Application of Adaptive Contrast Stretching Algorithm in Improving Face Recognition Under Varying Illumination Conditions

Chinedu God'swill Olebu* and Jide Julius Popoola
Department of Electrical and Electronics Engineering
Federal University of Technology, Akure
Ondo, Nigeria
*cgolebu@gmail.com

Date received: March 12, 2020
Revision accepted: August 10, 2020

Abstract

This work proposed a novel algorithm, adaptive contrast stretching algorithm (ACS), in improving face recognition under varying illumination conditions. The ACS algorithm, whose building blocks are tuned logarithm filter and anisotropic diffusion filter (ADF), was used to preprocess samples of face images obtained from the extended Yale face database B. The resulting preprocessed data was split into training and testing datasets. While the training dataset was used to train a deep convolutional neural network (DCNN), the testing dataset was subdivided into four subsets based on the azimuthal angle of illumination. In order to compare the recognition accuracy obtained from using the ACS algorithm, the face images in the training dataset were successively processed using discrete cosine transform, difference of Gaussian, weber faces, multi-scale retinex and single-scale retinex. The respective output images obtained from each technique were used to train the DCNN. The result obtained from each technique showed that the developed ACS algorithm significantly outperformed other algorithms used in this study with an accuracy of 95%. This value is 2.5% greater than the unimproved version of the ADF, which is currently one of the acclaimed techniques used by most computer vision researchers in the surveyed literature.

***Keywords**:* varying illumination, face recognition, recognition accuracy, adaptive contrast stretching, deep convolutional neural network

1. Introduction

Varying illumination is one of the limitations of face recognition technology. This is because in practical face recognition, the ambient conditions are not usually regulated. The implication is that a perfectly lit face image is not always guaranteed (Anila and Devarajan, 2012). Images of the same faces can appear differently due to the change in lighting conditions of its location. This is attributed to the fact that in such conditions, the inherent face image features such as edges are usually interpreted vaguely owing largely to differences in illumination which interfere with unique face features (Ramchandra and Kumar, 2013). To address this problem, a wide spectrum of algorithms have been introduced by researchers. According to Santamaria and Palacios (2005), most of these algorithms are relatively complex and to a large extent, lead to loss of important image information, reduction in the richness of face images and a more restricted mode of extracting illumination invariants. Thus, there is need to develop a robust and adaptive algorithm that will preserve the quality and richness of face images and improve the accuracy of face recognition under varying illumination conditions.

Several methods in reducing the effect of illumination variation in face recognition have been proposed. To name one, Aggarwal and Chellappa (2005) proposed an analysis-by-synthesis approach for face recognition under the condition of multiple light sources. The approach, which was based on the assessment of the hard non-linearity in the Lambert's law, is more realistic than a single light source. It was observed that the performance of the proposed algorithm did not require the knowledge of the number light sources. It was also observed that the proposed algorithm worked well even when recognizing faces illuminated by different number of light sources.

Similarly, Chen *et al.* (2006) proposed a novel method in normalizing illumination in the presence of illumination variation in face recognition technologies. The authors employed a discrete cosine transform (DCT) method to compensate for the illumination unevenness in the logarithm domain. A number of DCT coefficients were truncated in order to minimize variations under different lighting conditions, since variations in illumination lies in the low-frequency band of an image. The study outcome was further validated against Yale-B database and CMU-Pose, Illumination, and Expression (PIE) database. It turned out that the proposed technique succeeded in improving the performance of a real-time face recognition system. However, the truncation of the DCT coefficient limited the effectiveness of the technique due to significant reduction in the richness of the formed image.

Local Gabor exclusive XOR pattern (LGXP) is another algorithm that is used to encode the Gabor phase by utilizing the local XOR pattern (LXP) operator. This approach is a block-based Fisher's discriminator (BFLD) and was used to reduce the dimensionality of a proposed descriptor and enhance its discriminative power. The local patterns of Gabor magnitude and phase were finally used for face recognition. The method was evaluated using the face recognition technology (FERET) and the face recognition grand challenge (FRGC) 2.0 databases. After extensive performance evaluation and comparison of the method with other Gabor patterns, it was discovered that LGXP descriptor was very effective and outperforms most modern approaches (Xie *et al*., 2010). However, the major loophole in using this method was the increased dimensionality of the feature space.

Furthermore, Chunnian (2012) proposed a non-sampled contourlet transform method for face image enhancement. In the proposed method, the authors employed four different operations on the experimental face image. The face image was first normalized. A logarithmic transformation was then carried out on the face image to decompose the image into low frequency and high frequency sub-band components. This operation was followed by an adaptive normal shrink operation on both the low and high frequency components in other to eliminate any noise. Finally, a histogram equalization was then applied to the low frequency component with the aim to further reduce any remaining noise. The illumination invariant was eventually extracted by inverse non-sampled contourlet transform using the modified frequency components.

In the same vein, Zhou *et al.* (2013) proposed an adaptive scheme based on a fast bi-directional ensemble empirical mode decomposition (FBEEMD) and detail feature fusion. In the study, it was explained that FBEEMD is a fast feature of bi-directional, ensemble empirical mode decomposition (BEEMD) with a unique feature of low time-consuming surface interpolation and iteration computation decomposing an image into high-frequency sub-images that matches detailed feature and high frequency sub-images matching contour features. Two measures were proposed to calculate weights in quantifying the image feature. With this technique, an illumination-neural facial image was developed that helped to improve face recognition rate. These finding were verified by testing against Yale-B, PIE and FERET databases.

The study presented in Kang and Pan (2014) suggested a hybrid face recognition system that employed three image enhancement techniques. The first stage was histogram equalization (HE) – used to improve the overall contrast of the image. The logarithmic transform (LT) was then applied to enhance image details. Finally, the resulting image was filtered using a high pass filter to further recover the feature of the image. The authors' main motivation was the existence of low noise frequency featured in low-light regions of an image.

Based on Lambert reflectance model a new method was proposed by Zhuang *et al*. (2015) to tackle the issues concerning illumination variation in face recognition. A fast mean filter (FMF) was proposed to help repair the defects caused by the process of illumination invariants extraction. Non-linear normalization transformation was then used to increase the richness of the image information. The proposed method was compared with other state-of-the-art methods. It was established that the proposed method can extract more robust illumination invariants and also retain image information; thus, enhancing face recognition rates.

Yang *et al*. (2017) presented an illumination processing approach based on nonlinear dynamic range adjustment and gradient faces. In this method, the grayscale of face image was adjusted by nonlinear dynamic range adjustment using hyperbolic sine function in the logarithmic domain. The resulting gradient faces were then used to enhance the high frequency component of the face image and extract distinguishing facial features. Finally, these features were classified using principal component analysis (PCA).

Manhotra and Sharma (2017) formulated an illumination invariant face recognition algorithm based on the combination of gradient based illumination normalization and fusion of two illumination invariant descriptors. The ratio of the gradient amplitude and original image intensity provided the illumination invariant representation. Local binary pattern (LBP) and local ternary pattern (LTP) algorithms were used to extract the illumination invariants from the face image.

A Gabor phase based method was developed by Fan *et al*. (2017) in other to eliminate the effect of complex illumination. The authors used a set of two-dimensional (2-D) real Gabor wavelet with different directions to transform a normalized illumination on face images. The resulting multiple Gabor coefficients were then combined into a single representation extracting the illumination invariants.

Tran *et al*. (2017) proposed a method that combines both illumination pre-processing and singular value decomposition (SVD) to improve the efficiency of face recognition under varying illumination conditions. The training images were first preprocessed by an illumination preprocessing method and then

SVD was utilized to encode the pre-processed images. Following the application of LTP for feature extraction was illumination normalization. During the recognition phase, Chi-square method was used to check the dissimilarity measure between the probe images and the training set.

The above-mentioned chronological review from 2005 to 2017 shows various proposed to increase the efficiency of face recognition under uncontrolled illumination conditions. Similarly, the review shows some limitations of these each techniques. Although these techniques have succeeded in extracting the illumination invariants, they only retained little vital image information during preprocessing, thus, reducing recognition accuracy when applied to a face image recognition pipeline. Hence, it is important to develop an alternative algorithm that is not only immune to illumination variation but also has the ability to retain vital information of the face image, thereby improving overall face recognition accuracy under varying illumination conditions. To achieve this objective, this study was carried out with the aim of developing an adaptive contrast stretching (ACS) algorithm for preprocessing face images under varying illumination conditions.

2. Methodology

Divided into three stages, the modularized methodology used in this study is presented in Figure 1.

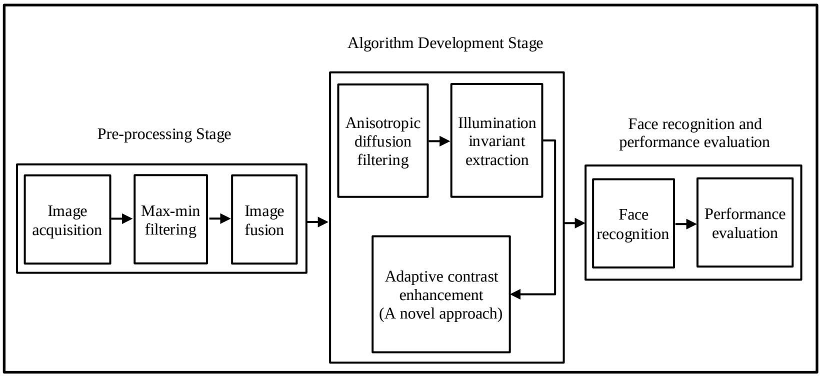

Algorithm Development Stage
Pre-processing Stage
Image acquisition
Max-min filtering
Image fusion
Anisotropic diffusion filtering
Illumination invariant extraction
Adaptive contrast enhancement (A novel approach)
Face recognition and performance evaluation
Face recognition
Performance evaluation

Figure 1. Modules of the methodology

2.1 Preprocessing Stage

2.1.1 Data Acquisition

This sub-stage focuses on data collection for the study. The first 20 characters of the extended Yale face database B, which comprised 1280 face images characterized by varying illumination, were retrieved. The retrieved 1280 face images formed the face image dataset for this study. The dataset was divided into two sets, namely training and testing sets. The training set contains 44 face images per character, summing up 880 face images for the said set. The testing set was first divided into subset 1, 2, 3 and 4 according to the azimuthal angle of illumination (denoted by P00A in the image file). Each character in subset 1 consisted of five face images having azimuthal angle in the range of -110 to -130. As an example, for the first character, face images contained in subset 1 are yaleB01_P00A-110E+40.pgm, yaleB01_P00A-110E+65.pgm, yaleB01_P00A-110E-20.pgm, yaleB01_P00 A-120E+40.pgm and yaleB01_P00A-110E+40.pgm. Subsets 2, 3, and 4 contained face images with azimuthal angle of illumination ranging from -25 to -35, +25 to +35, and +110 to +130, respectively. The overall steps involving the data preparation are shown in Figure 2. The samples of datasets per subset in each testing set from the extended Yale face database B are shown in Figure 3.

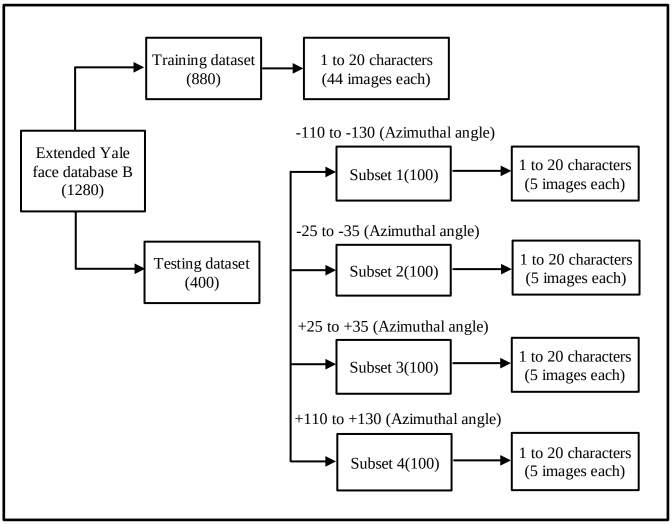
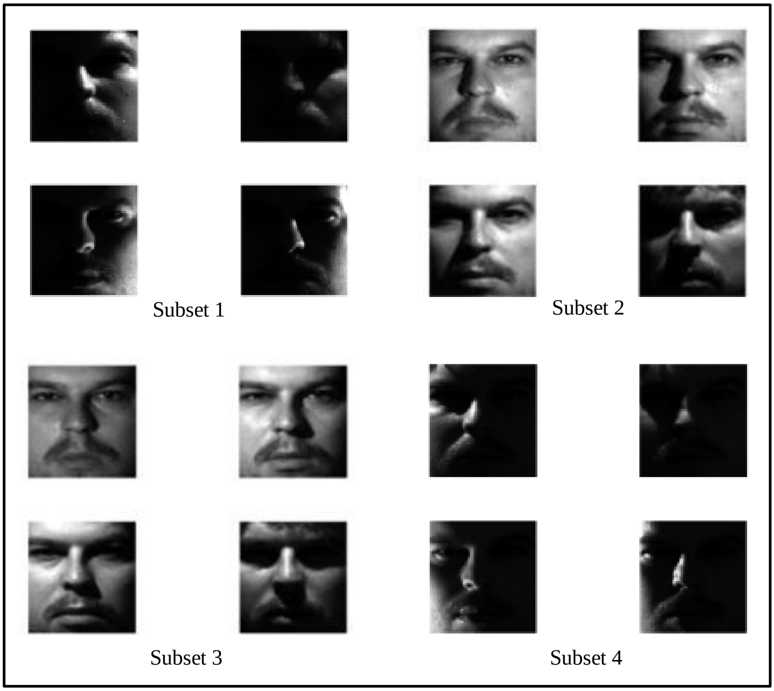

Training dataset (880)
1 to 20 characters (44 images each)
Extended Yale face database B (1280)
Testing dataset (400)
-110 to -130 (Azimuthal angle)
Subset 1(100)
1 to 20 characters (5 images each)
-25 to -35 (Azimuthal angle)
Subset 2(100)
1 to 20 characters (5 images each)
+25 to +35 (Azimuthal angle)
Subset 3(100)
1 to 20 characters (5 images each)
+110 to +130 (Azimuthal angle)
Subset 4(100)
1 to 20 characters (5 images each)

Figure 2. Stages of dataset acquisition



Subset 1

Subset 2

Subset 3

Subset 4

Figure 3. Samples of the testing datasets per subset

2.1.2 Max-min Filtering

After the image acquisition, the filtering phase was introduced using Lambert’s reflectance principle. By this principle, according to Unimap (2014), images are divided into two components – the high and the low frequency component. According to Chen *et al*. (2000), the Lambertian reflectance model is expressed mathematically in Equation 1.

$$I(x,y) = L(x,y) \times R(x,y) \tag{1}$$

where *L(x,y)* and *R(x,y)* represent the actual illumination and reflectance, respectively. *R(x,y)* is the high frequency component because it changes at a very fast rate, *L(x,y)* is the low frequency component because it changes at a very slow rate while *I(x,y)* is the actual face image. Under different image acquisition environment, as the essential surface feature of human face changes, *R* remains unchanged, while *L* changes slowly. This makes the reflectance *R* an illumination invariant. This assumption is not always true as even the high frequency component; *R* has its low frequency characteristics. Thus, illumination invariants was considered for both high and low frequency components. As a result, the local maximum and local minimum of the face image intensity, *I*, were computed and used to account for the high and low frequency components of the illumination. Equations 2a and 2b show the mathematical expression that was used for the *max* and *min* filters that were used to extract the neighborhoods local maximum and minimum of the face images pixel-by-pixel as provided in Verbeek *et al.* (1988).

$$L_{max} = \max_{(i,j)\in W} (I(x,y)|(x,y)\in W) \tag{2a}$$

$$L_{min} = \min_{(i,j)\in W} (I(x,y)|(x,y)\in W) \tag{2b}$$

where *W* is the neighborhood filter window. The *max* filter calculates the maximum of the pixels in the 3-by-3 neighborhood including the central pixel. The *min* filter calculates the minimum of the pixels in the 3-by-3 neighborhood including the central pixel. The 3-by-3 neighborhood topology was utilized in this study due to its simplicity.

2.1.3 Image Fusion

Image fusion is the third sub-stage of the preprocessing stage as shown in Figure 1. An illumination fusion operation was implemented on L_{max} and L_{min} image components by finding their average, pixel-by-pixel for each corresponding location. This was done by using the approach, proposed by Cheng *et al.* (2017), which enhanced the distinction of both the light shielding edges and other regions. The fused image, I_e, was obtained using Equations 3, 4 and 5 expressed mathematically by Cheng *et al.* (2017).

$$I_e = \frac{L_{max}(i,j) - I(i,j)}{L_{max}(i,j)} \tag{3}$$

$$t = mean(I_e) + 0.6\left(max(I_e) - mean(I_e)\right) \tag{4}$$

$$L_{max} = \begin{cases} L_{max}(x,y) & I_e(x,y) \geq t \\ L_{min}(x,y) & I_e(x,y) < t \end{cases} \tag{5}$$

2.2 Algorithm Development Stage

2.2.1 Anisotropic Diffusion Filtering

The anisotropic diffusion filter algorithm by Perona and Malik (1990) was adopted. The role of the filter was to facilitate a stronger relationship among neighbouring pixels of illumination and to better preserve image edge information. The mathematical expression of the anisotropic filter adopted from Perona and Malik (1990) is given in Equation 6.

$$L_{fused}^{t+1} = L_{fused}^{t} \; \frac{\lambda}{|\eta(x,y)|} \sum\nolimits_{p \in (x,y)} g_k\left(\left| \nabla L_{((x,y),p)} \right| \right) \nabla L_{((x,y),p)} \tag{6}$$

where t is the number of iterations; λ is a parameter used to measure the rate of diffusion of information across edges; $\eta(x,y)$ is the number of neighbors from all four directions (North, South, West, and East); and (x,y) denotes the pixel position in the discrete 2-D grid.

where ∇ used in Equation 6 is the continuous form gradient operator and g is the conduction coefficient. The symbol ∇ is a scalar quantity that measures the difference between neighboring pixels in each direction. According to Kamalaveni *et al*. (2015), the value of g was determined using Equation 7.

$$g = exp\left(-\left(\frac{||\nabla||}{K} \right)^2 \right) \tag{7}$$

The value of g was calculated for the four directions within the same neighborhood. In Equation 7, K is the gradient threshold parameter, whose value used in this study was based on the findings in Tsiotsios and Petrou (2013).

2.2.2 Illumination Invariant Extraction

In this sub-stage, the illumination invariant extraction was carried out based on the approach presented in Unimap (2014). This was obtained by making use of Lambert's reflectance model expressed mathematically in Equation 8.

$$R(x,y) = \frac{I(x,y)}{L_{fused}(x,y)} \tag{8}$$

where $I(x,y)$ is the original unprocessed face image and $L_{fused}(x,y)$ is the filtered fused image obtained from Equation 1.

2.2.3 Adaptive Contrast Enhancement

After extracting the illumination invariant, the resulting image passed through the developed adaptive contrast enhancement algorithm in order to enhance the edges of the invariant. The adaptive image contrast enhancement algorithm was built around the comparison of the entropy of the different enhanced images based on the amount of illumination, the azimuthal and elevation angle of illumination. The adaptive contrast enhancement algorithm developed is called the ACS algorithm.

Prior to determining the mean of the illumination invariant, *R* was first converted to a more precise double type, which also allowed for a reasonable scaling of the image. Following the conversion, the mean of all pixels in *R* was determined. According to Blanchet and Charbit (2014), this process is mathematically expressed in Equation 9.

$$R_{mean} = \frac{\sum_{i=1}^{i=N}\sum_{j=1}^{j=M} R(i,j)}{N \times M} \tag{9}$$

where *N* is the row length of *R*; *M* is its column length; $N \times M$ represents the size of *R*; and R_{mean} is the resulting mean image.

Two illumination scenarios were considered when developing the ACS algorithm. These were the uniform contrast improvement (UCI) and non-uniform contrast improvement (NCI). These two scenarios implemented contrast improvement based on the illumination characteristics of the face images, used in this study, provided by Gonzalez and Wood (2009). The relationship between the logarithm transformation of pixels and pixel intensities was exploited. The notion behind this method is that as pixels are logarithmically transformed, there is corresponding exponential increase in the intensities of pixels. However, rate of change of the exponential graph decreases with further decrease in the logarithm of the transformed pixel. The contrast of the illumination invariant obtained was enhanced for all *R* including the training and testing image data. The value of *R* with varying degree of illumination variation was contrast-improved. Its value assumes two forms; one with high value of illumination variation and another with low value of illumination variation. Each of the two classes of *R* has different illumination properties. The difference in the illumination properties was utilized to develop the ACS algorithm that helped in producing an optimally-normalized variant of *R*, which eventually aided the development of effective face recognition system. The algorithm below explains the various steps involved in developing the ACS algorithm for this study. In the algorithm (Figure 4), the variables with subscripts *M* was used in the UCI, while the variables with subscripts *N* were employed in the NCI scenario.

```
Input: $R_{marray} = \emptyset$; $R_{narray} = \emptyset$; $R^{total}_{uniform(M)} = \emptyset$; $R^{average}_{uniform(M)} = \emptyset$; $R^{sd}_{uniform(M)} = \emptyset$;
        $R^{average}_{nonuniform(N)} = \emptyset$; $R^{sd}_{nonuniform(N)} = \emptyset$;
Output: $M_{op}, N_{op}$
for $M$ = 0.5 to 8.0 step 0.5
    $R_{marray} \coloneqq R_{marray} + R_{entropym}$
    $R^{total}_{uniform(M)} \coloneqq R^{total}_{uniform(M)} + R^{total}_{uniform(M)}(M)$
    $R^{average}_{uniform(M)} \coloneqq R^{average}_{uniform(M)} + R^{average}_{uniform(M)}(M)$
    $R^{sd}_{uniform(M)} \coloneqq R^{sd}_{uniform(M)} + R^{sd}_{uniform(M)}(M)$
end
for $N$ = 0.2 to 2.0 step 0.2
    $R_{narray} \coloneqq R_{narray} + R_{entropyn}$
    $R^{average}_{nonuniform(N)} \coloneqq R^{average}_{nonuniform(N)} + R^{average}_{nonuniform(N)}(N)$
    $R^{sd}_{nonuniform(N)} \coloneqq R^{sd}_{nonuniform(N)} + R^{sd}_{nonuniform(N)}(N)$
end
                    $R^{min}_{uniform(M)} \leftarrow min(R^{sd}_{uniform(M)})$
                    $R^{min}_{nonuniform(N)} \leftarrow min(R^{sd}_{nonuniform(N)})$
                    $R^{max}_{uniform(M)} \leftarrow min(R^{average}_{uniform(M)})$
                    $R^{max}_{nonuniform(N)} \leftarrow min(R^{average}_{nonuniform(N)})$
//Optimal $M$ ($M_{op}$) and $N$ ($N_{op}$) are selected using lines 17-20
return $M_{op}$, $N_{op}$
```

Figure 4. The ACS algorithm

Two arrays, R_{marray} and $R_{narray,}$ are initialized an empty arrays. Then the 20th face image in the training set is selected. Two parameters, $R_{uniform}$ and $R_{nonuniform}$ are developed, such that each of them are adapted to normalize the different variants of R that are obtainable. Equations 10 and 11 mathematically define the value of $R_{uniform}$ and $R_{nonuniform}$, respectively.

$$R_{uniform} = \frac{1}{1 + \left(\frac{M}{R + eps}\right)^4} \tag{10}$$

$$R_{nonuniform} = \frac{1}{1 + \left(\frac{2.25}{(R + eps)^N}\right)} \tag{11}$$

where M in Equation 10 is an integer and varies between 0.5 and 8.0 in steps of 0.5 for uniform illumination. Similarly, N in Equation 11 is an integer that varies between 0.2 and 2.0 in the steps of 0.2 for non-uniform illumination; R is the illumination invariant and *eps* in both equations is epsilon, which is the distance of 1.0 to the next large double-precision number and has a numerical value of 2.2204×10^{-16} (Gonzalez and Wood, 2009).

The essence of determining the image entropy was to measure the degree of randomness of $R_{uniform}$. The entropy of $R_{uniform}$ was determined in order to establish a performance threshold, which was used in optimally selecting the best value for illumination invariant for the training and recognition phase. The degree of randomness $R_{uniform}$ and $R_{nonuniform}$ were computed using Equations 12a and 12b, respectively, given by Gonzalez *et al.* (2004).

$$R_{entropym} = ent(R_{uniform}) \tag{12a}$$

$$R_{entropyn} = ent(R_{nonuniform}) \tag{12b}$$

where *ent(.)* is a function that computes the entropy of an image. After the entropy measurement is the statistical analysis of the image. Hence, the total value of $R_{uniform}$ and $R_{nonuniform}$ for each column of M was computed for the total uniform and non-uniform components as $R^{total}_{uniform(M)}$ and $R^{total}_{nonuniform(N)}$, respectively. In addition, the standard deviation and average values of $R_{uniform}$ and $R_{nonuniform}$ were also computed as $R^{sd}_{uniform(M)}$, $R^{average}_{uniform(M)}$ and $R^{sd}_{nonuniform(N)}$, $R^{average}_{nonuniform(N)}$, respectively for each column value of M. The minimum value of $R^{sd}_{uniform(M)}$ and $R^{sd}_{nonuniform(N)}$ were then respectively determined using Equations 13a and 13b.

$$R^{min}_{uniform(M)}=min(R^{sd}_{uniform(M)}) \tag{13a}$$

$$R^{min}_{nonuniform(N)}=min(R^{sd}_{nonuniform(N)}) \tag{13b}$$

where *min(.)* computes the minimum of the arguments provided which shows the deviation of $R^{sd}_{uniform(M)}$ from the average value $R^{average}_{uniform(M)}$. Similarly, the maximum value of $R^{average}_{uniform(M)}$ and $R^{average}_{nonuniform(N)}$ were computed using Equations 14a and 14b.

$$R^{max}_{uniform(M)}=max(R^{average}_{uniform(M)}) \tag{14a}$$

$$R^{max}_{nonuniform(N)}=max(R^{average}_{nonuniform(N)}) \tag{14b}$$

where the function *max(.)* computes the maximum value of the argument provided. The optimal value of M, M_{op} was then selected from $R^{sd}_{uniform(M)}$ and $R^{average}_{uniform(M)}$. A value of M was chosen at the instance where $R^{max}_{uniform}$ is maximum and $R^{min}_{uniform}$ is minimum. The same process was repeated in selecting the optimal value of N, N_{op}. This led to the computation of $R^{max}_{nonuniform}$ and $R^{min}_{nonuniform}$. The corresponding images for the illumination invariant for both uniform and non-uniform cases were computed. Equations 15 and 16 are mathematical expressions that convert the optimal values of M_{op} and N_{op} to images.

$$R_{uniform(M_{op})} = \frac{1}{1+\left(\frac{M_{op}}{R+eps}\right)^{4}} \tag{15}$$

$$R_{nonuniform(N_{op})} = \frac{1}{1+\left(\frac{2.25}{(R+eps)^{N_{op}}}\right)} \tag{16}$$

The average value of $R_{uniform(M_{op})}$, $R^{average}_{uniform(M_{op})}$ and $R_{nonuniform(N_{op})}$, $R^{average}_{nonuniform(M_{op})}$ were also determined. Finally, either of $R_{uniform(M_{op})}$ or $R_{nonuniform(N_{op})}$ was selected based on the highest value of either of $R^{average}_{uniform(M_{op})}$ or $R^{average}_{nonuniform(M_{op})}$, which is mathematically described in Equation 17.

$$R_{selected} = \begin{cases} R_{uniform(M_{op})}, & \frac{R^{average}_{uniform(M_{op})}}{R^{average}_{nonuniform(N_{op})}} > 1 \\ R_{nonuniform(N_{op})}, & \frac{R^{average}_{uniform(M_{op})}}{R^{average}_{nonuniform(N_{op})}} < 1 \end{cases} \tag{17}$$

2.3 Face Recognition and Performance Evaluation Stage

This is the last phase of the study as shown in Figure 1. In this subsection, a deep convolutional neural network was designed and trained to recognize the processed images obtained using the various image processing techniques used that formed the foundation of this study. Similarly, a deep convolutional neural network was designed and trained to recognize the processed images obtained using the developed ACS algorithm. The interactive MATLAB® deep learning toolbox (Mathworks, 2017) was used to implement this module with a learning rate of 0.0001 and a maximum epoch of 10. It is also worthy to note that the specifications of the Windows 8 machine used for the implementation are 64-bit operating system, installation memory of 4.00 gigabyte and processor designation of Intel® Celeron® CPU N3050 with a speed of 1.60 GHz. After the training, the face recognition accuracies of the dataset was obtained by testing the network using the testing sets per subset. The activities in this module are presented in the succeeding sections.

2.3.1 Train a DCNN

In this sub-stage, the applicability of deep convolutional neural network, also known as deep learning for the face image classification problem considered in this work was evaluated. The deep learning architecture utilized in this study consisted image input layer, 2-D convolutional layer, rectified linear unit (ReLU), max-pooling layer, fully connected layer, soft-max layer and classification layer. Figure 5 illustrates the architecture adopted in this study. Brief information on each layer of the DCNN architecture employed is shown below.

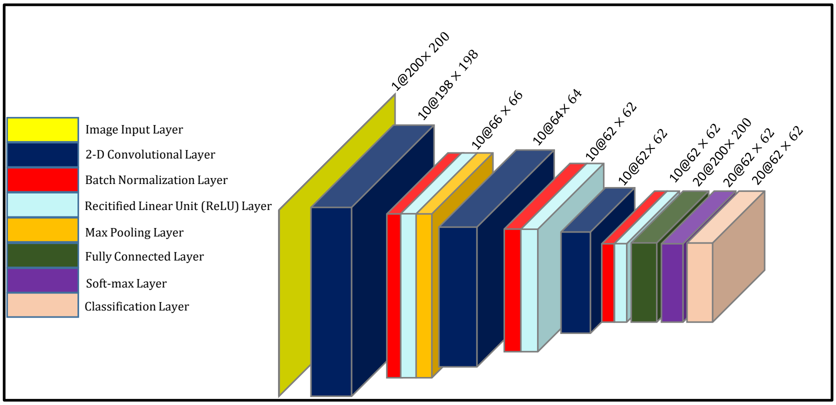


Figure 5. The utilized DCNN architecture

The first layer of the DCNN was the image input layer. In this layer, the image data were acquired and the above previous operations were implemented to produce the image information that would be processed. The 80% of the images were used in training the neural network while the remaining 20% were used in classification. The sizes of the image used in this layer were consistent in order to level hyper-parameters when going deep down the DCNN layers. The size of the final processed face image after passing through the ACS algorithm was 200 × 200. This image was then passed to the convolutional layer for further processing.

In the 2-D convolutional layer, a mask of size 2 × 2 was used (Havaei *et al.*, 2017). Using the mask, the convolutional operation was implemented by adding the multiplication of each element of the mask mapped to the corresponding elements in the local neighbourhood as described earlier. A total of 10 filters of size 3 × 3 with randomly generated kernel weights were used in the same region of inputs.

The ReLU layer serves as an activation of the output of the convolutional layer. In this layer, each element in the output of the convolutional layer were replaced by the maximum of '0' and the value of the element – that is, all negative pixel values are replaced with '0' and positive pixel values are retained (Nair and Hinton, 2010). Mathematically, the function of the ReLU Layer is represented in Equation 18.

$$R_{ReLU}(i,j)= max(0,R_{con}(i,j)) \tag{18}$$

where $R_{con}(i,j)$ is the output pixel value of the convolutional layer and $R_{ReLU}(i,j)$ is the output value after applying the ReLU filter.

As mentioned previously, the max-pooling layer further reduces the dimension of the image layer by finding the maximum of all the element within the $N \times N$ local neighbourhood. This layer carries out a non-linear down sampling operation after the convolutional layer is passed through the ReLU activation function (Mathworks, 2017). In this work, a filter of size 3 × 3 with a stride of three was chosen for the max-pooling layer. Equation 19 defines mathematically the max-pooling operation applied.

$$R_{mp}=max(pixel\ elements\ in\ a\ neighborhood) \tag{19}$$

where R_{mp} is the corresponding output and $max(.)$ is a function that computes the maximum value of pixel elements in a neighbourhood.

The fully-connected layer (FCL) output a column vector of k dimensions where k is the number of possible classes predictable by the network. This vector contains the probabilities for each class of any image being classified. In this study, all part of the neurons were interconnected to form the single vector that was be used in predicting the trained network.

Following the FCL is the soft-max layer. The soft-max layer provides the soft-max activation function for a multi-class classification problem. The soft-max activation that was used in the study is defined by Bishop (2006) and is expressed in Equations 20 and 21.

$$\mathrm{p}(C_r|x)= \frac{p(x|C_r)p(C_r)}{\sum_{j=1}^{k} p(x,C_j)p(C_j)} = \frac{exp(a_r)}{\sum_{j=1}^{k} exp(a_j)} \tag{20}$$

$$a_r=In(p(x|C_r)p(C_r) \tag{21}$$

where $p(C_r\,|x) = 1$ and $p(C_j\,|x) = 0$. $p(x\,|C_r)$ is the conditional probability of the sample given class r, and $p(C_r)$ is the class prior probability.

The final layer is the classification layer. This layer uses the probabilities returned by the soft-max activation function for assignment to one of the mutually exclusive classes.

3. Results and Discussion

3.1 Recognition Accuracies using Different Algorithms

This subsection presents the result of the recognition accuracies of the seven algorithms developed as shown in Figure 6. The figure illustrates the recognition accuracies obtained after processing the datasets retrieved from the extended Yale face database B. The result of the preprocessing accuracies shows that the raw image data performed poorly with an accuracy value ranging from 10 to 57%. On the contrary, the anisotropic diffusion filter (ADF) algorithm performed satisfactorily well with accuracies that are above 90% for subsets 2, 3 and 4.

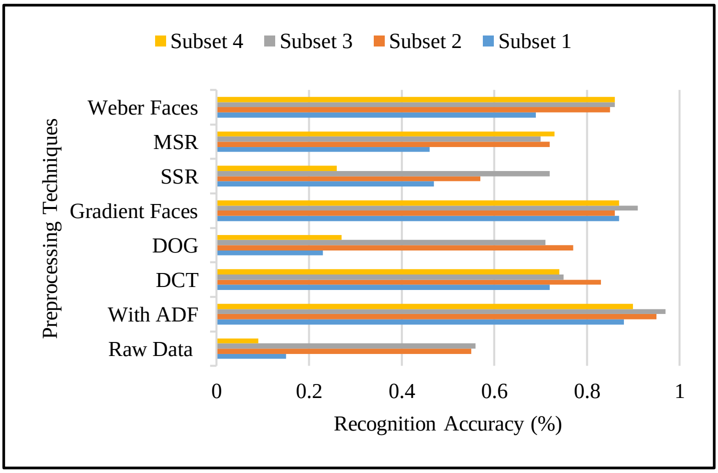


Figure 6. Recognition accuracies using different techniques

However, the percentage recognition for subset 1 is less than 90%, which is comparable with the recognition rate of the gradient faces algorithm on the same subset. All other algorithms performed relatively low on all the image subsets when compared with ADF. This result buttresses the finding of Animasahun and Popoola (2015) stating that adopting the application of appropriate preprocessing technique usually enhances the recognition potential of the face recognition pipeline.

3.2 Image Entropy for UCI Scenario

In the developed ACS algorithm, face information of the training data were obtained and represented in terms of the image entropies. Figures 7 illustrates the relationship between the average entropy and *M*-values and the standard deviation of entropies and *M*-values specifically for the uniform contrast improvement scenario which forms a huge part of the ACS algorithm. Figure 7a shows a steep increase in the average entropy (which measures the randomness of the features in the image data) as the *M*-values increase and thereafter, a gradual depreciation in the average entropy with respect to the *M*-values. Conversely, Figure 7b initially shows a gradual decay of the standard deviation of the entropies and later, a gradual decay of the standard deviation of the entropy with an increase in *M*-value.

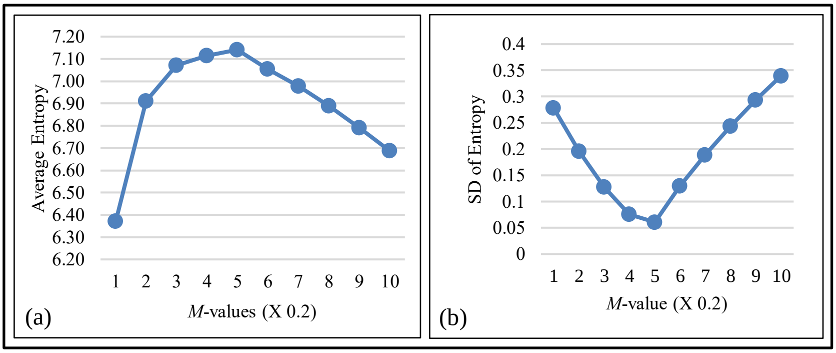


Figure 7. Variation of average entropies (a) with M Variation of the standard deviation of entropies (b) for each value of M

In Figure 7a, when the *M*-value is 1, the entropies of the image samples recorded a greater value of 7.1 which is 1.1 times greater than the lowest scoring value of *M*. When compared with the Figure 7b of the same *M* value considered previously, the entropies for the training sets are relatively consistent with negligible variations – evident in the value of the lower value of the standard deviation of the entropy when *M* is 1. However, the standard deviation of the entropy at values of *M* other than 1 is seen to increase linearly. This behavior implies that a single representation can be used to compute similar $R_{uniform}$ of each image sample when the M-value is set at an optimal value of M in the case of the UCI scenario.

3.3 Image Entropy for NCI Scenario

For the NCI scenario, the average entropy values obtained were similar to those obtained in the UCI. However, the standard deviation of the entropies exhibited a strange variation in the values obtained. Figure 8 illustrates both the average entropy and its standard deviation.

It can be observed in Figure 8a that the best average entropy was achieved at an M-value of 1. This implies that more information can be obtained from the face image at values of M. Besides, an M-value of 0.8 exhibited the same average entropy value as when M is 1. In comparison, the standard deviation of the entropy at an M-value of 0.8 is greater than that obtained when M-value is 1. Hence, an M-value of 1 is still the optimal value. Arguably some other values of M may exhibit lower variation in entropies. However, they exhibit lower average values. The variation of the image entropy of the image dataset with the value of M further supports the claims of Sabuncu (2006) that image entropy varies closely with the quality of the image.

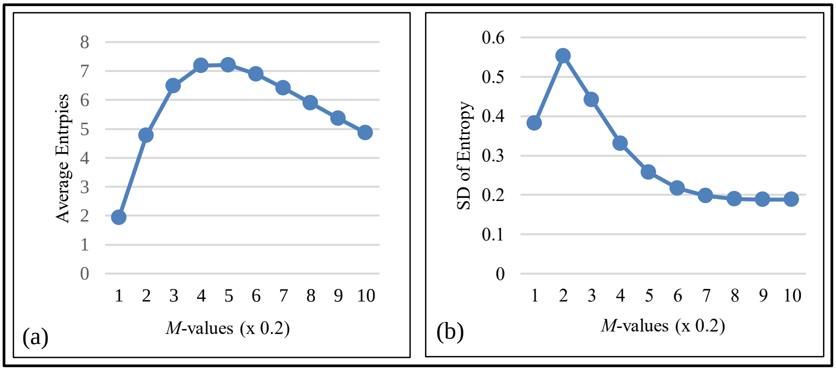


Figure 8. Variation of average entropies (a) with M variation of the standard deviation of entropies (b) for each value of M

3.4 Recognition Accuracies after Implementing the Full ACS Algorithm

In order to validate the hypothesis in the previous subsections where the UCI and NCI scenarios were considered in the developed ACS algorithms, a parametric sweep was carried out for each M-value on the UCI and NCI algorithm in each subset. Different recognition accuracies were obtained from different *M*-values. The variation of the recognition accuracies for each *M*-values are depicted in Figure 9.

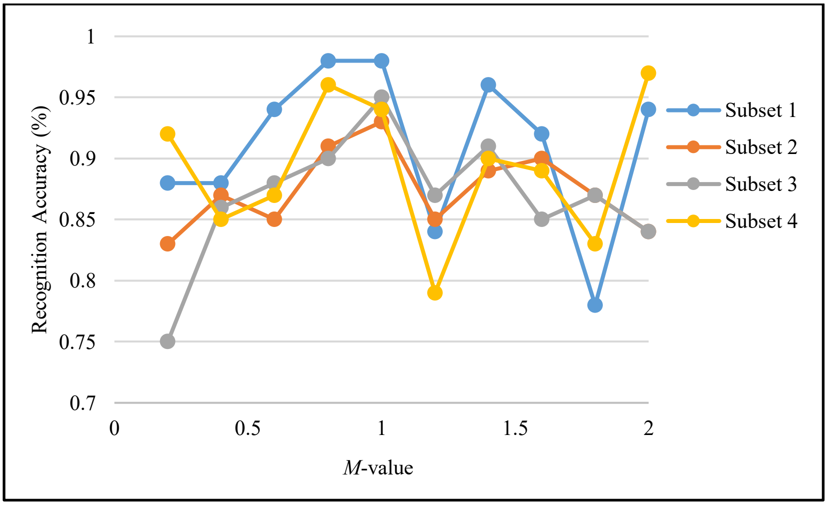


Figure 9. Determining the optimal value *M*-Value for maximum recognition accuracy

The obtained recognition accuracy for face images in subset 1 is 98% for both *M*-values of 0.8 and 1.0. Similarly, for the same *M*-value, the recognition accuracy for subsets 2 and 4 seems to overlap at an accuracy of 94%. Subset 3 attained an accuracy of 95%. This implies that at an *M*-value of 1, optimal performance for all subsets were achieved. This further validates the principle established using the previously outlined UCI and NCI scenarios in the developed ACS algorithm.

3.5 Experimental Analysis

Since the optimal *M*-value was chosen for both the UCI and NCI scenarios, then an experiment was done in comparison with other preprocessing algorithms used in the subsection 2. Figure 10 shows the recognition accuracies obtained from each preprocessing technique including the ACS algorithm with optimal value of *M*. Also, the pictorial representation of the some face image samples preprocessed using the developed ACS algorithm is shown in Figure 11.

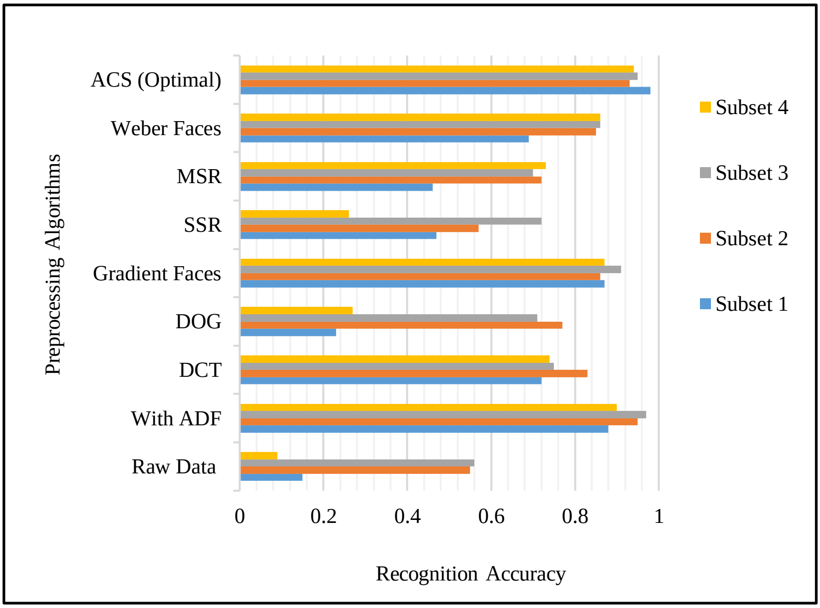


Figure 10. Recognition accuracies using other techniques and the developed ACS algorithm

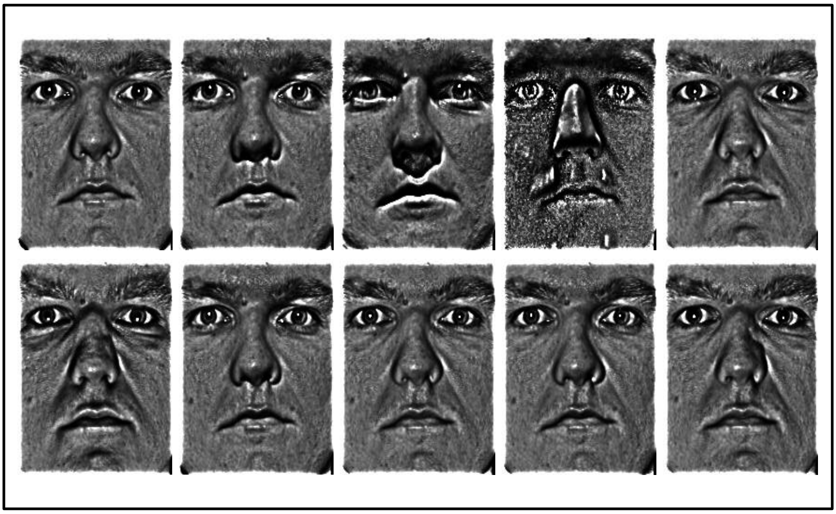


Figure 11. Some face samples obtained after implementing the ACS algorithm

The optimal ACS algorithm yielded a recognition accuracy in subset 1 that is greater than the corresponding accuracy of ADF for the same subset by a factor of 10% (Figure 10). A difference of 2% was attained for subsets 1 and 2 using the ADF and ACS techniques. For subset 4, the optimal ACS algorithm demonstrated a recognition accuracy that is 4% greater than the recognition accuracy of the ADF technique. Furthermore, the average recognition accuracy using the ACS algorithm was 2.5% greater than the recognition accuracy obtained when the ADF technique was used. This fact is further buttressed by Figure 11 which shows the face image samples extracted from the extended Yale face database B whose illumination variation has been significantly normalized. Overall, the developed ACS algorithm offered a better performance among other state-of-the-art algorithms considered in the literature.

4. Conclusion and Recommendation

In this study, a new technique called the ACS was developed and implemented in addressing the problem on varying illumination in face recognition systems. The extended Yale face database B was used to validate the developed ACS algorithm. In comparison with other state-of-the-art techniques, the ACS algorithm performed satisfactorily in preprocessing the face samples obtained from the database. This was evident when a DCNN pipeline was employed to measure the accuracy of recognizing face images for different subset classifications in the dataset obtained from the extended Yale face database B. It was found out that the ACS algorithm to a large extent outperformed other algorithms considered in this study with an accuracy ranging from 94 to 98%. However, the execution time of the algorithm was unideal for real-time deployment in face recognition systems. Hence, future work should done to improve the overall implementation speed of the algorithm, which could engender its application in real-time face recognition systems.

5. References

Aggarwal, G., & Chellappa, R. (2005). Face recognition in the presence of multiple illumination sources. Proceedings of the IEEE International Conference on Computer Vision, Beijing, China, 2, 1169-1176.

Anila, S., & Devarajan, N. (2012). Preprocessing technique for face recognition applications under varying illumination conditions. Global Journal of Computer Science and Technology, Graphics & Vision, 12(11), 13-18.

Animasahun, I.O., & Popoola, J.J. (2015). Application of mel frequency ceptrum coefficients and dynamic time warping for developing an isolated speech recognition system. International Journal of Science and Technology, 4(1), 1-8.

Bishop, C.M. (2006). Pattern recognition and machine learning (1st Ed.). New York, USA: Springer-Verlag.

Blanchet, G., & Charbit, M. (2014). Digital signal and image processing using MATLAB®: Fundamentals (2nd Ed.). Hoboken, New Jersey, US: John Wiley & Sons Inc.

Chen, H.F., Belhumeur, P.N., & Jacobs, D.W. (2000). In search of illumination invariants. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Hilton Head Island, USA, 2, 1-8.

Chen, W., Er, M.J., & Wu, S. (2006). Illumination compensation and normalization for robust face recognition using discrete cosine transform in logarithm domain. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, 36(2), 458-466. https://doi.org/10.1109/TSMCB.2005.857353

Cheng, Y., Li, Z., & Han, Y. (2017). A novel illumination estimation for face recognition under complex illumination conditions. IEICE Transactions on Information and Systems, E100-D, 4, 923-926. https://doi.org/10.1587/transinf.2016edl8218

Chunnian, F. (2012). Nonsubsampled contourlet transform based illumination invariant extracting method. International Journal on Advances in Information Sciences and Service Sciences, 4(17), 47-55. https://doi.org/10.4156/AISS.VOL4.ISSUE17.5

Fan, C., Wang, S., & Zhang, H. (2017). Efficient Gabor phase based illumination invariant for face recognition. Advances in Multimedia, 1-11. https://doi.org/10.1155/2017/1356385

Gonzalez, R., & Woods, R. (2009). Digital image processing (3rd Ed.). New Jersey, USA: Pearson Education International.

Gonzalez, R., Woods, R., & Eddins, S. (2004). Digital image processing using MATLAB (3rd Ed.). NJ, USA: Pearson Education Inc.

Havaei, M., Davy, A., Warde-farley, D., Biard, A., Courville, A., Bengio, Y., & Larochelle, H. (2017). Brain tumor segmentation with deep neural networks. Medical Image Analysis, 35, 18-31. https://doi.org/10.1016/j.media.2016.05.004

Kamalaveni, V., Rajalakshmi, R.A., & Narayanankutty, K.A. (2015). Image denoising using variations of Perona-Malik model with different edge stopping functions. Procedia Computer Science, 58, 673-682. https://doi.org/10.1016/j.procs.2 015.08.087

Kang, Y., & Pan, W. (2014). A novel approach of low-light image denoising for face recognition. Advances in Mechanical Engineering, 1-13. http://dx.doi.org/10.1155/2014/256790

Manhotra, S., & Sharma, R. (2017). Face recognition under varying illuminations using local binary pattern and local ternary pattern fusion. International Journal of Computational Engineering Research, 7(7), 69-77.

Mathworks. (2017). Introducing deep learning with MATLAB. Retrieved from https://it.unt.edu/sites/default/files/deep_learning_ebook.pdf.

Nair, V., & Hinton, G. (2010). Rectified linear units improve restricted Boltzmann machines. Proceedings of the 27th International Conference on Machine Learning, Haifa, Israel, 807-814.

Perona, P., & Malik, J. (1990). Scale-space and edge detection using anisotropic diffusion. IEEE Transactions on Pattern Analysis and Machine Intelligence, 12(7), 629-639.

Ramchandra, A., & Kumar, R. (2013). Overview of face recognition system challenges. International Journal of Scientific & Technology Research, 2(8), 234-236.

Sabuncu, M. (2006). Entropy-based image registration (Dissertation). Princeton University, New Jersey, United States.

Santamaria, M.V., & Palacios, R.P. (2005). Comparison of illumination normalization methods for face recognition. Retrieved from https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.183.6407&rep=rep1&type=pdf

Tran, C.K., Tseng, C.D., Shieh, C.S., & Lee, T.F. (2017). Face recognition under varying illumination conditions: Improving the recognition accuracy for local ternary patterns based on illumination normalization methods and singular value decomposition. Journal of Information Hiding and Multimedia Signal Processing, 8(4), 957-966.

Tsiotsios, C., & Petrou, M. (2013). On the choice of the parameters for anisotropic diffusion in image processing. Pattern Recognition, 46(5), 1369-1381. https://doi.org/10.1016/j.patcog.2012.11.012

Unimap. (2014). Image processing in frequency domain. Retrieved from https://portal.unimap.edu.my/portal/page.

Verbeek, P.W., Vrooman, H.A., & Van Vliet, L.J. (1988). Low-level image processing by max-min filters. Signal Processing, 15(3), 249-258. https://doi.org/10.1016/0165-1684(88)90015-1

Xie, S., Shan, S., Chen, X., Member, S., & Chen, J. (2010). Fusing local patterns of Gabor magnitude and phase for face recognition. IEEE Transactions on Image Processing, 19(5), 1349-1361. https://doi.org/10.1109/TIP.2010.2041397

Yang, Z.J., Nie, X.F., Xue, H., & Xiong, W.Y. (2017). Face illumination processing using nonlinear dynamic range adjustment and gradientfaces. In: Yang, T., Fakharian, A. (Eds.), Proceedings of the 2nd Annual International Conference on Electronics, Electrical Engineering and Information Science, Xi'an, Shaanxi, China, 117, 202-208.

Zhou, Y., Zhou, S., Zhong, Z., & Li, H. (2013). A de-illumination scheme for face recognition based on fast decomposition and detail feature fusion. Optics Express, 21(9), 11294. https://doi.org/10.1364/OE.21.011294

Zhuang, L., Chan, T.H., Yang, A.Y., Sastry, S.S., & Ma, Y. (2015). Sparse illumination learning and transfer for single-sample face recognition with image corruption and misalignment. International Journal of Computer Vision, 114(2-3), 272-287. https://doi.org/10.1007/s11263-014-0749-x