M. R. Faraji and X. Qi, "An effective neutrosophic set-based preprocessing method for face recognition," in *Proc. Int. Conf. Multimedia Expo*, 2013.

[35] S. M. Pizer, E. P. Amburn, J. D. Austin, R. Cromartie, A. Geselowitz, T. Greer, B. ter Haar Romeny, J. B. Zimmerman, and K. Zuiderveld, "Adaptive histogram equalization and its variations." *Computer Vision, Graphics, and Image Processing*, vol. 39, no. 3, pp. 355–368, 1987.

[36] H.-D. Liu, M. Yang, Y. Gao, and C. Cui, "Local histogram specification for face recognition under varying lighting conditions," *Image and Vision Computing*, vol. 32, no. 5, pp. 335–347, 2014.

[37] S. Shan, W. Gao, B. Cao, and D. Zhao, "Illumination normalization for robust face recognition against varying lighting conditions," in *IEEE Int. Workshop on Analysis and Modeling of Faces and Gestures*, pp. 157–164, 2003.

[38] F. Samaria and A. Harter, "Parameterisation of a stochastic model for human face identification," in *IEEE Workshop on Applications of Computer Vision*, Dec. 1994, pp. 138–142.

[39] A. Lanitis, "Facial biometric templates and aging: problems and challenges for artificial intelligence," in *AIAI-2009 Workshops Proc.*, 2009.

[40] The fg-net aging database, nov. 2007. [Online]. Available: http://webmail.cycollege.ac.cy/~alanitis/fgnetaging/

[41] Y. Adini, Y. Moses, and S. Ullman, "Face recognition: the problem of compensating for changes in illumination direction," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 19, pp. 721–732, 1997.

[42] T. Ahonen, A. Hadid, and M. Pietikainen, "Face description with local binary patterns: application to face recognition," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 28, no. 12, pp. 2037–2041, 2006.

[43] T. Jabid, M. Kabir, and O. Chae, "Local directional pattern (ldp) for face recognition," in *Digest of Technical Papers Int. Conf. Consumer Electronics*, 2010, pp. 329–330.

[44] F. Zhong and J. Zhang, "Face recognition with enhanced local directional patterns," *Neurocomputing*, vol. 119, no. 0, pp. 375–384, 2013.

[45] A. Ramirez Rivera, R. Castillo, and O. Chae, "Local directional number pattern for face analysis: face and expression recognition," *IEEE Trans. Image Processing*, vol. 22, no. 5, pp. 1740–1752, 2013.

[46] A. E. Essa and V. K. Asari, "Local directional pattern of phase congruency features for illumination invariant face recognition," *Proc. SPIE*, vol. 9094, pp. 90 940G–90 940G–8, 2014.

[47] Z. Lei, M. Pietikainen, and S. Li, "Learning discriminant face descriptor," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 36, no. 2, pp. 289–302, 2014.

[48] R. A. Kirsch, "Computer determination of the constituent structure of biological images," *Computers and Biomedical Research*, vol. 4, no. 3, pp. 315–328, 1971.

[49] H. Wang, S. Li, Y. Wang, and J. Zhang, "Self quotient image for face recognition," in *Int. Conf. Image Processing*, vol. 2, pp. 1397–1400, Oct. 2004.

[50] H. Wang, S. Z. Li, and Y. Wang, "Face recognition under varying lighting conditions using self quotient image," in *Proc. Sixth IEEE Int. Conference on Automatic Face and Gesture Recognition*, 2004, pp. 819–824.

[51] Z. Chen, C. Liu, F. Chang, X. Han, and K. Wang, "Illumination processing in face recognition," *Int. J. Pattern Recognition and Artificial Intelligence*, vol. 28, no. 5, p. 1456011, 2014.

[52] T. Chen, W. Yin, X. S. Zhou, D. Comaniciu, and T. Huang, "Illumination normalization for face recognition and uneven background correction using total variation based image models," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, June 2005, pp. 532–539.

[53] ——, "Total variation models for variable lighting face recognition," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 28, no. 9, pp. 1519–1524, 2006.

[54] T. Zhang, B. Fang, Y. Yuan, Y. Y. Tang, Z. Shang, D. Li, and F. Lang, "Multiscale facial structure representation for face recognition under varying illumination," *Pattern Recognition*, vol. 42, no. 2, pp. 251–258, 2009.

[55] C.-P. Chen and C.-S. Chen, "Lighting normalization with generic intrinsic illumination subspace for face recognition," in *Tenth IEEE Int. Conference on Computer Vision*, vol. 2, Oct. 2005, pp. 1089–1096.

[56] X. Xie and K.-M. Lam, "An efficient illumination normalization method for face recognition," *Pattern Recognition Letters*, vol. 27, no. 6, pp. 609–617, 2006.

[57] Y. K. Park, S. L. Park, and J. K. Kim, "Retinex method based on adaptive smoothing for illumination invariant face recognition," *Signal Processing*, vol. 88, no. 8, pp. 1929–1945, 2008.

[58] W. Chen, M. J. Er, and S. Wu, "Illumination compensation and normalization for robust face recognition using discrete cosine transform in logarithm domain," *IEEE Trans. Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 36, no. 2, pp. 458–466, April 2006.

[59] X. Xie, W.-S. Zheng, J. Lai, P. Yuen, and C. Suen, "Normalization of face illumination based on large-and small-scale features," *IEEE Trans. Image Processing*, vol. 20, no. 7, pp. 1807–1821, July 2011.

[60] T. Zhang, Y. Y. Tang, B. Fang, Z. Shang, and X. Liu, "Face recognition under varying illumination using gradientfaces," *IEEE Trans. Image Processing*, vol. 18, no. 11, pp. 2599–2606, 2009.

[61] B. Wang, W. Li, W. Yang, and Q. Liao, "Illumination normalization based on weber's law with application to face recognition," *IEEE Signal Processing Letters*, vol. 18, no. 8, pp. 462–465, 2011.

[62] Y. Wu, Y. Jiang, Y. Zhou, W. Li, Z. Lu, and Q. Liao, "Generalized weber-face for illumination-robust face recognition," *Neurocomputing*, vol. 136, no. 0, pp. 262–267, 2014.

[63] A. Baradarani, Q. M. J. Wu, and M. Ahmadi, "An efficient illumination invariant face recognition framework via illumination enhancement and DD-DTcWT filtering," *Pattern Recognition*, vol. 46, no. 1, pp. 57–72, 2013.

[64] J. Chen, S. Shan, C. He, G. Zhao, M. Pietikainen, X. Chen, and W. Gao, "Wld: a robust local image descriptor," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1705–1720, 2010.

[65] Y. S. Huang and C. Y. Li, "An effective illumination compensation method for face recognition," in *Advances in Multimedia Modeling*, ser. Lecture Notes in Computer Science, vol. 6523, pp. 525–535, 2011.

[66] R. Basri and D. W. Jacobs, "Lambertian reflectance and linear subspaces," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 25, no. 2, pp. 218–233, 2003.

[67] X. Zou, J. Kittler, and K. Messer, "Illumination invariant face recognition: A survey," in *First IEEE Int. Conf. Biometrics: Theory, Applications, and Systems*, pp. 1–8, Sept. 2007.

[68] S. Nikan and M. Ahmadi, "Local gradient-based illumination invariant face recognition using local phase quantisation and multi-resolution local binary pattern fusion," *IET Image Processing*, 2014.

[69] T. Sim, S. Baker, and M. Bsat, "The cmu pose, illumination, and expression (pie) database," in *Proc. IEEE Int. Conf. Automatic Face and Gesture Recognition*, pp. 46–51, 2002.

[70] A. S. Georghiades, P. N. Belhumeur, and D. Kriegman, "From few to many: illumination cone models for face recognition under variable lighting and pose," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 23, no. 6, pp. 643–660, 2001.

[71] K. Lee, J. Ho, M. Yang, and D. Kriegman, "Visual tracking and recognition using probabilistic appearance manifolds," *Computer Vision and Image Understanding*, 2005.

[72] A. M. Martinez and A. C. Kak, "Pca versus lda," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 23, no. 2, pp. 228–233, 2001.

[73] C.-C. Chen, J. DaPonte, and M. Fox, "Fractal feature analysis and classification in medical imaging," *IEEE Trans. Medical Imaging*, vol. 8, no. 2, pp. 133–142, 1989.

[74] O. Al-Kadi and D. Watson, "Texture analysis of aggressive and nonaggressive lung tumor ce ct images," *IEEE Trans. Biomedical Engineering*, vol. 55, no. 7, pp. 1822–1830, 2008.

[75] P. Kim, K. Iftekharuddin, P. Davey, M. Toth, A. Garas, G. Hollo, and E. Essock, "Novel fractal feature-based multiclass glaucoma detection and progression prediction," *IEEE J. Biomedical and Health Informatics*, vol. 17, no. 2, pp. 269–276, 2013.

[76] A. Zlatintsi and P. Maragos, "Multiscale fractal analysis of musical instrument signals with application to recognition," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 21, no. 4, pp. 737–748, 2013.

[77] N. Sarkar and B. Chaudhuri, "An efficient differential box-counting approach to compute fractal dimension of image," *IEEE Trans. Systems, Man and Cybernetics*, vol. 24, no. 1, pp. 115–120, 1994.

[78] S. Buczkowski, S. Kyriacos, F. Nekka, and L. Cartilier, "The modified box-counting method: Analysis of some characteristic parameters," *Pattern Recognition*, vol. 31, no. 4, pp. 411–418, 1998.

[79] C. J. Traina, A. Traina, L. Wu, and C. Faloutsos, "Fast feature selection using fractal dimension," in *Proc. 15th Brazil Symposium Databases*, pp. 158–171, 2000.

[80] R. C. Gonzalez and R. E. Woods, "Digital Image Processing" Prentice Hall, 2007.

[81] M. H. Fazel Zarandi, M. R. Faraji, and M. Karbasian, "Interval type-2 fuzzy expert system for prediction of carbon monoxide concentration in mega-cities," *Applied Soft Computing*, vol. 12, no. 1, pp. 291–301, 2012.

[82] Z. Guo, D. Zhang, and D. Zhang, "A completed modeling of local binary pattern operator for texture classification," *IEEE Trans. Image Processing*, vol. 19, no. 6, pp. 1657–1663, 2010.

Appendix

Compute TPR and FPR from a face database for ROC curves

To compute the true positive rate (TPR) and the false positive rate (FPR) and then plot a ROC curve, we need the respective scores and labels. Thus, we first explain how scores and labels in our experiments are computed. Similar to our experiments on Yale B, extended Yale B, CMU-PIE, and AR databases, we assume there is only one reference (training) image among R images for each subject in the training database and we conduct R experiments on each database (R is 6 for the Yale B and the extended Yale B databases, 21 for the CMU-PIE database, and 8 for the AR database).

Given a face database with M subjects and K images per subject, we use one image as the reference image I_r (r = 1,...,R; $R \leq K$). We then define the training dataset as $T^r = [t^r_m]_{1\times M}$, where t^r_m is the feature vector of the reference image I_r for subject m (m = 1,...,M). The corresponding probe (testing) dataset can be defined as $P^r = [p^r_{m,k}]_{(K-1)\times M}$, where $p^r_{m,k}$ is the feature vector of the kth probe image, $k \neq r$ (i.e., k=1,...,$r-1, r+1$,...,K) for subject m (m = 1,...,M).

True positive means we correctly assign a face from the individual x to the same person x. Therefore, we define $Mat\_true^r = [dis(t^r_m, p^r_{m,k})]_{(K-1)\times M}$ to store the similarity measure between the feature vector of the rth reference image for subject m in the training dataset $T^r (m = 1, ..., M)$ and the feature vector of each image for the same subject m in the corresponding probe dataset P^r ($K - 1$ images per subject), where $dis(t^r_m, p^r_{m,k})$ is the l_2-norm-based distance between t^r_m and $p^r_{m,k}$.

False positive means we incorrectly assign a face from the individual x to the wrong person y ($x \neq y$). Therefore, we define $Mat\_false^r = [\overline{dis}(t^r_{\hat{m}}, p^r_{m,k})]_{(K-1)\times M}$ to store the similarity measure between the feature vectors of all reference images in the training dataset excluding the rth reference image for subject m and the feature vectors of the images for the same subject m in the corresponding probe dataset ($K - 1$ images per subject). Here, we use $\hat{m}$ to represent all subjects excluding subject m, where $\overline{dis}(t^r_{\hat{m}}, p^r_{m,k})$ is the average of the l_2-norm-based distances between $p^r_{m,k}$ and $t^r_{\hat{m}}$ (i.e., $\overline{dis}(t^r_{\hat{m}}, p^r_{m,k}) = \frac{\sum_{n=1;n\neq m}^{M} dis(t^r_n, p^r_{m,k})}{M-1}$).

Fig. 1 illustrates a graphical view of the notations and the computed distances. Green

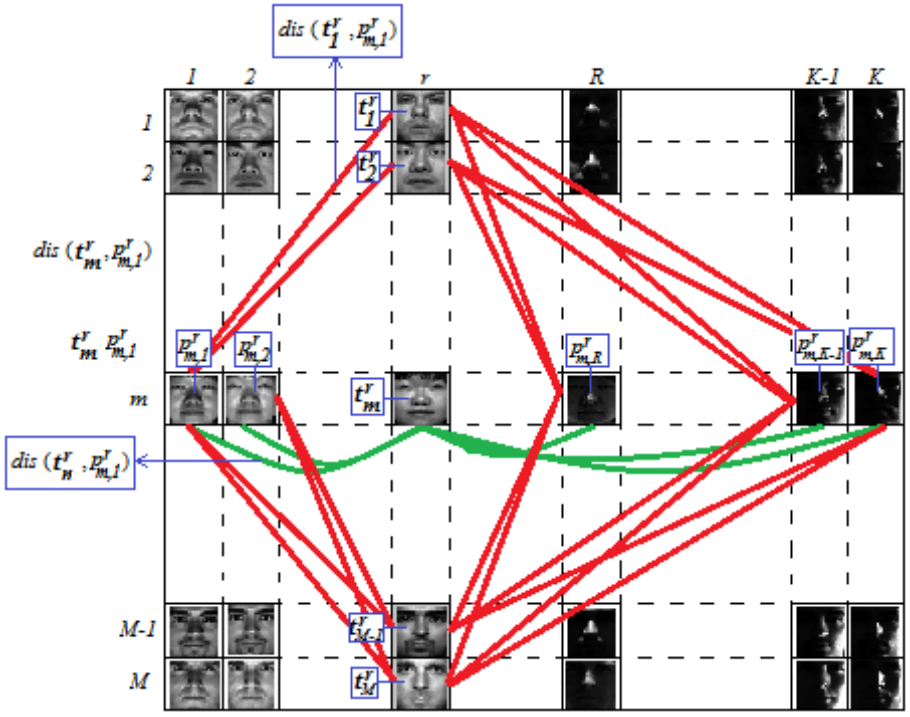

Fig. 1: Illustration of the process to compute scores and labels for ROC curves

curved connecting lines indicate the distance between probe images from one subject and their respective reference image. Red straight connecting lines indicate distances between probe images from one subject and the reference images from other subjects.

Now, we convert Mat_true^r and Mat_false^r to their respective 1D vectors with length $(K-1) \times M$. We call these 1D vectors $true\_scores^r$ and $false\_scores^r$, respectively. These score vectors can be concatenated to form the new vector S^r of the size of $2 \times (K-1) \times M$, which contains scores needed for plotting the ROC curve for the reference image I_r. The respective label vector of the same size of $2 \times (K-1) \times M$ can be simply obtained by assigning the label 1's to the scores corresponding to $true\_scores^r$ elements and assigning the label 0's to the scores corresponding to $false\_scores^r$ elements.

Next, we compute the TPR and the FPR for the obtained scores and their respective labels for the reference image I_r. To this end, we first sort the scores in a descending order and weigh them into two classes of weights based on their labels. In the first class (Weights1), if a score belongs to the false scores, we set its weight value as 1's and otherwise

Scores	Labels	Sorted Scores	Weights1	Weights2	Cumulative Weights1	Cumulative Weights2
17	1	20	1	0	1	0
14	1	18	1	0	2	0
13	1	17	0	1	2	1
10	1	16	1	0	3	1
11	1	15	1	0	4	1
18	0	14	0	1	4	2
15	0	13	0	1	4	3
12	0	12	1	0	5	3
20	0	11	0	1	5	4
16	0	10	0	1	5	5

TP	FN	TPR
1	4	0.2
2	3	0.4
2	3	0.4
3	2	0.6
4	1	0.8
4	1	0.8
4	1	0.8
5	0	1
5	0	1
5	0	1

FP	TN	FPR
0	5	0
0	5	0
1	4	0.2
1	4	0.2
1	4	0.2
2	3	0.4
3	2	0.6
3	2	0.6
4	1	0.8
5	0	1

Fig. 2**:** Illustration of the process to compute TPR and FPR from a set of ten random scores and their respective labels.

as 0's. In the second class (Weights2), if a score belongs to the true scores, we set its weight value as 0's and otherwise as 1's. Second, we compute cumulative weights for each class of weights. As a result, each cumulative weights class has the lowest value of 0 or 1 and the highest value of $(K-1)\times M$ (i.e., the total number of true or false labels). We then consider $2\times(K-1)\times M$ indices (i.e., the number of indices equals the number of values in both classes) to compute true positive (TP), false negative (FN), false positive (FP), and true negative (TN). In other words, the indices are integers ranging from 1 to $2\times(K-1)\times M$. For each index, the respective cumulative weight in class 1 (Weights1) indicates TP, and FN is also simply obtained by subtracting its corresponding TP value from the highest cumulative weight in class 1 (i,e., $(K-1)\times M$). Similarly, the respective cumulative weight

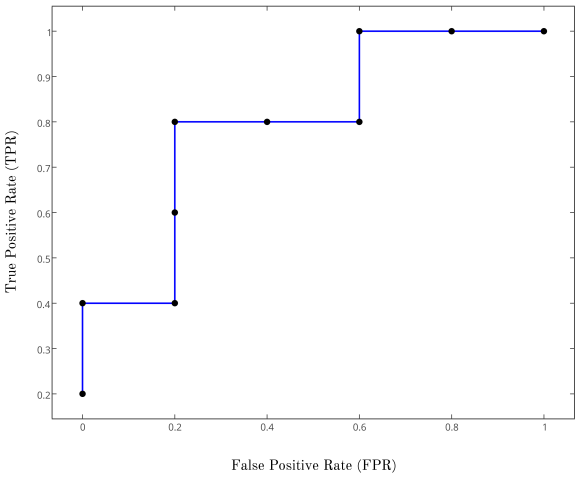

Fig. 3**:** ROC curve for the TPR and FPR values in Fig 2.

in class 2 (Weights2) indicates FP, and TN is obtained by subtracting its corresponding FP value from the highest cumulative weight in class 2 (i,e., $(K-1) \times M$).

Finally, for each index, TPR and FPR are simply computed as follows: TPR $= \frac{\text{TP}}{\text{TP+FN}}$ and FPR $= \frac{\text{FP}}{\text{FP+TN}}$. Fig. 2 illustrates the process to compute TPR and FPR for ten randomly generated scores with their labels. The top table contains the original scores and their labels together with their sorted scores, weights in two classes, and cumulative weights in two classes. The bottom left table presents computed TP, FN, and TPR. The bottom right table presents FP, TN, and FPR. The respective ROC curve for these TPR and FPR is shown in Fig. 3.

Following this procedure, we compute TPR and FPR values for all the reference images (i.e., r= 1,..., R). Finally, TPR and FPR for the database can be easily computed as the average of all TPR and FPR values across R experiments.

Vita

Mohammadreza Faraji

Published Journal Articles

- Face Recognition Under Illumination Variations Based on Eight Local Directional Patterns, Mohammadreza Faraji, Xiaojun Qi, *IET Biometrics*, vol. 4(1), pp. 10-17, 2015.
- Face recognition under varying illumination based on adaptive homomorphic eight local directional patterns, Mohammadreza Faraji, Xiaojun Qi, *IET Computer Vision*, DOI: 10.1049/iet-cvi.2014.0200.
- Face Recognition under Varying Illumination with Logarithmic Fractal Analysis, Mohammadreza Faraji, Xiaojun Qi, *IEEE Signal Processing Letters*, vol. 21(12), pp. 1457-1461, 2014.
- Interval type-2 fuzzy expert system for prediction of carbon monoxide concentration in mega-cities, M. H. Zarandi, Mohammadreza Faraji, M. Karbasian, *Applied Soft Computing*, vol. 12(1), pp. 291-301, 2012.
- An Exponential Cluster Validity Index for Fuzzy Clustering with Crisp and Fuzzy Data, M. H. Zarandi, Mohammadreza Faraji, M. Karbasian, *Scientia Iranica Transaction on Industrial Engineering*, vol. 17(2), pp. 95-110, 2010.

Published Conference Papers

- An effective neutrosophic set-based preprocessing method for face recognition, Mohammadreza Faraji, Xiaojun Qi, in *IEEE International Conference on Multimedia and Expo*, 2013.