Research Journal of

Information Technology

ISSN 1815-7432

Academic Journals Inc.

www.academicjournals.com

Research Journal of Information Technology 7 (1): 22-41, 2015
ISSN 1815-7432 / DOI: 10.3923/rjit.2015.22.41
© 2015 Academic Journals Inc.

Comparative Analysis of Illumination Compensation Techniques in Face Recognition

[1]M. Jasmine Pemeena Priyadarsini, [2]K. Murugesan, [2]Srinivasa Rao Inbathini and [2]V. Harish Goutham
[1]School of Electronics Engineering, VIT University, India
[2]Sree Sastha Institute of Engineering and Technology Chennai, India

Corresponding Author: M. Jasmine Pemeena Priyadarsini, School of Electronics Engineering, VIT University, India

ABSTRACT

This study was designed to use Histogram Equalization and SVM based method to build a face recognition system under variant pose and illumination condition. Illumination compensation plays a vital role in face recognition. If the system can detect illumination variant faces, it increases the efficiency of the system. To compare the latest illumination compensation algorithm, Oriented Local Histogram Equalization (OLHE) which has proven to have exceptionally high performance under extreme lighting conditions with the previous state of the art algorithms such as Local Binary Pattern (LBP) and Local gradient Oriented Binary Patterns (LGOBP) as they encode micro-patterns giving the efficient descriptors for face recognition. It proposed that the illumination compensation algorithm OLHE will give the highest efficiency rates among the three algorithms taken into consideration and is the best illumination compensation algorithm available by analyzing their performance on the datasets CMU-PIE and extended YALE B.

Key words: LBP, LGOBP, OLHE and SVM

INTRODUCTION

The problem of face recognition has been one of the most prominent areas of machine vision for about a decade. Current systems have advanced to be fairly accurate in recognition under constrained scenarios but extrinsic imaging parameters such as pose, illumination and facial expression still cause much difficulty in correct recognition. In day-to day life face recognition system for still image and video have been in limelight both for the commercial purpose and security reasons. Due to increased importance of security in recent years face recognition has received substantial attention from researches in biometrics, pattern recognition field and computer vision communities. Various techniques have been developed for entrance control in building, personal access of computers, or for the criminal investigation.

With increase in demand of security it has added a cost factor for face recognition system which makes it difficult to reach to every man. The face recognition systems can extract the features of face and compare this with the prepared database.

Naturally people are good at face recognition through brain and nerve cells. It is very difficult to understand how our brain function stimulate and recognize the faces. Its altogether a different approach that the faces are recognize. Human faces have been in research for more than twenty years in which many new techniques have developed.

A face recognition system Hadid *et al.* (2004), is a software application to identify or verify a person's image from still image or a video sequence. Unfortunately developing a computational model for face recognition from video sequences is quite difficult due to involvement of complex face under different pose and illumination condition.

Approaches in face recognition according to Corcoran and Costache (2005), is such a challenging yet interesting problem that it has attracted researchers who have different backgrounds: Psychology, pattern recognition, neural networks, computer vision and computer graphics. It is due to this fact that the literature on face recognition is vast and diverse. To have a clear and high-level categorization, Apsychological study guideline is suggest to show that how humans use holistic and local features (Lizama *et al.*, 1997). Specifically, Holistic matching methods, Feature-based (structural) matching methods and Hybrid methods.

Holistic method use the whole face region as the raw input to a recognition system. One of the most widely used representations of the face region is Feature-based (structural) matching methods. In Hybrid methods, local features such as the eyes, nose and mouth are first extracted and their locations and local statistics (geometric and/or appearance) are fed into a structural classifier.

Just as the human perception system uses both local features and the whole face region to recognize a face, a machine recognition system should use both. Using Principal-Component Analysis (PCA) (Priyadarsini *et al.*, 2012) many face recognition techniques have been developed eigen faces which use a nearest neighbor classifier, feature-line-based methods which replace the point-to-point distance with the distance between a point and the feature line linking two stored sample points. Utilizing higher order statistics, Independent-Component Analysis (ICA) is argued to have more representative power than PCA and hence may provide better recognition performance than PCA. Most earlier methods belong to the category of structural matching methods, using the width of the head, the distances between the eyes and from the eyes to the mouth, etc., angles between eye corners, mouthextrema, nostrils and chin top. Without finding the exact locations of facial features, Hidden Markov Model (HMM) based methods use strips of pixels. In the hybrid method category, a brief review the modular eigenface method a hybrid representation based on PCA and Local Feature Analysis (LFA).

To apply various feature extraction and classification algorithms in public database such as Extended YALE B and CMU-PIE. To evaluate the best illumination compensation method to be adopted in real time environment. To study the performance metrics of the algorithm of illumination and pose variant images.

To use feature extraction and classification algorithms (LBP, LGOBP and OLHE) and SVM based method to build a face recognition system under variant pose and illumination condition Shan *et al.* (2003). To build a real time training and testing database. To vary the facial pose in training database as well as testing database. To vary amount of noise, little bit illumination. To observe Recognition Rate accuracy, False Rejection Rate and False Acceptance Rate based on above variations.

To compensate the illumination variation in faces, To describe LBP which is one of the best performing texture descriptors, to describe LGOBP which is based on the LBP and to analyze OLHE technique which is proposed to perform the two algorithms.

MATERIALS AND METHODS

The vision measure the amount of received light in the form of independent pixels as the result of introduction among various objects and light source; the value measured is the brightness. The measurement of the flow and transfer of radiant energy is called as radiometry. The appropriate tool to know the relation between the flow and transfer of energy is image creation. The radiometric approach to know about brightness is avoided normally because of its complexity and instability. The brightness measured does not provide a precise quantitative measurement. To have a precise quantitative measurement can be achieved by task specific illumination that allows the location of objects of interest on a qualitative level. Photometry is a branch of science, studies the sensation of radiant light energy in human eye, both disciplines describe similar phenomena using similar quantities. The basic radiometric quantity is radiant flux and its photometric counterpart is luminous flux.

The field of biometrics examines the physical or behavioral traits that can be used to determine a persons identity. Facial recognition is the part that everyone prefers because no physical interaction and perfect identification. The face images are collected such as CMU-PIE and EXTENDED YALE B, under different illumination and pose with distance and environment conditions. Both the type of image is taken to the next step, feature extraction. The various feature extraction techniques are LBP, LGOBP and OLHE. These feature extracted image from different algorithms are given as an input to the different types of classification algorithms. By taking a image from different feature algorithms to the different classification algorithms performance for face recognition. A novel formation for the more accurate and quick recognition in all invert images and for different camera captured images are explained in the results discussion.

Local binary patterns: Local Binary pattern is the best texture descriptor that has been commercially used over the years for many applications. It was introduced by Ahonen *et al.* (2006). LBP has computational efficiency and it is invariant to monotonic gray level changes.

LBP operator: LBP characterizes the local image texture by the sholding the 3×3 neighborhood for each pixel with center pixel value. As a result a binary number is obtained it can assign the neighborhood of the center pixel by a set of sampling points with any radius as done by Ahonen *et al.* (2006) and Hadid *et al.* (2004). If the sampling point does not fall in the center of a pixel, bilinear interpolation can be used:

$$LBP_{p,r} = \sum_{p=0}^{p-1} s(g_i - g_c)2^i,\ S(x) = \begin{cases} 1 \text{ if } x \geq 0 \\ 0 \text{ otherwise} \end{cases} \tag{1}$$

where, g_c is the gray level of the center pixel and g_i is the gray level of the surrounding pixels. Notation P is the number of sampling points on a radius of a circle (Maturana *et al.*, 2009) (Fig. 1a-c).

Face description: The face image will undergo pixel wise LBP operation based on the no. of sampling points and the radius set. The P, R = (8,1) is used as it is proven efficient. Now by using the above, a local description of face is made and globalizing the local descriptors to form a texture description of the whole face (Fig. 2a-b).

Local gradient oriented binary patterns: LGOBP is developed from LBP and a robust texture descriptor. LGOBP encodes higher order pixel wise information when compare to LBP and the invariant property of monotonic gray level in LBP is preserved (Ahonen *et al.*, 2006; Liao and Chung, 2009).

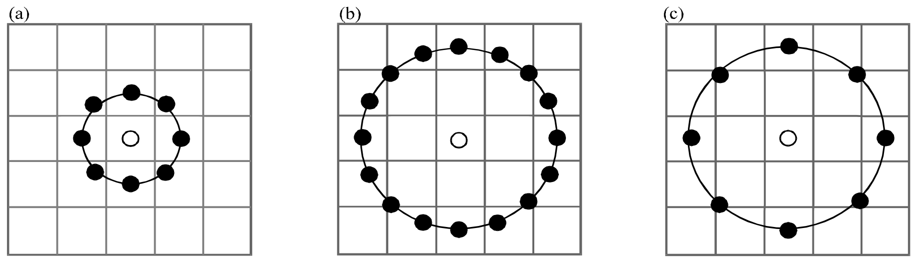

Fig. 1(a-c): Circular (a), (8,1), (b) (16, 2) and (c) (8, 1) neighbourhood are illustrated (Maturana *et al*., 2009)

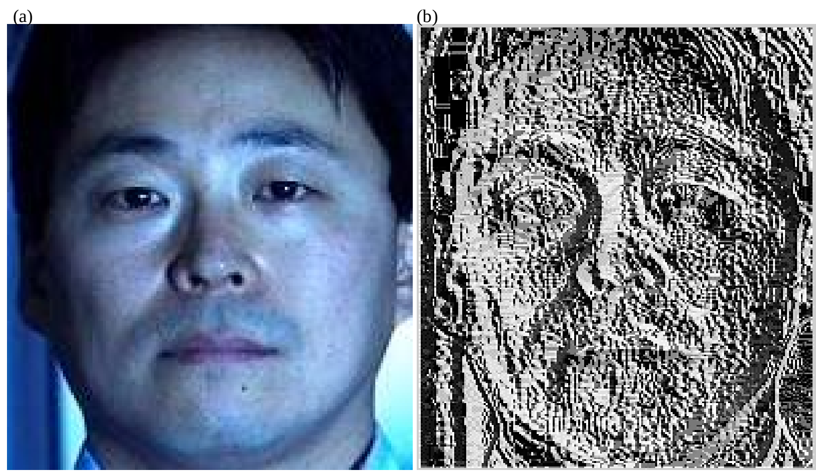

Fig. 2(a-b): Test image under low light conditions and LBP processed test image

LGOBP operator: For each pixel $\psi(\Delta_{vc})$, be the gradient orientation angle. A circularly symmetric neighborhood system with radius R is taken at center pixel (V_c) with k neighboring pixels uniformly located on the circle. Neighboring pixels that do not fall in the image grid are interpolated using bilinear interpolation. The k neighboring pixels are denoted as vi.
where, i = 1, 2,....k and $\psi(\Delta_{vc})$ gives the gradient orientation angle of V_i (Fig. 3a-b).

The gradient orientation space is uniformly divided into 4 subspaces, the pixels with gradient orientations that fall in the same subspace are considered with the same gradient orientations. Each pixel is given a label which means each pixel is given a label according to its gradient orientation:

$$\iota(v) = \begin{cases} 1, \text{ if } \psi(\nabla_v) \in [0, \frac{\pi}{2}] \\ 2, \text{ if } \psi(\nabla_v) \in [\frac{\pi}{2}, \pi)] \\ 3, \text{ if } \psi(\nabla_v) \in [\pi, \frac{3\pi}{2}] \\ 4, \text{ if } \psi(\nabla_v) \in [\frac{3\pi}{2}, 2\pi] \end{cases} \tag{2}$$

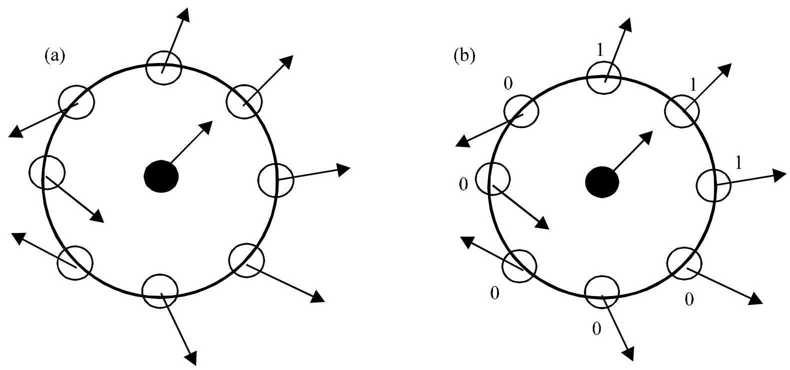

Fig. 3(a-b): Arrows represent the gradient orientations of different pixels and binary numbers are assigned to each neighboring pixel

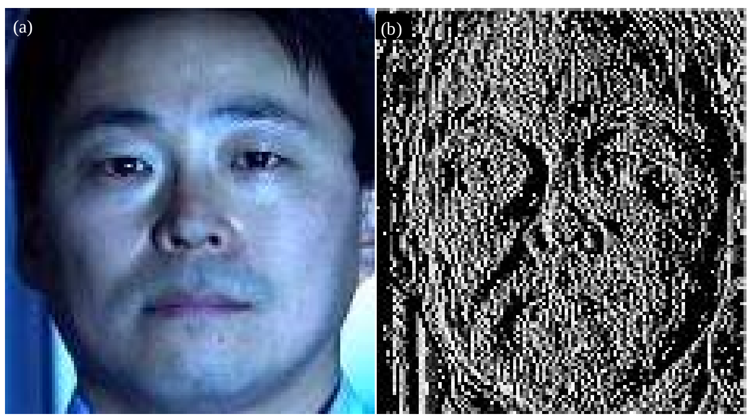

Fig. 4(a-b): A test image (a) Under low light conditions and (b) LGOBP processed

After the labels are allocated using the equation to analyze the pixel wise information between the center pixel V_c and the neighboring pixel V_i the corresponding labels obtained from Eq. 6 are compared and each neighboring pixel is assigned a binary number by the following equation:

$$B(v_i) = \begin{cases} 1, \text{ if } \iota(v_i) = \iota(v_c) \\ 0, \text{ if } \iota(v_i) \neq \iota(v_c) \end{cases} \tag{3}$$

Face recognition using LGOBP: LGOBP encodes the 2nd order pixel information because the gradient orientations already contain the 1st order pixel-wise properties, LGOBP is also monotonic gray level transformation invariant even though the gradient magnitude of each pixel changes their gradient orientations remain the same (Liao and Chung, 2009). Similar to LBP, LGOBP is uniform as the bit wise transition from 0-1 of the neighboring pixels are less than or equal to 2. All LGOBP image to from global image (Fig. 4a-b).

Oriented local histogram equalization: The difference between LHE and OLHE is the orientation (direction) of the edges of the image. In LHE, for each pixel, histogram equalization is performed on the width-by-height window centered on a particular pixel using:

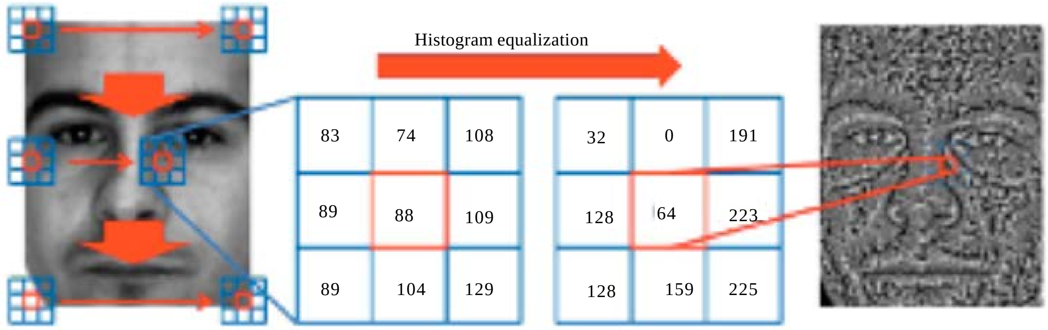

Fig. 5: Illustration of LHE

$$f(x) = \text{round}\left[\frac{cdf(x) - cdf_{min}}{\omega.h - cdf_{min}}.(L-1)\right] \tag{4}$$

where, x is the pixel intensity, cdf(x) is the cumulative distribution function of the histogram, cdf_{min} is the minimum intensity in the window and L is the number of output gray levels Lee *et al.* (2012). In a square window the center pixel is known as the anchor. Every pixel in the image repeats the above method and uses f(x) for its new intensity value (Fig. 5).

LHE operator: The generalized LHE operator:

$$\mathcal{L}_k^{\xi\,\eta}(Iw \times H) = I'_{W*H} \tag{5}$$

where, ξ and η are the relative positions of the anchor point to the pixel to be processed and $I_{(W*H)}$ is the input image and $I_{(W*H)}$ is the histogram equalized image. W and H are the width and height of the window respectively. The pixel which has to be processed will have a high intensity value when it is brighter than all the neighboring pixels before local histogram equalization.

OLHE operator: The anchor positions are changed to get the orientation out of LHE. In 3×3 windows 8 operators (ξ, η), other than the center pixel are oriented and hence they are known are as:

$$\begin{aligned}
\mathcal{O}_k^{\searrow} &\equiv \mathcal{L}_k\left(\frac{(k-1)}{2}, \frac{-(k-1)}{2}\right), & \mathcal{O}_k^{\downarrow} &\equiv \mathcal{L}_k\left(0, \frac{-(k-1)}{2}\right), \\
\mathcal{O}_k^{\swarrow} &\equiv \mathcal{L}_k\left(\frac{-(k-1)}{2}, \frac{-(k-1)}{2}\right), & \mathcal{O}_k^{\rightarrow} &\equiv \mathcal{L}_k\left(0, \frac{(k-1)}{2}, 0\right), \\
\mathcal{O}_k^{\leftarrow} &\equiv \mathcal{L}_k\left(\frac{-(k-1)}{2}, 0\right), & \mathcal{O}_k^{\swarrow} &\equiv \mathcal{L}_k\left(\frac{(k-1)}{2}, \frac{(k-1)}{2}\right), \\
\mathcal{O}_k^{\uparrow} &\equiv \mathcal{L}_k\left(0, \frac{(k-1)}{2}\right), & \mathcal{O}_k^{\nearrow} &\equiv \mathcal{L}_k\left(\frac{(-k-1)}{2}, \frac{(k-1)}{2}\right)
\end{aligned} \tag{6}$$

where, k is an odd number. These OLHE operators are applied on basic image elements to give results. LHE and OLHE operators give the same result for ideal edges. LHE operators do not

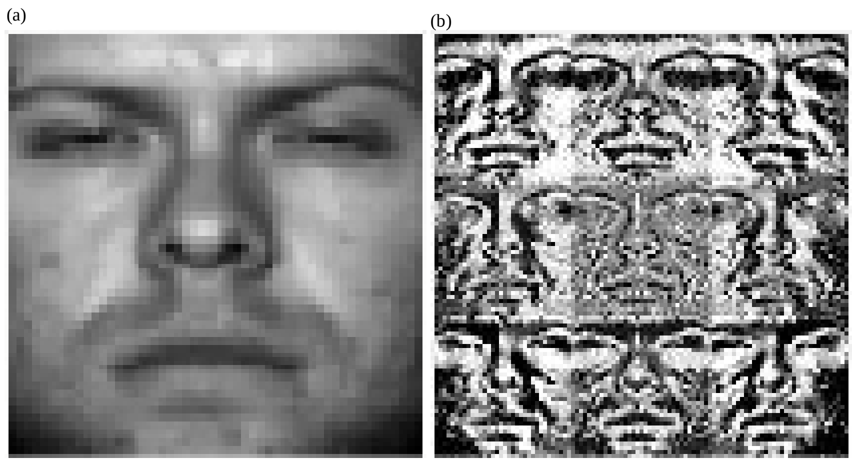

Fig. 6(a-b): Test image, (a) Under low light conditions and (b) OLHE processed

consider the direction of color gradients and hence give uniform responses which is not the case of OLHE operators. For example, if the intensities of the pixels of a local window at its upper left side are smaller than the intensity of the given pixel the strongest result (Fig. 6a-b).

Face recognition using OLHE

OLHE C: C stands for averaged. When a lower feature dimension is requested, the OLHE images can be combined to form the image that has the same dimensions of the input image (Fig. 7). This can be given by:

$$\begin{aligned}\mathcal{O}_k^c(\mathrm{I}) \equiv [&\mathcal{O}_k^{\searrow}(\mathrm{I})\, \mathcal{O}_k^{\downarrow}(\mathrm{I})\, \mathcal{O}_k^{\swarrow}(\mathrm{I})\, \mathcal{O}_k^{\rightarrow}(\mathrm{I}) \\ &\mathcal{O}_k^{\leftarrow}(\mathrm{I})\, \mathcal{O}_k^{\nearrow}(\mathrm{I})\, \mathcal{O}_k^{\uparrow}(\mathrm{l})\, \mathcal{O}_k^{\nwarrow}(\mathrm{l})]\end{aligned} \tag{7}$$

Generalized discriminant analysis: GDA is one of the efficient feature extractors. It has been developed from Fisher discriminant analysis. This technique has been derived from linear discriminant analysis to handle non linearity in images.

Implementation: Let us consider 'x' to be the input image and 'w' is a vector with coefficient α_i and 'x' is to be projected using kernel function $\phi(\bullet)$ and so following equation is obtain:

$$W = \sum_{i=1}^{n} \alpha_i \phi(x_i) \tag{8}$$

All the solutions of W lie in the span of ϕ (x_i). The main concept of GDA is about mapping the input space into a convenient feature space such that e input space is affiliated to the variables in a nonlinear manner:

$$\|a\text{-}b\| = \text{square root of } (\|a\|^2+\|b\|^2-2.a.b) \tag{9}$$

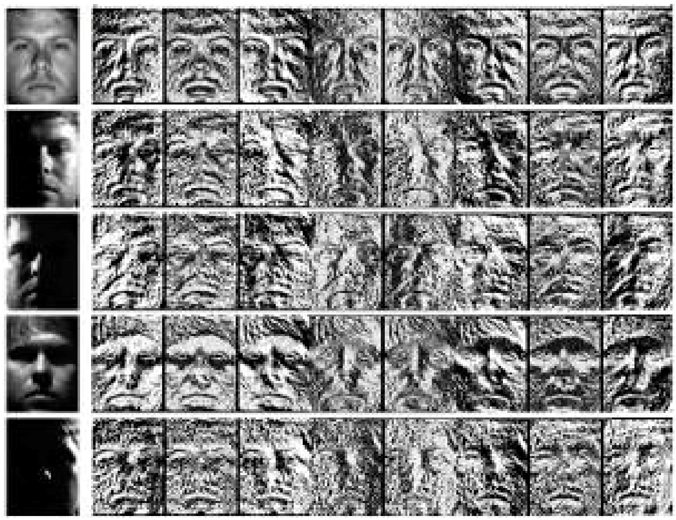

Fig. 7: Various test images and the 8 OLHE images obtained using the above operators

An eigenvalue solution gives us a clearer idea of how the concept of GDA is implemented. The Eigen values λ and eigenvectors W have to be found for the solution to the following equation:

$$\lambda S_W W = S_B W \tag{10}$$

The largest value of the previous equation gives the maximum of λ:

$$\lambda = \frac{\alpha^T\ KWK\alpha}{\alpha^T KK\alpha} \tag{11}$$

From this equation, first the matrices K and W are computed followed by the decomposition of K using eigenvector decomposition. Eigenvectors **β** and eigenvalues λ are secondly computed as they are present in the system. Eigenvectors W is then computed using α and then normalized which is followed by the computation of projections of test points on W.

Support vector machine: SVM works on the basis of searching for the best separating hyperplane, in the input space, between two classes (Deniz *et al*., 2003). The SVM finds the Optimal Separating Hyperplane when given a set of points of two classes. This Optimal Separating Hyperplane is a hyperplane that separates the largest fraction of points of the same class of the same side while the distance is maximized from either classes to the hyperplane. This margin is the only type of classifier that maximizes the margin which is the length between the hyperplane and the closest data point of each class (Cristianini and Shawe-Taylor, 2000). If the vector set are separated without error and has a maximum margin, this vector set is optimally separated.

The OSH is derived using the saddle point of Lagrange's functional given as:

$$L(w, b, \alpha) = \frac{1}{2} \| w \|^2 - \sum_{i=1}^{l} \alpha_i \{y_i\ [(w.\ x_i) + b] - 1\} \tag{12}$$

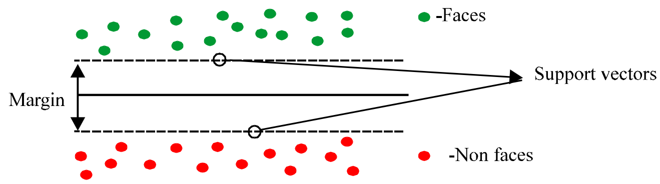

Fig. 8: Classification of data using a support vector machine

where, α_i is the Lagrange's multiplier.

In this case a 1 Dimensional Support vector machine has been used and the separating hyperplane is determined by the values of alpha and bias.

Once this has been done the test image will be sorted according to its group by the use of a decision making device that will sort the test image according to its group. Thus giving us the output, it is a Facial image or not (Fig. 8).

Programing methodologies

Local binary patterns: The traditional LBP methods suggested by Ahonen *et al.* (2006) was adapted initially. This algorithm led to high computational time and later it noticed that it was also encoding redundant/irrelevant data. A new and faster method (computer vision using local binary pattern-book), Fast Correlation Based Filtering (FCBF) algorithm. This algorithm selects only the distinct LBP patterns with the input facial image by rejecting those features that are more correlated than the image itself. The total time to process a single image in traditional LBP is around 7 sec but the FCBF processes in less than 0.2 sec and is given as:

$$r = \frac{\sum_i (x_i - \bar{x}_i)(y_i - \bar{y}_i)}{\sqrt{\sum_i (x_i - \bar{x}_i)^2}\sqrt{\sum_i (y_i - \bar{y}_i)^2}} \tag{13}$$

Local gradient oriented pattern: This is a derivative of the above mentioned LBP (Fig. 9). It captures neighbourhood gradient orientation information which is not considered in the above algorithm. In this algorithm, the gradient of every element is consider the 2-D array which is the input image. Once the gradients are obtained, use the LBP algorithm to check if the gradient of the neighbouring pixels are equivalent to the center pixel. If it is, then assigned '1', if not '0'.

By considering the gradient orientation of the neighbouring pixels the features of image are distinguish for more efficient machine learning (Fig. 10).

Oriented local histogram equalization: The 8 operators given by Lee *et al.* (2012) have been implemented in the algorithm. The above mentioned operators will encode the relative luminance of each pixel in 8 different directions using LHE operators. Hence, 8 images were obtain, will be used machine learning. The obtained images can be processed in two formats (Fig. 11).

OLHE C: The total size of this format is 8×W×H which will take a higher time for processing but this format of description retains very high information on edges and its orientation. As this image

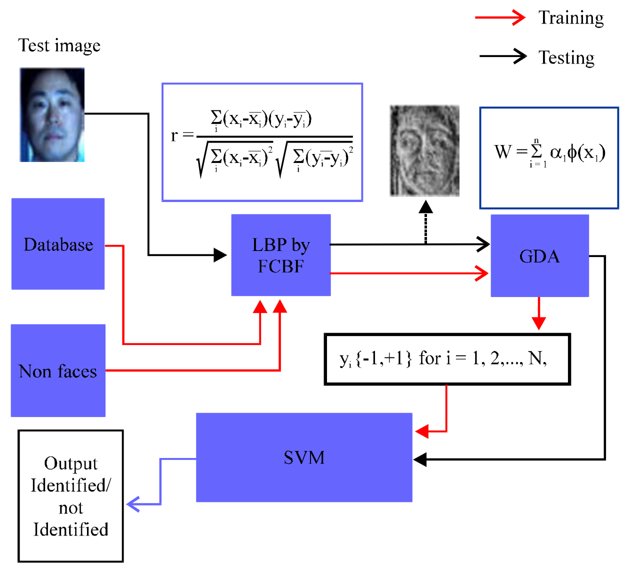

Fig. 9: Generalized block diagram for local binary patterns

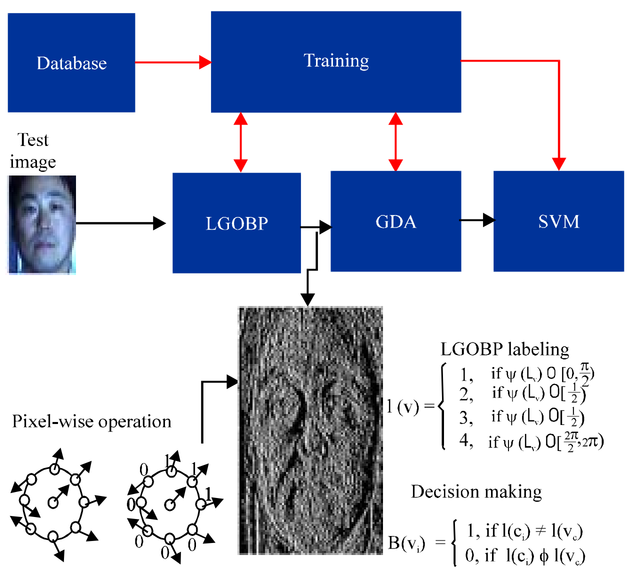

Fig. 10: Basic block operation of the local gradient oriented binary pattern

is too big the processing time is equivalently high. So, image size will be reduced. This will decreases the processing time but this must not depreciate the performance of the face recognition system.

Local binary patterns algorithm

Step 1: Input an image (imread) and filter dimensions (3×3)
Step 2: Convert the image from RGB to Gray
Step 3: Identify the size of the image
Step 4: Check if the filter dimensions are equal

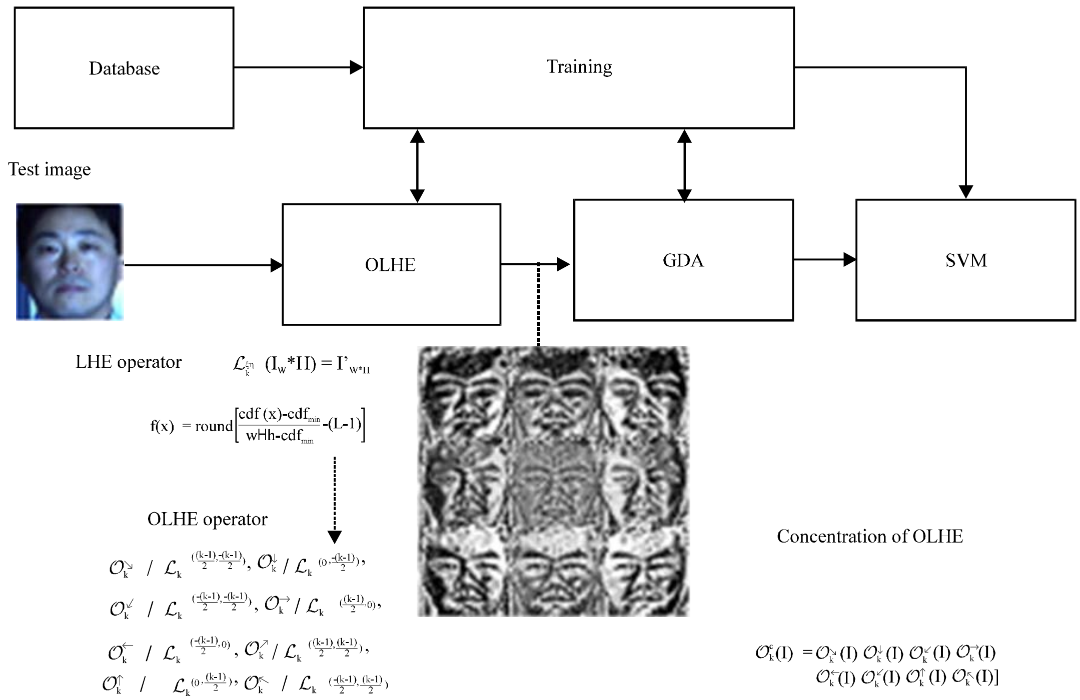

Fig. 11: Basic block operation of oriented local histogram equalization

Step 5: Initialize the variable filt with zeros
Step 6: Find out the number of neighbors according to filter dimensions.[nNeigh = numel(filt-1)]
Step 7: Initialize inmat dims variable with filt dims
Step 8: Reshape inmat dims into a one dimensional array
Step 9: Initialize indexing variable ihelix
Step 10: Concatenate ihelix by rotating inmat dims
Step 11: Then find the filter center with help of the equation [ceil((nNeight+1)/ 2]
Step 12: Now the filter center is removed
Step 13: Initialized variable filt is set with (filtcenter = 1) and the first element is -1
Step 14: Initializing the output variable to the size of the input image
Step 15: According to the indexing perform 2-dimensional filtering using Fir filter which returns the central part of the co-relation
Step 16: If the co-relation output is positive according to LBP equation, then one's and zeros will be assign
Step 17: Repeating the above process for all the neighbours so get the required output and summing all 8 images
Step 18: Output image is gotten

Extracting LBP features for faces

Step 19: Open standard dialog box for getting the directory to variable dir_name
Step 20: List the contents in the variable <files>

Step 21: Initialize LBP_face
Step 22: Initializing variable to call folders under nested loop to call the files within it
Step 23: Processing each file using LBP
Step 24: Processing the output LBP with Generalized Discriminant Analysis (GDA)
Step 25: The features are stored at LBP_face and are assigned by targets for the same, where LBP_face is sued as training data taken from group given by targets
Step 26: The trained Support Vector Machine (SVM) is saved as SVM structure LBP

OLHE Algorithm

Step 1: Read the input image
Step 2: Set the window size
Step 3: Initialize variables imrange = 256 and imrange new = 255
Step 4: Initialize variables n-posi equal to the window size to the power 2, then half bw becomes maskbw-1/2
Step 5: Assign the width of image to dimension1 and assign the height of the image to dimension 2
Step 6: Fill imageout {i} with zeros of the size of the input image
Step 7: Initialize variables, anchor array 2 dim as mask bw$^{\wedge 2}$ and initializing variable anchoronpatchX_ary, anchoronpatchY_ary,anchor_X_shift_ary anchor_Y_shift _ary as 1×9 zeros
Step 8: Initializing anchor array Idx = 1
Step 9: The above variable are assigned values to perform pixel wise peration on the image
Step 10: Initialize workinghist and working cdf to an array of zeros
Step 11: Initialize patch Temp EQ with zeros
Step 12: To compute the Local Histogram Equalization (LHE) for the whole image by initializing the patch temp with a 3×3 window of the input image
Step 13: Then extract the intensity values from the patch window and update the workinghist. The path is removed and a next patch is added and working hist are updated simultaneously
Step 14: Find the working cdf by finding the cumulative sum of the working histogram
Step 15: Thr cdf min (cumulative distribution function) of the working hist
Step 16: By using the LHE operator obtain histogram equalization of a local window
Step 17: After that perform OLHE operation by using the initialized anchor values
Step 18: Initialize 3×dim1 and 3×dim2 and fill it with zeros
Step 19: Add the obtained OLHE images into the above and remove the image borders

LGOBP Algorithm

Step 1: Read the input image
Step 2: Obtain the numeric gradient of each pixel for the input image
Step 3: Select no of sampling points (p) around the center pixel with radius (r)
Step 4: We label each neighboring pixel in the neighborhood according to where they lie in the 0,2 pie quadrants
Step 5: If the gradient orientation of the center pixel is equal to its neighbours then it assign binary ‘1’ to that particular pixel or a ‘0’

Step 6: Then concatenate the binary bits to form a decimal number, globalize to form the texture descriptor of the image

Algorithm for generalized discriminant Analysis

Step 1: Input the pre-processed image and the no. of dimensions
Step 2: Eliminate all the repeated elements
Step 3: Calculate the dimensions of the input image
Step 4: Sort the data according the index
Step 5: Compute gram matrix with the input data
Step 6: Gram matrix is computed by measuring the Euclidean distance by taking the transpose of the input image and compute the Euclidean distance (L2) between every vector
Step 7: Using steps 5 and 6 compute the Kernel matrix for the given input data
Step 8: Compute a centering matrix using the kernel matrix
Step 9: Eigen vector decomposition is performed on the Kernal matrix
Step 10: Sort the eigenvalues in descending order
Step 11: Remove eigenvectors with relatively small values
Step 12: Recompute the kernel matrix
Step 13: Construct a diagonal block matrix W with no. of classes in the image
Step 14: Determine the new dimensionality after reduction.
Step 15: Project the Kernel matrix into the new block diagonal matrix W
Step 16: Compute the block diagonal matrix
Step 17: Normalize the obtained values

Algorithm for support vector machine

Step 1: Input training data and group names (output of feature extractor and their targets)
Step 2: By default. kernel function is chosen.
Step 3: Check validity of all input variables
Step 4: Eliminate NaN elements
Step 5: Autoscale data is crated using the training data by computing mean and standard deviation
Step 6: Shift and scale each element of the data matrix obtained by the training data
Step 7: Hessian matrix is created using the target data
Step 8: By quadratic programming the variable alpha and others are determined using the above hessian matrix
Step 9: Calculate the line separating the support vectors
Step 10: Calculate the bias value by applying the largest alpha value
Step 11: Using alpha and the bias value calculate the separating value of the hyperplane

Output of trained classifier: Structure consist of:

- Support vectors
- Alpha
- Bias

- Kernel function
- Group names (targets)
- Support vector indices
- Scale data

Classification using support vector machine

- Input the trained data and the sample to be classified
- Verify the input data, if trained data is a structure the no. of arguments
- Get the group names from the trained data
- Sort the group index in terms of ascending order
- Shift the data if necessary using the scale data
- Use a decision making device
- Based on the alpha, bias and is kernel function the data classified
- If classified a binary '1' is the output else a binary '0'

RESULTS AND DISCUSSION

The comparative analysis of the three algorithms, OLHE, LBP and LGOP were performed on two public available database Extended Yale B (Patel *et al*., 2012) and CMU-PIE. These experiments provide the comparative analysis between the 3 state-of-the-art pre processing algorithms.

Dealing with only illumination compensation, considered only frontal illumination variant faces. Extended Yale B database offers the cropped version in portable gray scale format, of size 640×486 pixels and the algorithms were analyzed using standard protocols. Only the illumination variant images in CMU PIE out of all other images. The original study done by Hsieh and Tung (2009), the background removed and the image was reduced to a size of 80×100 pixels, when the algorithm is capable encoding such large information, it proposed that the faces would be recognized if image was much smaller which would rather offer faster computation time the background was removed and the image was reduced to 18×28 pixels.

The Support Vector Machine is used as a classifier in this case where as (Hsieh and Tung, 2009) used the K-nearest neighbor classifiers. Thus OLHE for the first time is being tested with SVM along with other state of the art algorithms.

The comparative analyze of the thee pre-processing histogram equalization (LBP, LGOBP, OLHE) were performed on two public available database Extended Yale B (Fig. 10 and 11), CMU PIE. The database offers the cropped version in portable gray scale format, of size 640×486 pixels and the algorithms were analyzed using standard protocols. It proposed that the faces would be recognized if image was much smaller which would rather offer faster computation time, the background was removed and the image was reduced to 18×28 pixels. The Support Vector Machine was used as a classifier in this case where as (Hsieh and Tung, 2009) used the K-nearest neighbor classifiers. Thus OLHE for the first time is being tested with SVM. Percentage recognition is obtain at rate as 97%, False Rejection Rate as 0.03 and False Acceptance Rate as 0.03 for OLHE.

Performance evaluation on extended: The extended Yale B contains 2214 images with frontal illumination variation faces of 39 individuals under 64 different illumination conditions. The uniformly lit images of these individuals were taken as the training images and the rest were taken as probe images for classification.

To test the efficiency of the system the test images were computed by a varying No. of probe images (Fig. 12). The Recognition Rate (%RR-Fig. 13), False Acceptance Rate (FAR-Fig. 14) and False Rejection Rate (FRR-Fig. 15) have been accounted for in Table 1 from which it can see that the OLHE out performs the other two algorithms LBP and LGOBP. It not only outperforms the other two algorithms but also is consistent for the test images.

Table 1: Comparison of the face identification Recognition Rate (RR%), False Acceptance Rate (FAR) and False Rejection Rate (FRR) of LBP, LGOBP and OLHE on Extended Yale B database

	Extended yale B database (SVM-LBP/LGOBP/OLHE)								
	LBP			LGOBP			OLHE		
No. of test images	RR(%)	FAR	FRR	RR (%)	FAR	FRR	RR (%)	FAR	FRR
15	80	0.25	0.20	86.66	0.20	0.14	80	0.08	0.20
20	85	0.10	0.15	85.00	0.12	0.15	90	0.07	0.10
50	84	0.09	0.16	84.00	0.10	0.16	91	0.06	0.09
100	87	0.07	0.13	90.00	0.08	0.10	93	0.05	0.07
150	86	0.06	0.14	91.00	0.08	0.09	93	0.03	0.07
200	88	0.05	0.12	91.00	0.07	0.09	85	0.03	0.04
300	94	0.05	0.06	92.00	0.06	0.08	97	0.03	0.03

Fig. 12: Training data from Extended Yale B used in the performance evaluation of the 3 algorithms

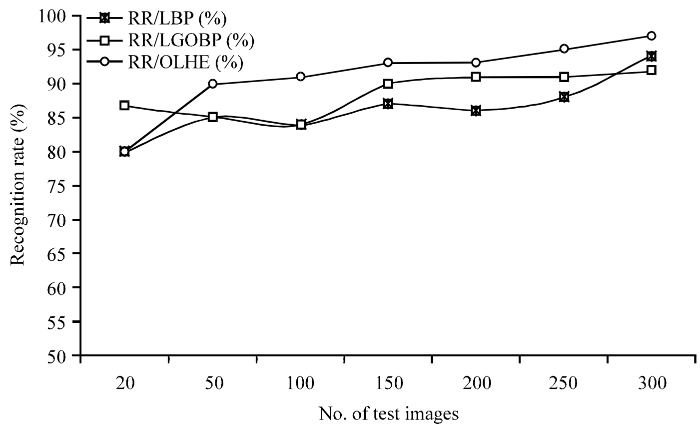

Fig. 13: Recognition rates for the three algorithms LBP, LGOBP and OLHE on extended YALE B

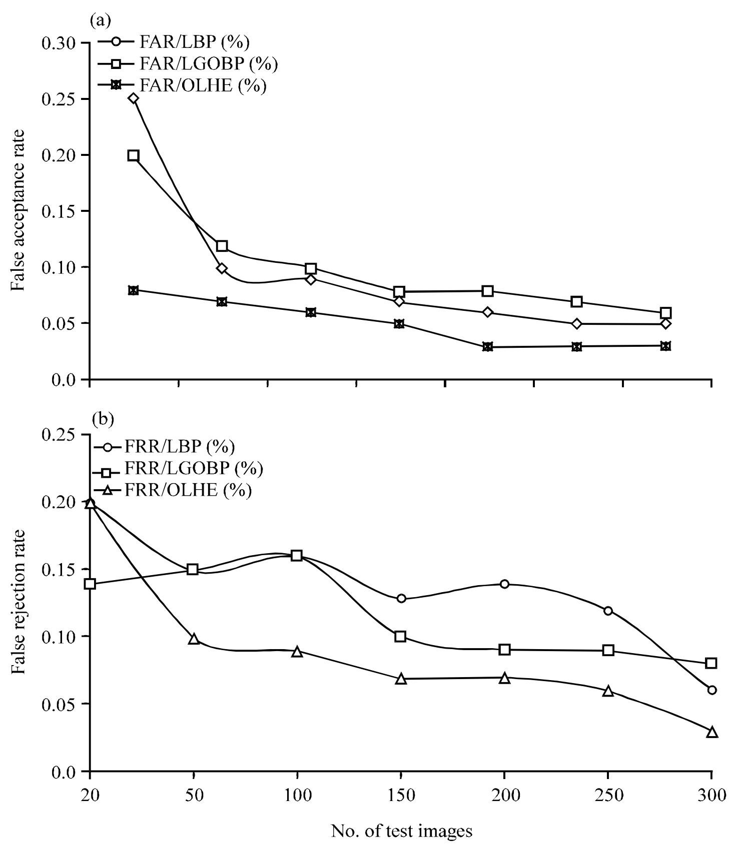

Fig. 14(a-b): Rate of the three algorithms LBP, LGOBP and OLHE, (a) False acceptance rate and (b) False rejection rate

Note that the OLHE has had almost all the test images recognized in every iteration. It also has the highest accuracy on the largest sample images.

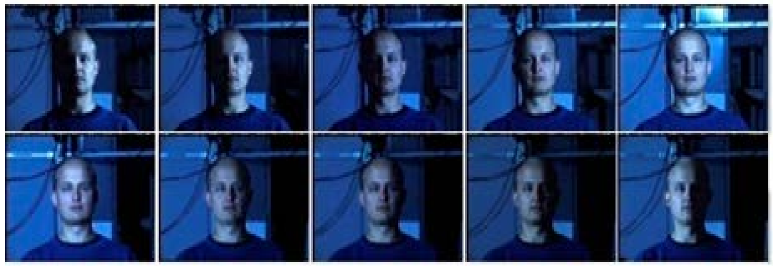

Fig. 15: Original images from the database

Table 2: Timing variation of OLHE with reduction of size in image in CMU-PIE database

Image size	Time (sec)
640×486	22.398
80×100	2.796
18×28	0.041

Where:

$$\text{Recognition rate (\%)} = \frac{\text{No. of persons recognized correctly}}{\text{Total No. of persons in the database}}$$

$$\text{FAR} = \frac{\text{No. of unauthorized persons accepted}}{\text{Total No. of persons in the database}}$$

$$\text{FRR} = \frac{\text{No. of authorized persons rejected}}{\text{Total No. of authorized persons}}$$

Time taken by MATLAB on simulation for different size of images. The original images of CMU-PIE database is 640×486 and it cropped the images into 80×100 pixel and 18×28 pixels and calculated the run-time for the simulation of images (Kukharev and Forczmanski, 2007). The total time is displayed in a Table 2.

Performance analysis of CMU-PIE: Different lighting conditions, The 21 frontal face image used such under different lighting conditions with no background. As it know that OLHE preserves data and the suggested compact representation of OLHE leads to data loss, the image size reduced drastically to 18×28 pixels per image, this method also decreases the computation time for each image. CMU-PIE (Patel *et al*., 2012) database contains 1250 images of 68 individuals under 43. Only three hundred image used for testing. The performance of the three preprocessing algorithms were tested on this modified database by taking 2 samples of each individual (Fig. 15-18).

The performance of LBP, LGOBP and OLHE on the modified database is shown in Table 3. The OLHE performed exceptionally well, it not only outclasses the previous state of the art algorithms LBP and LGOBP but also provides consistent result on every iteration of the test.

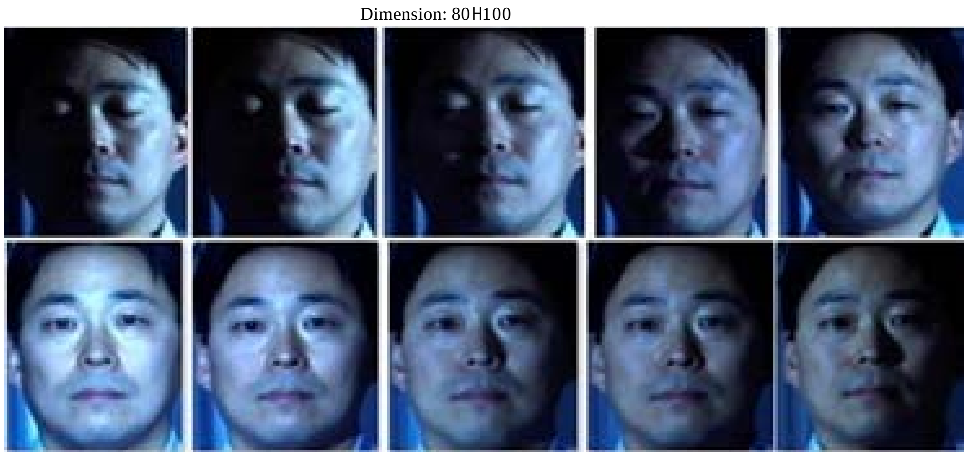

Fig. 16: Only the frontal face illumination variant images

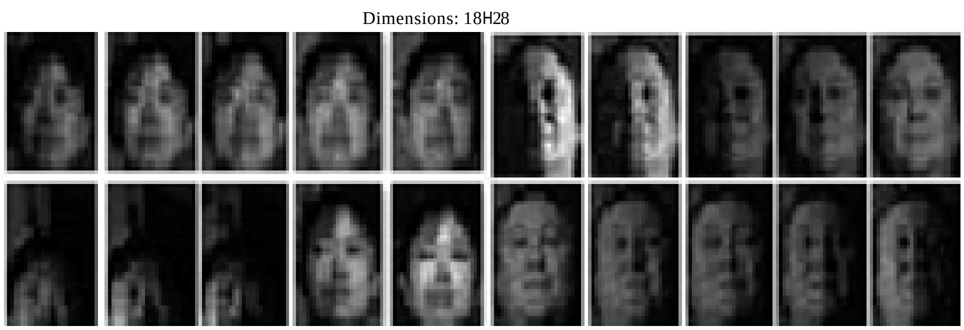

Fig. 17: Down-sampled images of size 18×28

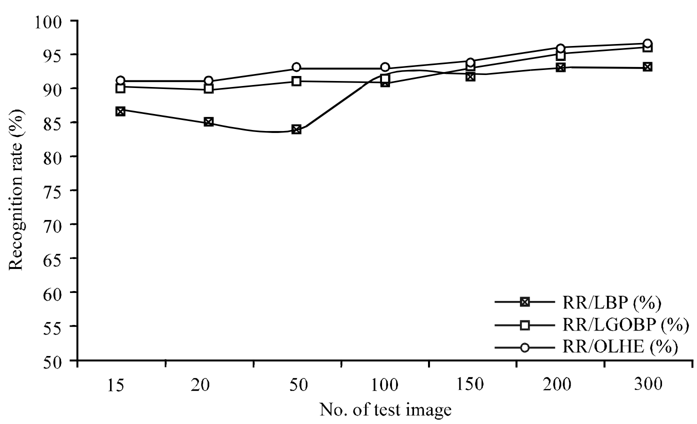

Fig. 18: Percentage recognition rate of the three algorithms LBP, LGOBP and OLHE on CMU-PIE database

The Support Vector Machine was used as a classifier in present case where as Hsieh and Tung (2009) used the K-nearest neighbor classifiers. Thus OLHE for the first time is being tested

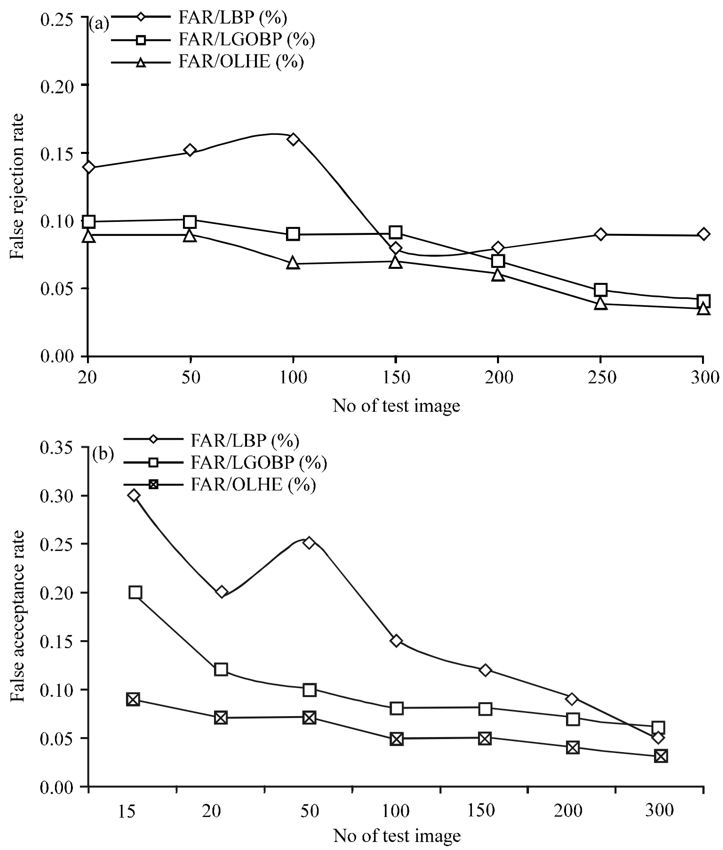

Fig. 19(a-b): Rate of the three algorithms LBP, LGOBP and OLHE, (a) False rejection and (b) False acceptance

Table 3: Comparison of the face identification Recognition Rate (RR%), False Acceptance Rate (FAR) and False Rejection Rate (FRR) of LBP, LGOBP and OLHE on CMU-PIE database

	CMU-PIE database (SVM-LBP/LGOBP/OLHE)								
	LBP			LGOBP			OLHE		
No. of test images	RR(%)	FAR	FRR	RR(%)	FAR	FRR	RR(%)	FAR	FRR
15	86.66	0.30	0.14	90	0.20	0.10	91	0.09	0.09
20	85	0.20	0.15	90	0.12	0.10	91	0.07	0.09
50	84	0.25	0.16	91	0.10	0.09	93	0.07	0.07
100	92	0.15	0.08	91	0.08	0.09	93	0.05	0.07
150	92	0.12	0.08	93	0.08	0.07	94	0.05	0.06
200	93	0.09	0.09	95	0.07	0.05	96	0.04	0.04
300	93	0.05	0.09	96	0.06	0.04	96.5	0.03	0.035

with SVM. Percentage rate is obtain as 96.5% (Fig. 18), False Rejection Rate as 0.035 (Fig. 19a) and False Acceptance Rate as 0.03 (Fig. 19b) for OLHE.

CONCLUSION

As it is proven that the efficiency of a face recognition system critically relies on compensating the illumination variation of the face images. The three state of the art algorithms LBP, LGOBP and OLHE were critically analyzed on the two datasets which leads us to the conclusion that OLHE is the best performing illumination compensation till date. Although a compact representation of OLHE without data loss is not available. Present experiment by reducing the size of the training and test images led to a faster and accurate representation of OLHE. Thus making it acceptable for practical usage.

REFERENCES

Ahonen, T., A. Hadid and M. Pietikainen, 2006. Face description with local binary patterns: Application to face recognition. IEEE Trans. Pattern Anal. Mach. Intell., 28: 2037-2041.

Corcoran, P. and G. Costache, 2005. Automated sorting of consumer image collections using face and peripheral region image classifiers. IEEE Trans. Consumer Electron., 51: 747-754.

Cristianini, N. and J. Shawe-Taylor, 2000. An Introduction to Support Vector Machines and Other Kernel-based Learning Methods. Cambridge University Press, Cambridge, UK., ISBN-13: 9780521780193, Pages: 189.

Deniz, O., M. Castrillon and M. Hernandez, 2003. Face recognition using independent component analysis and support vector machines. Pattern Recognit. Lett., 24: 2153-2157.

Hadid, A., M. Pietikainen and T. Ahonen, 2004. A discriminative feature space for detecting and recognizing faces. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Volume 2, June 27-July 2, 2004, Washington, DC., USA., pp: 797-804.

Hsieh, P.C. and P.C. Tung, 2009. A novel hybrid approach based on sub-pattern technique and whitened PCA for face recognition. Pattern Recogn., 42: 978-984.

Kukharev, G. and P. Forczmanski, 2007. Facial images dimensionality reduction and recognition by means of 2DKLT. Mach. Graph. Vision Int. J., 16: 401-425.

Lee, P.H., S.W. Wu and Y.P. Hung, 2012. Illumination compensation using oriented local histogram equalization and its application to face recognition. IEEE Trans. Image Process., 21: 4280-4289.

Liao, S. and A.C.S. Chung, 2009. Face recognition with salient local gradient orientation binary patterns. Proceedings of the 16th IEEE International Conference on Image Processing, November 7-10, 2009, Cairo, pp: 3317-3320.

Lizama, E., D. Waldoestl and B. Nickolay, 1997. An eigenfaces-based automatic face recognition system. Proceedings of the IEEE International Conference on Systems, Man and Cybernetics, Computational Cybernetics and Simulation, Volume 1, October 12-15, 1997, Orlando, FL., pp: 174-177.

Maturana, D., D. Mery and A. Soto, 2009. Face recognition with local binary patterns, spatial pyramid histograms and naive Bayes nearest neighbor classification. Proceedings of the International Conference of the Chilean Computer Science Society, November 10-12, 2009, Santiago, Chile, pp: 125-132.

Patel, V.M., T. Wu, S. Biswas, P.J. Phillips and R. Chellappa, 2012. Dictionary-based face recognition under variable lighting and pose. IEEE Trans. Inform. Forensics Secur., 7: 954-965.

Priyadarsini, M.J.P., K. Murugesan, S.R. Inbathini and V. Kumar, 2012. Efficient face recognition in video by bit planes slicing. J. Comput. Sci., 8: 26-30.

Shan, S., W. Gao, B. Cao and D. Zhao, 2003. Illumination normalization for robust face recognition against varying lighting conditions. Proceedings of the IEEE International Workshop on Analysis and Modeling of Faces and Gestures, October 17, 2003, Nice, France, pp: 157-164.