IOSR Journal of Electronics and Communication Engineering (IOSR-JECE)
e-ISSN: 2278-2834,p- ISSN: 2278-8735.Volume 9, Issue 2, Ver. IV (Mar - Apr. 2014), PP 20-36
www.iosrjournals.org

Digital Face Recognition Techniques for Matching with Sketches

Surya Raj[1], Navya Jose[2]
[1,2]*(Department of Electronics, College of Engineering Cherthala/ CUSAT, India)*

Abstract: *Face recognition is a well studied problem in many application domains. However, matching sketches with digital face images is a very important law enforcement application that has received relatively less attention. The state-of art face recognition algorithms cannot be used directly and require additional processing to address the non-linear variations present in sketches and digital face images. One of the important clues in solving crimes and apprehending criminals is matching sketches with digital face images. This paper presents an automated algorithm that extracts discriminating information from local regions of both sketches and digital face images. Structural information along with minute details present in local facial regions are encoded using multi scale circular Weber's local descriptor. Further, an evolutionary Memetic Optimization is proposed to assign optimal weights to every local facial region to boost the identification performance. Since, forensic sketches or digital face images can be of poor quality, a pre-processing technique is used to enhance the quality of images and improve the identification performance. Comprehensive experimental evaluation on different sketch databases show that the proposed algorithm yields better identification performance compared to existing face recognition algorithms and two commercial face recognition systems.*

Keywords: *Difference of Gaussian(DoG), Scale Invariant Feature Transform(SIFT), Extended uniform Circular Local Binary Pattern(EUCLBP), Genetic Algorithm(GA),Multi scale Local Binary Pattern(MSLBP), Local Feature- based Descriminent Analysis(LFDA), Cumulative Match Characteristic(CMC), Multi scale Circular Weber's Local Descriptor(MCWLD), Memetic Algorithm(MA), Weber's Local Descriptor(WLD).*

I. INTRODUCTION

Face recognition is becoming an increasingly important for many applications including human machine interfaces, multimedia, security, communication, visually mediated interaction and anthropomorphic environments. However, matching sketches with digital face images is a very important law enforcement application that has received relatively less attention. Current research efforts in the field involve developing more accurate and robust algorithms for face detection and recognition as well as algorithms to classify faces with respect to gender, race, and age Forensic sketches are drawn based on the recollection of an eye-witness and the expertise of a sketch artist. One of the most difficult problems is that the process of identifying a person from facial appearance has to be performed differently for each image, because there are so many conflicting factors altering facial appearance. There are many face recognition techniques are present like SIFT, LBP, WLD, MCWLD etc. This seminar paper is comparative study of these methods. Among these the most advanced technique is memetically optimized MCWLD which is a discriminative approach for matching sketch-digital image pairs using modified Weber's local descriptor and memetically optimized weighted chi-square distance.

II. FACE RECOGNITION USING SIFT FEATURES

2.1 BASIC APPROACH

Scale Invariant Feature Transform (SIFT) features are features extracted from images to help in reliable matching between different views of the same object. The extracted features are invariant to scale and orientation, and are highly distinctive of the image. They are extracted in four steps. The first step computes the locations of potential interest points in the image by detecting the maxima and minima of a set of Difference of Gaussian (DoG) filters applied at different scales all over the image. Then, these locations are refined by discarding points of low contrast. An orientation is then assigned to each key point based on local image features. Finally, a local feature descriptor is computed at each key point. This descriptor is based on the local image gradient, transformed according to the orientation of the key point to provide orientation invariance. Every feature is a vector of dimension 128 distinctively identifying the neighborhood around the key point.

2.2. THE METHOD

Face identification is supposed to discriminate between different faces. In this paper, the nearest-neighbour template matching is presented. The nearest neighbor is defined as the keypoint with minimum Euclidean distance for the invariant descriptor vector. If there are multiple training images of the same face, then we define the second-closest neighbor as being the closest neighbor that is known to come from a different face than the first. Because of the variability in facial appearance statistical methods are often used. Here use SIFT features grouped using a probabilistic model initialized with a few parts of an object. It learns the parts incrementally and tries adding possible parts to the model during the process of training. Then use the Expectation-Maximization (EM) algorithm to update the model. SIFT method is implemented as the following stages: Creating the Difference of Gaussian Pyramid, Extrema Detection, Noise Elimination, Orientation Assignment, Descriptor Computation, Keypoints Matching[5].

2.2.1. CREATING THE DIFFERENCE OF GAUSSIAN PYRAMID

The first stage is to construct a Gaussian "scale space" function from the input image. This is formed by convolution (filtering) of the original image with Gaussian functions of varying widths. The difference of Gaussian (DoG), D(x, y, σ), is calculated as the difference between two filtered images, one with k multiplied by scale of the other:

$$D(x,y,\sigma) = L(x,y,k\sigma)-L(x,y,\sigma) \qquad (1)$$

These images, L(x, y, σ), are produced from the convolution of Gaussian functions, G(x, y, kσ),with an input image, I(x, y):

$$L(x,y,\sigma) = G(x,y,\sigma) * I(x,y) \qquad (2)$$

$$G(x,y,\sigma) = \frac{1}{2\pi\sigma^n} \exp\{-\frac{x^2+y^2}{2\sigma^2}\} \qquad (3)$$

This is the approach used in the implementation. First, the initial image, I, is convolved with a Gaussian function, G_0, of width σ_0. Then use this blurredimage,L_0, as the first image in the Gaussian pyramid and incrementally convolve it with a Gaussian, G_i, of width σ_i to create the i_{th} image in the image pyramid, which is equivalent to the original image filtered with a Gaussian, G_k, of width $k\sigma_0$. The effect of convolving with two Gaussian functions of different widths is most easily found by converting to the Fourier domain, in which convolution becomes multiplication i.e.:

$$G_{\sigma i} * G_{\sigma o} * f(x) \rightarrow \tilde{G}\sigma i\tilde{G}\sigma o\tilde{f} \qquad (4)$$

The Fourier transform of a Gaussian function e^{ax^2} is given by:

$$F_x[e^{ax^2}](t) = \sqrt{\frac{\pi}{a}\ e^{-\pi^2(t)^2/a}} \qquad (5)$$

Substituting this into equation (4) and equating it to a convolution with a single Gaussian of width $k\ \sigma_0$ we get:

$$e^{-t^2\sigma_i{}^2} e^{-t^2\sigma_o{}^2} = e^{-t^2k^2\sigma_o{}^2} \qquad (6)$$

Performing the multiplication of the two exponentials on the left of this equation and comparing the coefficients of $-t^2$gives:

$$\sigma_i{}^2 + \sigma_o{}^2 = k^2\sigma_0{}^2 \qquad (7)$$

And so we get:

$$\sigma_i = \sigma_0\sqrt{(k^2} - 1) \qquad (8)$$

After subsampling of the low-passed filtered images to form the lower levels of the pyramid no longer have access to the original image at the appropriate resolution, and so cannot filter with G_k directly. Each octave of scale space in divided into an integer number, *s*, of intervals and we let k=$2^{1/s}$. We produce s+3 images for each octave in order to form s+2 difference of Gaussian (DoG) images and have plus and minus one scale interval for each DoG for the extrema detection step. Once a complete octave has been processed, then subsample the Gaussian image that has twice the initial value of σ by taking every second pixel in each row and column. This greatly improves the efficiency of the algorithm at lower scales. The process is shown in Fig 1.

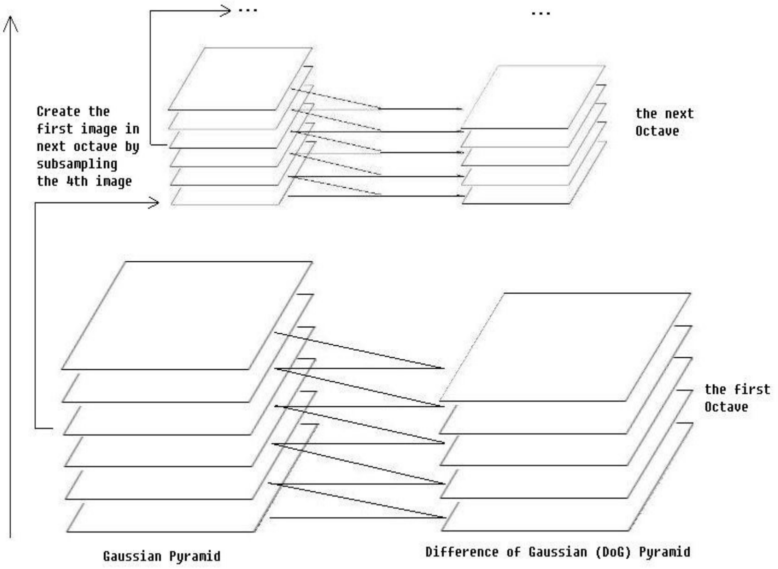
Create the first image in next octave by subsampling the 4th image
the next Octave
the first Octave
Gaussian Pyramid
Difference of Gaussian (DoG) Pyramid

Fig 1: Bottom: On the left is the Gaussian pyramid, with neighboring images separated by a constant scale factor. These are subtracted to give the DoG pyramid on the right. Top: The Gaussian with σ twice that of the original is sub sampled and used to construct the next octave

2.2.2. EXTREMA DETECTION

This stage is to find the extrema points in the DoG pyramid. To detect the local maxima and minima of D(x, y, σ), each point is compared with the pixels of all its 26 neighbors (Fig 2). If this value is the minimum or maximum this point is an extrema. Then improve the localization of the keypoint to subpixel accuracy, by using a second order Taylor series expansion. This gives the true extrema location as:

$$z = -\left(\frac{\partial^2 D}{\partial x^2}\right)^{-1} \frac{\partial D}{\partial x} \qquad (9)$$

where D and its derivatives are evaluated at the sample point and $x=(x,y,\sigma)^T$ is the offset from the sample point.

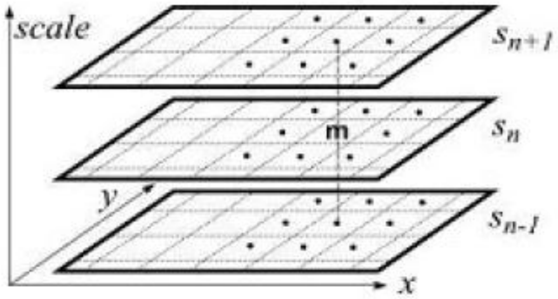

Fig 2: An extrema is defined as any value in the DoG greater than all its neighbours in scale-space.

2.2.3 KEY POINTS ELIMINATION

This stage attempts to eliminate some points from the candidate list of keypoints by finding those that have low contrast or are poorly localised on an edge. The value of the keypoint in the DoG pyramid at the extrema is given by:

$$D(z) = D + \frac{1}{2}\frac{\partial D^{-1}}{\partial x} z \qquad (10)$$

If the function value at **z** is below a threshold value this point is excluded.

To eliminate poorly localized extrema we use the fact that in these cases there is a large principle curvature across the edge but a small curvature in the perpendicular direction in the difference of Gaussian function.

A 2x2 Hessian matrix, H, computed at the location and scale of the keypoint is used to find the curvature. With these formulas, the ratio of principal curvature can be checked efficiently:

$$H = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix} \qquad (11)$$

$$\frac{D_{xx} + D_{yy}}{D_{xx} D_{yy} - (D_{xy})^2} < \frac{(r+1)^2}{r} \qquad (12)$$

So if inequality (12) fails, the key point is removed from the candidate list.

2.2.4 ORIENTATION ASSIGNMENT

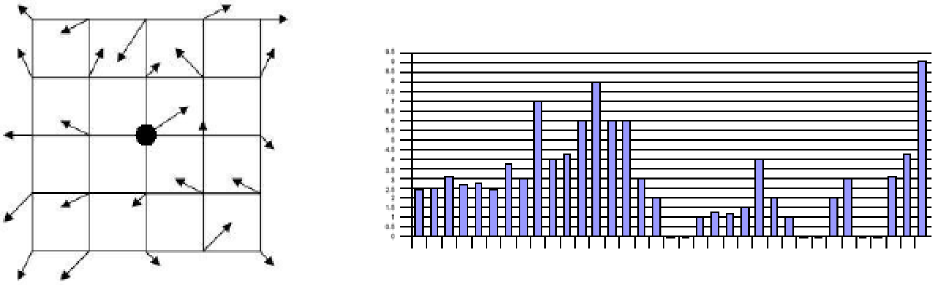

Fig 3: Left: The point in the middle of the left figure is the key point candidate. The orientations of the points in the square area around this point are pre computed using pixel differences. Right: Each bin in the histogram holds 10 degree, so it covers the whole 360 degree with 36 bins in it. The value of each bin holds the magnitude sums from all the points precomputed within that orientation.

This step aims to assign a consistent orientation to the key points based on local image properties. An orientation histogram is formed from the gradient orientations of sample points within a region around the key point as illustrated in Figure 3. A 16x16 square is chosen in this implementation. The orientation histogram has 36 bins covering the 360 degree range of orientations. The gradient magnitude, m(x, y), and orientation, θ(x, y), are precomputed using pixel differences:

$$m(x, y) = \sqrt{(L(x+1, y) - L(x-1, y))^2 + (L(x, y+1) - L(x, y-1))^2} \qquad (13)$$

$$\theta(x, y) = \arctan\left(\frac{L(x, y+1) - L(x, y-1)}{L(x+1, y) - L(x-1, y)}\right) \qquad (14)$$

Each sample is weighted by its gradient magnitude and by a Gaussian-weighted circular window with a σ that is 1.5 times that of the scale of the keypoint. Peaks in the orientation histogram correspond to dominant directions of local gradients. We locate the highest peak in the histogram and use this peak and any other local peak within 80% of the height of this peak to create a keypoint with that orientation. This computes the location, orientation and scale of SIFT features that have been found in the image. These features respond strongly to the corners and intensity gradients. The SIFT features appear mostly in the eyes, nostrils, the top of the nose and the corners of the lips for face images. In Figure 4, keypoints are indicated as arrows. The length of the arrows indicates the magnitude of the contrast at the keypoints, and the arrows point from the dark to the bright side.

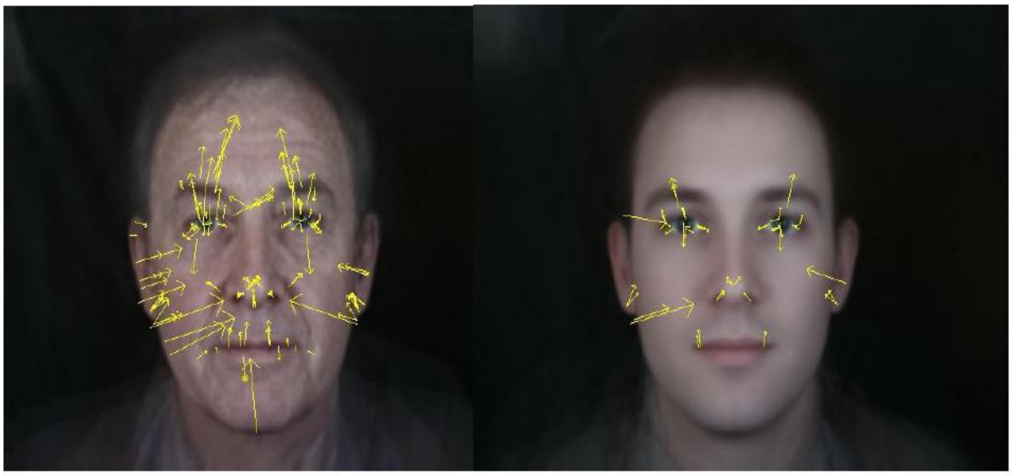

Fig 4: Keypoint vectors are drawn on the two men's face images. Most of the features appear on the eyes, nostrils, the top of nose, the corners of the mouth and the earlobes.

2.2.5 DESCRIPTOR COMPUTATION:

In this stage, a descriptor is computed for the local image region that is as distinctive as possible at each candidate keypoint. The image gradient magnitudes and orientations are sampled around the keypoint location. These values are illustrated with small arrows at each sample location on the first image of Figures. A Gaussian weighting function with σ related to the scale of the keypoint is used to assign a weight to the magnitude. We use a σ equal to one half the width of the descriptor window in this implementation. In order to achieve orientation invariance, the coordinates of the descriptor and the gradient orientations are rotated relative to the key point orientation. This process is indicated in Figure 5. In our implementation, a 16x16 sample array is computed and a histogram with 8 bins is used. So a descriptor contains 16x16x8 elements in total.

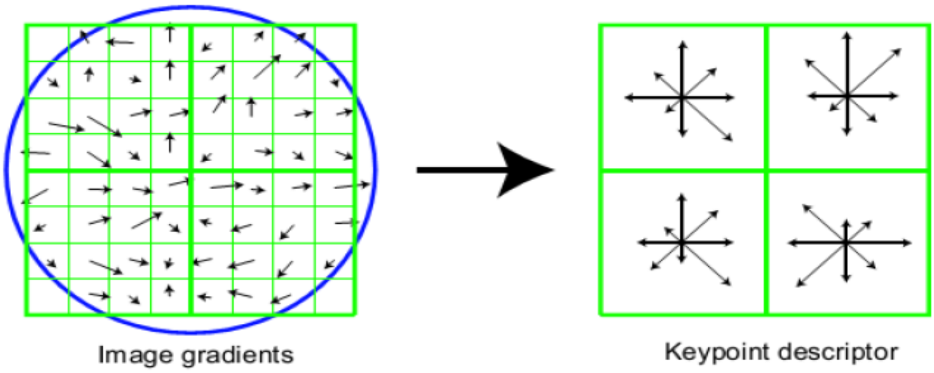

Fig 5: Left: the gradient magnitude and orientation at a sample point in a square region around the keypoint location. These are weighted by a Gaussian window, indicated by the overlaid circle. Right: The image gradients are added to an orientation histogram. Each histogram include 8 directions indicated by the arrows and is computed from 4x4 subregions. The length of each arrow corresponds to the sum of the gradient magnitudes near that direction within the region.

2.2.6 TRANSFORMATION

In this stage, some matching tests are running to test the repeatability and stability of the SIFT features. An image and a transformed version of the image are used as indicated in Figure 6. The features of the two images are computed separately. Then each keypoint in the original image (model image) is compared to every keypoints in the transformed image using the descriptors computed in the previous stage. For each comparison, one feature is picked in each image. f1 is the descriptor array for one key point in the original image and f2 is the descriptor array for a key point in the transformed image. The most likely value for each pair of the keypoints is computed by:

$$error_{ij} = \sum_{i=0}^{i=n1} \sum_{j=0}^{j=n2} |f1_i - f2_j| \qquad (15)$$

If the number of features in the two images is n1 and n2, then there are n1*n2 possible pairs altogether. All these data are sorted in ascending order of matching error. Then the first two qualified pairs of the keypoints are chosen to set the transformation.

To apply the transform, the following functions are introduced into the implementation. The transform gives the mapping of a model image point (x, y) to a transformed image point (u, v) in terms of an image scaling, s, an image rotation, θ, and an image translation,[t_x, t_y]:

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} s\,cos\theta & -s\,sin\theta \\ s\,sin\theta & s\,cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} \qquad (16)$$

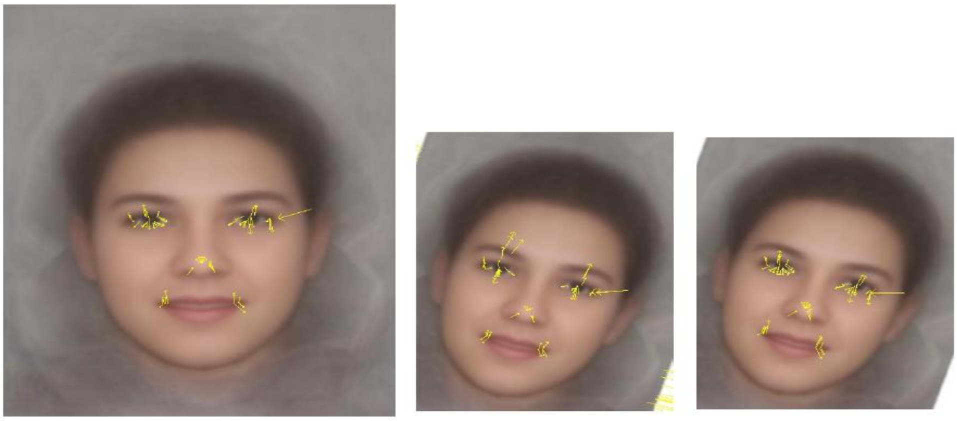

Fig 6: The first image is the original image with its own features. The second image is the transformation of the original with the features after operation. The third one is the transformation of the original image with the features before the operation.

2.3 ANALYSIS

Here calculation of the SIFT features for an image is carried out and have experimented with some simple matching schemes between images. The SIFT features described in this implementation are computed at the edges and they are invariant to image scaling, rotation, addition of noise. They are useful due to their distinctiveness, which enables the correct match for keypoints between faces. Noise adjustment is a very essential part for this approach which could result in inefficient or false matching. However, the used parameters which should help the keep the feature matching robust to noise in this implementation.

2.4. DRAWBACKS

The average number of SIFT features extracted decreases with decreasing the resolution of the image. However, the resolutions up to50% give comparable results to those at full scale, while at 25% the accuracy decreases significantly. SIFT computes only for the regions of interest (located around detected interest points) that have usually already been normalized with respect to scale and rotation.

III. Face Description With Local Binary Patterns

Local Binary Pattern (LBP) is a simple yet very efficient texture operator which labels the pixels of an image by thresholding the neighborhood of each pixel with the value of the center pixel and considers the result as a binary number. Due to its discriminative power and computational simplicity, LBP texture operator has become a popular approach in various applications. Here the derivation of efficient LBP based face descriptions which combine into a single feature vector the global shape, the texture and eventually the dynamics of facial features is carried out. The obtained representations are then applied to face and eye detection, face recognition and facial expression analysis problems, yielding in excellent performance.

3.1. BASIC APPROACH

Local Binary Patterns (LBP) based descriptor is a widely used texture operator because of its robustness to graylevel changes and high computational efficiency. Basic LBP is a window based feature extractor where the texture descriptor is computed based on the neighborhood. It assigns a binary value to every neighboring pixel by thresholding it with respect to the central pixel. The binary patterns thus obtained from the neighboring pixels are transformed to a gray-level value and is assigned to the central pixel. LBP representation of a given face image is generated by dividing the image into grids and computing histograms to measure the frequency of LBP values within each grid[4].

3.1.1 FEATURE EXTRACTION USING EUCLBP:

An extension of basic LBP is to have the pixel neighbours well separated on a circle around a central pixel. The circle can have different diameters and varying number of neighbors to account for texture at different scales. Similar to basic LBP, Circular LBP (CLBP) descriptor is computed as shown in Eqns. (17) and (18):

$$C_{N,R}(p,q) = \sum_{i=0}^{N-1} f(n_i - n_c)2^i \qquad (17)$$

$$f(.) = \begin{cases} 1 & if\ n_i - n_c \geq 0 \\ 0 & otherwise \end{cases} \qquad (18)$$

where N is the number of neighbors, $n_c$corresponds to the gray-level intensity of center pixel of the circle, and $n_i$corresponds to the gray-level intensities of Nevenly spaced pixels on a circle of radius R.

CLBP is extended to Uniform Circular Local Binary Patterns to achieve robustness to rotation variations and dimensionality reduction. A binary pattern is called uniform binary pattern if it has at most two bitwise transitions from 0 to 1 or vice-versa. Descriptor histogram is computed where every uniform pattern has a separate bin and all non-uniform patterns are assigned to a single bin. The concatenation of all histograms pertaining to each grid constitutes the image signature. Uniform CLBP is described using Eqns. (19) and (20).

$$C_{N,R}^{riu2}(p,q) = \begin{cases} \sum_{i=0}^{N-1} f(n_i - n_c)2^i & if\ U(C_{N,R}) \leq 2 \\ N+1 & otherwise \end{cases} \qquad (19)$$

Where,

$$U(C_{N,R}) = \sum_{i=1}^{N-1} |f(n_i - n_c) - f(n_{i-1} - n_c)| + |f(n_{N-1} - n_c) - f(n_0 - n_c)| \qquad (20)$$

where n_c corresponds to the gray-level intensity of center pixel of the circle and $n_i$corresponds to the gray-level intensities of N evenly spaced pixels on a circle of radius R.$riu2$ represents the use of rotation invariant uniform patterns.

Huang*et.al.* proposed a method to encode the exact difference of gray-level intensities and reported a marked improvement in the performance of texture descriptors. This forms the motivation to further extend Uniform CLBP to encode exact gray-level difference along with the original encoding. The proposed descriptor is called *Extended Uniform Circular Local Binary Pattern*. It provides information assimilated from the exact gray-level difference and adds a complimentary layer of discrimination on top of the original descriptor. Fig.7 (a) explains feature extraction using the proposed EUCLBP. Layer 1 is Uniform CLBP that encodes difference of signs while the other three layers encode the exact gray-level differences. We experimentally observed that Layer 1 and Layer 2 of EUCLBP are the most discriminating. Therefore, the final descriptor is the concatenation of Layer 1 and Layer2 histograms [6]

3.1.2 WEIGHTED CHI SQUARE MATCHING USING GENETIC OPTIMIZATION:

As shown in Eqn. (21), weighted chi square distance measure can be used to match EUCLBP descriptors:

$$\chi^2(x,y) = \sum_{i,j} w_j \frac{(x_{i,j} - y_{i,j})^2}{x_{i,j} + y_{i,j}} \qquad (21)$$

Where x and y are the normalized histograms (EUCLBP features), i and j correspond to the i^{th} bin of the j^{th} local region and $\omega_j$is the weight for the j^{th} region. Ahonen proposed to assign weights proportional to the identification accuracy of local facial regions.

A Genetic Algorithm (GA) based weight optimization technique is used here. It is a search optimization technique stimulated by the process of natural selection and evolution. GAs is efficient in searching large, non-linear search spaces. The motivation to use genetic optimization to find optimum weights for each facial patch is inspired by several properties of GA. The problem of finding the optimal weights embroils searching very large spaces and finding several suboptimal solutions. GAs is well proven in searching very large spaces to quickly reach to the near optimal solution. Fig. 7(b) represents the genetic search process to find optimal weights in weighted chi square distance measure based matching. The steps involved area) Encoding b) Initial population c) Fitness Function d) Mutation. The search process is repeated till convergence, i.e. till the identification accuracy for new generation does not improve. At this point, weights pertaining to the best performing chromosome are used for testing.

Genetic algorithm finds optimal weights for each facial region. It also enables to discard redundant and non discriminating regions whose contribution towards recognition accuracy is very low (i.e. the weight for that region is 0). This leads to dimensionality reduction and better computational efficiency because then we do not need to compute texture descriptors for poor performing facial regions during testing.

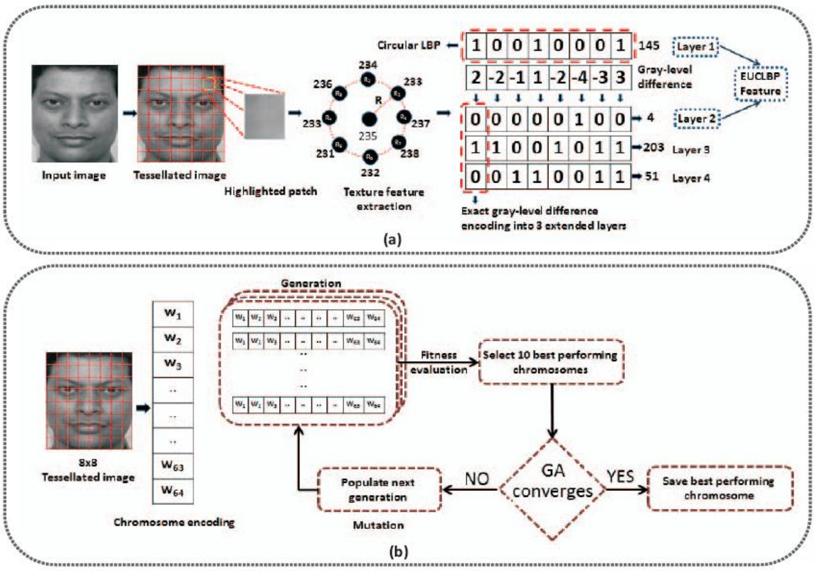

Fig. 7.Illustrating the details of feature extraction and matching, (a) feature extraction using EUCLBP and (b) weight optimization using genetic algorithm.

3.2 SKETCHES TO DIGITAL FACE RECOGNITION:

A face sketch is primarily an edge representation of the actual face in which prominent edges are highlighted; edge preserving approaches can be used for this task. As shown in Fig. 1, Laplacian pyramid are constructed for the sketch-digital image pairs to preserve edges. Laplacian pyramid provides an added advantage of addressing non-linearity by providing mechanism to extract and match facial features at different levels of granularity.

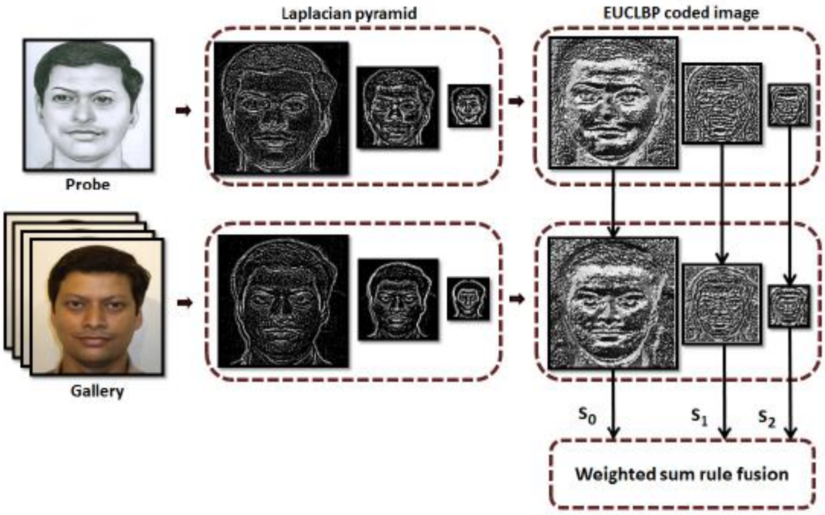

Fig. 8. Illustrating the steps involved in the EUCLBP+GA algorithm.

The sketch to digital face recognition algorithm is described as follows:

1) For a given sketch-digital face image pair, Laplacian pyramid is generated and a sequence of band pass images is obtained. This step transforms both sketches and digital images to an intermediate representation where these two different modalities can be efficiently compared. The corresponding levels of Laplacian pyramid of sketches and digital images are matched.

2) In both sketch and digital image, all the levels of Laplacian pyramid are tessellated into non-overlapping regions. On moving down the pyramid, number of regions is decreased from 8 × 8 for the first level, 7 × 7 for the second level and 6 × 6 for the lowest level. This accounts for minor registration errors for corresponding facial regions in sketches and digital images.
3) For each facial region, texture feature are computed using EUCLBP with radius R= 2 and number of neighboring pixels N= 8. The local texture descriptors corresponding to each facial region are concatenated to obtain feature vector at every level of Laplacian pyramid.
4) To match the EUCLBP features extracted from sketch and digital images at each pyramid level, texture histograms are first normalized. The weighted Chi-square distance measure is then used to compute dissimilarity between sketches and digital images. Every facial region at each Laplacian level is assigned a weight obtained using genetic optimization.
5) Finally, weighted sum rule fusion is applied to combined distance score of each Laplacian pyramid:

$$F_{fused} = \omega_0 * s_0 + \omega_1 * s_1 + \omega_2 * s_2 \qquad (22)$$

wherew_0, w_1 and w_2 are the weights assigned to different levels of Laplacian pyramid, s_0, s_1 and $s_2$are the corresponding distance scores and F_{fus} is the fused score. In the experiments, we observe thatw_0= 0.2, w_1= 0.3 and w_2= 0.5 provide the best recognition performance.
6) In identification mode (1: N), this procedure is applied for each probe and top matches are obtained based on the fused scores.

3.3 CONCLUSION

The EUCLBP+GA approach outperforms the SIFT approach described in section II. This approach gives better performance even in cases where images are not perfectly aligned. It generates a holistic description of the face image by combining histograms obtained from every local region. The algorithm utilizes the observation that sketch artists focus on structural details along with discriminating and prominent features of the face. The algorithm utilizes different levels of granularity and texture features to encode facial signatures both in sketches and digital images.

IV. Local Feature-Based Discriminant Analysis (LFDA)

In LFDA, individual representation of both sketches and photos using SIFT feature descriptors and multi scale local binary patterns (MLBP) is used. Multiple descriminent projections are then used on partitioned vectors of the feature based representation for minimum distance matching. LFDA offers substantial improvements in matching forensic sketches to the corresponding face images.

There are two different types of face sketches are present such as *Viewed sketches* and *Forensic sketches*. Viewed sketches are sketches that are drawn while viewing a photograph of the person or the person himself. Forensic sketches are drawn by interviewing a witness to gain a description of the suspect. Both forensic sketches and viewed sketches pose challenges to face recognition due to the fact that probe sketch images contain different textures compared to the gallery photographs they are being matched against. However, forensic sketches pose additional challenges due to the inability of a witness to exactly remember the appearance of a suspect and her subjective account of the description, which often results in inaccurate and incomplete forensic sketches.
Two key difficulties in matching forensic sketches:

1) Matching across image modalities
2) Performing face recognition despite possibly inaccurate depictions of the face.

In order to solve the first problem, we use local feature-based descriminent analysis (LFDA) to perform minimum distance matching between sketches and photos.

4.1 THE LFDA FRAMEWORK

In this sketch matching framework, two feature descriptors are used**:** *SIFT* and *LBP*. The SIFT feature descriptor quantizes both the spatial locations and gradient orientations within an s×s-sized image patch, and computes a histogram in which each bin corresponds to a combination of a particular spatial location and orientation. For each image pixel, the histogram bin corresponding to its quantized orientation and location is incremented by the product of 1)the magnitude of the image gradient at that pixel and 2) the value of a Gaussian function centered on the patch with a standard deviation of s=2. Trilinear interpolation is used on the quantized location of the pixel, which addresses image translation noise. The final vector of histogram values is normalized to sum to one. Here sampling SIFT feature descriptors is done from a fixed grid and do not use SIFT keypoint detection, so the SIFT feature descriptor is computed at predetermined locations. For the local binary pattern feature descriptor, extend the LBP to describe the face at multiple scales by combining the LBP descriptors computed with radii $r\epsilon\{1,3,5,7\}$ and refer to as the multi scale local binary pattern (MLBP) [2].

The choice of the MLBP and SIFT feature descriptors was based on reported success in heterogeneous face recognition and through a quantitative evaluation of their ability to discriminate between subjects in sketches and photos. SIFT feature descriptors can be successfully used for viewed sketch recognition. SIFT feature descriptors have also been shown to perform comparatively with LBP feature descriptors in a standard face recognition scenario. These feature descriptors are well-suited for sketch recognition because they describe the distribution of the direction of edges in the face; this is the information that both sketches and photos contain. By densely sampling these descriptors, sufficient discriminatory information is retained to more accurately determine a subject's identity over previously used synthesis methods:

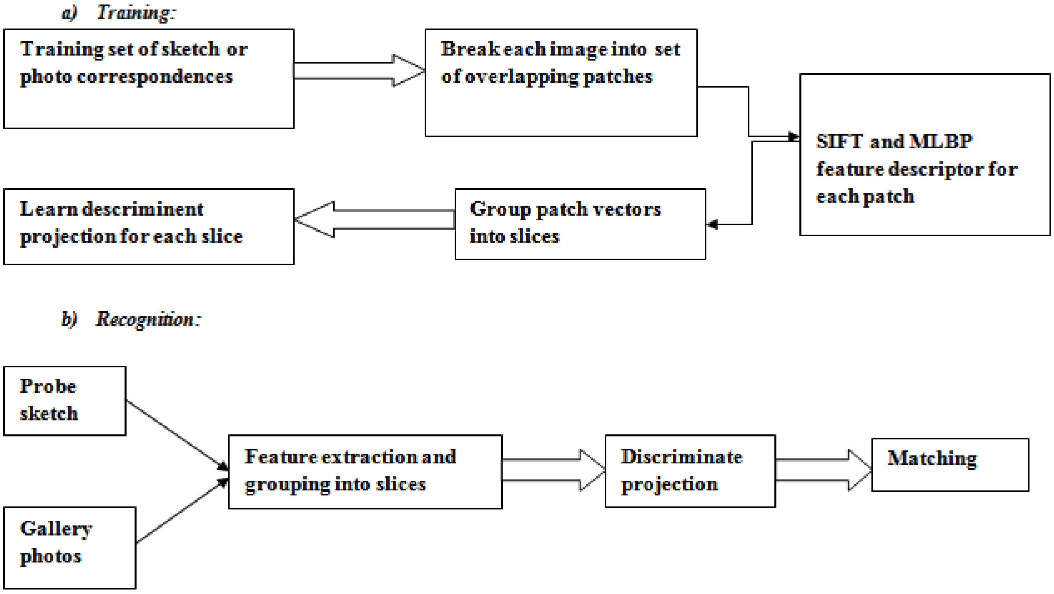
a) Training:
Training set of sketch or photo correspondences
Break each image into set of overlapping patches
SIFT and MLBP feature descriptor for each patch
Group patch vectors into slices
Learn descriminent projection for each slice
b) Recognition:
Probe sketch
Gallery photos
Feature extraction and grouping into slices
Discriminate projection
Matching

Fig. 9. *An overview of the (a) training and (b) recognition using the LFDA framework. Each sketch and photo are represented by SIFT and MLBP feature descriptors extracted from overlapping patches. After grouping "slices" of patches together into feature vectors we learn a descriminent projection for each slice. Recognition is performed after combining each projected vector slice into a single vector and measuring the normed distance between a probe sketch and a gallery photo.*

4.2 ADVANTAGES OF LFDA

LFDA has several advantages over traditional linear descriminent analysis.

1) LFDA is more effective in handling large feature vectors. The idea of segregating the feature vectors into slices allows us to work on more manageable sized data with respect to the number of training images.
2) Because the subspace dimension is fixed by the number of training subjects, when dealing with the smaller sized slices, the LFDA algorithm is able to extract a larger number of meaningful features. This is because the dimensionality of each slice subspace is bounded by the same number of subjects as a subspace on the entire feature representation would be.

4.3 ANALYSIS

In order to handle the combination of a large feature size and small sample size, an ensemble of linear discriminant classifiers called LFDA is proposed. The LFDA method is chosen because it is designed to work with a feature descriptor representation and it resulted in high recognition accuracy. LFDA was able to reduce the error in half. LFDA is able to significantly improve the recognition performance over the basic feature-based sketch matching framework.

V. WLD: A Robust Local Image Descriptor

The Weber Local Descriptor (WLD) inspired by Weber's Law, which states that the change of a stimulus that will be just noticeable is a constant ratio of the original stimulus. When the change is smaller than this constant ratio of the original stimulus, a human being would recognize it as background noise rather than a valid signal. The WLD descriptor employs the advantages of SIFT in computing the histogram using the gradient and its orientation, and those of LBP in computational efficiency and smaller support regions. But WLD distinctly differs from SIFT and LBP [3].

WLD is a dense descriptor computed for every pixel and depends on both the local intensity variation and the magnitude of the center pixel's intensity. Texture classification with WLD is carried out using 2D WLD histograms. The smaller size of the support regions for WLD makes it capture more local salient patterns. Furthermore, WLD can be easily extended to extract the multi granularity features by multi scale analysis techniques.

5.1 BASIC APPORACH

WLD consists of two components:

1) Differential excitation
2) Orientation.

The *differential excitation* component is a function of the ratio between two terms: One is the relative intensity differences of a current pixel against its neighbors; the other is the intensity of the current pixel. The *orientation* component is the gradient orientation of the current pixel. For a given image, we use the two components to construct a concatenated WLD histogram.

5.1.1 DIFFERENTIAL EXCITATION:

For calculating differential excitation $\mathcal{E}(x_c)$ of a pixel x_c first intensity differences of x_c with its neighbors x_i, $i = 1, 2, \ldots, p$ are calculated as follows:

$$\Delta I_i = I_i - I_c \qquad (23)$$

Then the ratio of total intensity difference of x_c with its neighbors x_i to the intensity of x_c is determined as follows:

$$f_{ratio} = \sum_{i=0}^{p-1} \left(\frac{\Delta I_i}{I_c}\right) \qquad (24)$$

Arctangent function is used as a filter on *Eqn*(2) to enhance the robustness of WLD against noise which results in:

$$\varepsilon(x_c) = \arctan\left[\sum_{i=0}^{p-1} \left(\frac{\Delta I_i}{I_c}\right)\right] \qquad (25)$$

The differential excitation $\mathcal{E}(x_c)$ may be positive or negative. The positive value indicates that the current pixel is darker than its surroundings and negative value means that the current pixel is lighter than the surroundings.

5.1.2 GRADIENT ORIENTATION:

For a pixel x_c the gradient orientation is calculated as follows:

$$\theta(x_c) = arctan\left[\frac{I_{73}}{I_{51}}\right] \qquad (26)$$

Where $\boldsymbol{I_{73}=I_7-I_3}$ is the intensity difference of two pixels on the left and right of the current pixel x_c, and $\boldsymbol{I_{51}=I_5-I_1}$is the intensity difference of two pixels directly below and above the current pixel, $\theta \epsilon \left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$.

The gradient orientations are quantized into T dominant orientations as:

$$\phi_t = \frac{2t}{T}\pi \; where \; t = mod\left(\left\lfloor \frac{\theta'}{2\pi/_T} + \frac{1}{2}\right\rfloor, T\right) \qquad (27)$$

Where $\theta' \epsilon [0, 2\pi]$ and is defined in terms of gradient orientation computed by Eqn (4).

In case T=8, the dominant orientations are $\phi_t = \frac{t\pi}{4}, t = 0, 1, \ldots, T-1$; all orientations located in the interval $\left[\phi_t - \left(\frac{t\pi}{4}\right), \phi_t + \left(\frac{t\pi}{4}\right)\right]$ are quantized as ϕ_t .

5.1.3 WLD HISTOGRAM:

After calculating differential excitation and dominant orientation, WLD descriptor is build. Corresponding to each dominant orientation: $t = 0, 1, 2, \ldots, T\text{-}1$ differential excitations are organized as a histogram H_t. Then each histogram H_t: $t = 0, 1, 2, \ldots, T\text{-}1$ is evenly divided into M sub histograms $H_{m,t}$: $m = 0, 1,$

2, …, M-1, each with S bins. These histograms form a histogram matrix, where each column corresponds to a dominant direction Each row of this matrix is concatenated as a histogram H_m= {H_m,t: t = 0, 1, 2, …, T-1}. Subsequently, histograms H_m: m = 0, 1, 2, …,M-1 are concatenated into a histogram H = { H_m: m = 0, 1, 2, …, M-1}. This histogram is referred to as WLD descriptor. This descriptor involves three free parameters: T, the number of dominant orientations, M the number of segments of each histogram corresponding to a dominant orientation and S, the number of bins in each segment.

5.2 COMPARISON WITH EXISTING DESCRIPTORS

Table 1: Comparison of WLD with other Descriptors.

According to the FLS Framework

:

	FILTERING	LABELING	STATISTICS
LBP	Intensity difference	Thresholding at zero	Histogram over the binary strings
SIFT	Intensity difference	Orientation quantization	Histogram over the weighted gradient on dominant orientation
WLD	Weber ratio	Orientation quantization	Histogram over the differential excitation on dominant orientation

Note that in the FLS framework, the step *filtering* depicts the inter pixel relationship in a local image region; the step *labeling* (which includes quantization and mapping) describes the intensity variations which cause psychology redundancies; the step s*tatistics* capture the attribute which is not in adjacent regions.

WLD computes the difference between the center pixel x_c and its neighbors like LBP in the first stage, these differences are added together and then divided by the center pixel x_c to obtain the differential excitation like the Weber fraction. Differently from LBP, WLD uses the gradient orientations to describe the direction of edges. The gradient orientations are then quantized to eight dominant orientations in the second stage. Differently from SIFT, WLD use the differential excitation but not the weighted gradient to compute the histogram. Here compute the frequency of the occurrence of differential excitations for each bin of the histogram.

VI. Memetically Optimized MCWLD

Multi-scale Circular WLD (MCWLD) is one of the most advanced types of image descriptors when compared to the above mentioned methods. Here WLD is optimized for matching sketches with digital face images by computing multi-scale descriptor in a circular manner. Structural information along with minute details present in local facial regions is encoded using MCWLD. An evolutionary memetic optimization is proposed to assign optimal weights to every local facial region to boost the identification performance. This automated algorithm that extracts discriminating information from local regions of both sketches and digital face images. This algorithm yields better identification performance compared to existing face recognition algorithms and two commercial face recognition systems .Here three different types of sketches are used for performance evaluation. 1) Viewed sketches, drawn by a sketch artist while looking at the digital image of a person. 2) Semi-forensic sketches, drawn by a sketch artist based on his recollection from the digital image of a person. 3) Forensic sketches, drawn based on the description of an eyewitness from his recollection of the crime scene [7].

6.1 PREPROCESSING TECHNIQUES:

A pre-processing technique is presented for enhancing the quality of forensic sketch-digital image pairs, Fig(10). Preprocessing forensic sketches enhances the quality and improves the performance by at least 2 − 3%.Thesteps involved in the pre-processing technique is as follows:

1) In order to enhance the color image, it is divided into channels as red and luma channels respectively. These are processed using Multi- scale Retinex Algorithm (MSR).
2) Then we undergoing the wavelet based denoising using Gaussian distribution based soft threshold method.
3) Noise removal in the previous step may lead to blurring of edges. The Wiener filter (symmetric LPF) is used to restores genuine facial edges.
4) After computing the globally enhanced red and luma channels, DWT fusion algorithm is applied to compute a feature rich and enhanced face image, F. Single level DWT (with db 9/7 mother wavelet) is applied to obtain the detailed and approximation bands of these images.

5) Let $f_{LL}^{j}, f_{LH}^{j}, f_{HL}^{j}, f_{HH}^{j}$ be the four subbands where j = 1, 2, LL represents the approximation band, and LH, HL, and HH represent the detailed subbands. To preserve features of both the channels, coefficients from the approximation band are averaged:

$$f_{LL}^{e} = mean(f_{LL}^{1}, f_{LL}^{2}) \quad (28)$$

Where f_{LL}^{e} is the approximation band of the enhanced image. All three detailed subbands are divided into windowsof size 3×3 and the sum of absolute pixels in each window is calculated. For the i^{th} window in HL sub band of the two images, the window with maximum absolute value is selected to be used for enhanced sub band f_{HL}^{e}. Similarly, enhanced subbandsf_{LH}^{e} and f_{HH}^{e} are also obtained.

6) Finally, inverse DWT is applied on the four subbands to generate a high quality face image:

$$F = IDWT(f_{LL}^{e}, f_{LH}^{e}, f_{HL}^{e}, f_{HH}^{e}) \quad (29)$$

This DWT fusion algorithm is applied on both forensic sketches and digital face images.

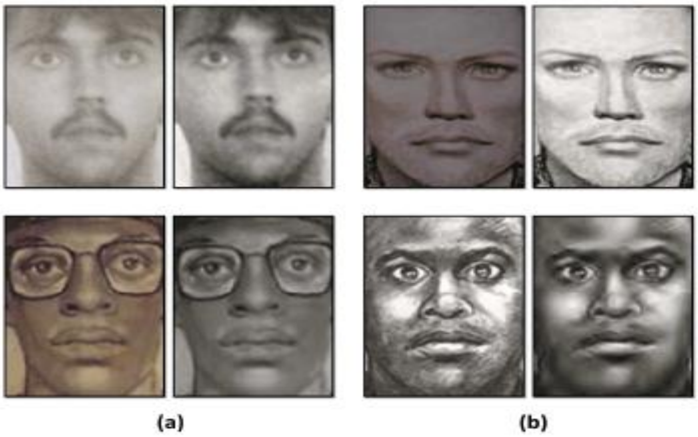

***Fig. 10**: Quality enhancement using the pre-processing technique. (a) represents digital face image before and after pre-processing and (b) represents forensic sketches before and after pre-processing.*

6.2 FEATURE EXTRACTION USING MCWLD:

MCWLD has two components: 1) *differential excitation* and 2) *gradient orientation*. MCWLD representation for a given face image is constructed by tessellating the face image and computing a descriptor for each region. As shown in Fig11, MCWLD descriptor is computed for different values of parameters P and R, where P is the number of neighboring pixels evenly separated on a circle of radius R centered at the current pixel. Multi-scale analysis is performed by varying the radius R and number of neighbors P.

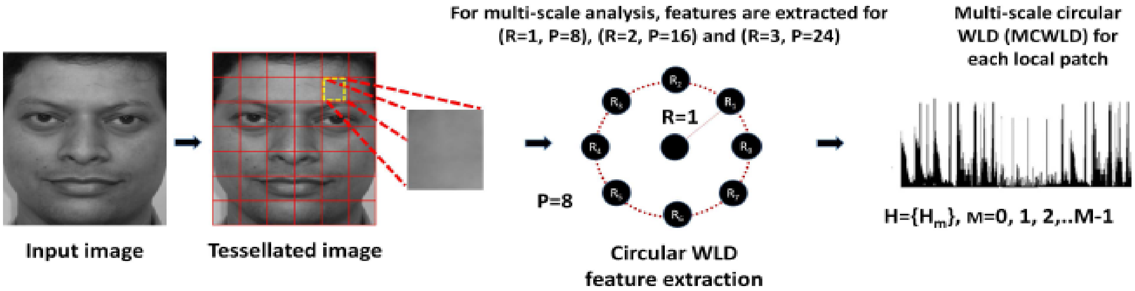
For multi-scale analysis, features are extracted for (R=1, P=8), (R=2, P=16) and (R=3, P=24)
Multi-scale circular WLD (MCWLD) for each local patch
R=1
P=8
H={H_m}, M=0, 1, 2,..M-1
Input image
Tessellated image
Circular WLD feature extraction

***Fig.11:** Steps involved in the algorithm for matching sketches with digital face images.*

6.2.1 DIFFERENTIAL EXCITATION:

Differential excitation is computed as an arctangent function of the ratio of intensity difference between the central pixel and its neighbors to the intensity of central pixel. The differential excitation of central Pixel $\mathcal{E}(x_c)$ is computed as:

$$\xi(x_c) = arctan \left\{\sum_{i=0}^{P-1}\left(\frac{x_i - x_c}{x_c}\right)\right\} \qquad (30)$$

where x_c is the intensity value of central pixel and P is the number of neighbors on a circle of radius R. If $\mathcal{E}(x_c)$ is positive, it simulates the case that surroundings are lighter than the current pixel. In contrast, if $\mathcal{E}(x_c)$ is negative, it simulates the case that surroundings are darker than the current pixel.

6.2.2 ORIENTATION:

The orientation component of WLD is computed as:

$$\theta(x_c) = \arctan\left\{\frac{x_{\left(\frac{P}{2}+R\right)} - x_{(R)}}{x_{(P-R)} - x_{\left(\frac{P}{2}-R\right)}}\right\} \qquad (31)$$

The orientation is further quantized into T dominant orientation bins.

6.2.3 CIRCULAR WLD HISTOGRAM:

A 2D histogram of circular WLD feature $CWLD\left(\xi_j, \bar{\theta}_t\right)$ is constructed where j = 0, 1... N−1, t = 0, 1... T −1 and N is the dimension of the image. Each column in the 2D histogram corresponds to a dominant orientation θ_t and each row corresponds to a differential excitation interval. A *four step approach* is followed to compute CWLD descriptor.

Step-1: The 2D histogram $CWLD\left(\xi_j, \bar{\theta}_t\right)$ is further encoded into1D histograms. Differential excitations, $\mathcal{E}$, are regrouped into T orientation sub-histograms, H(t), where t = 0, 1, ..., T − 1 corresponds to each dominant orientation.

Step-2: Within each dominant orientation, range of differential excitation is evenly divided into M intervals and then reorganized into a histogram matrix. Each orientation sub histogram in H(t) is thus divided into M segments, $H_{m,t}$
where m = 0, 1, ...,M –1.Each sub-histogram segment $H_{m,t}$ is further composed of S bins and is represented as:

$$H_{m,t} = h_{m,t,s} \qquad (32)$$

where s = 0, 1, ..., S –1.

Step-3: Sub-histogram segments, $H_{m,t}$, across all dominant orientations are reorganized into M one dimensional histograms.

Step-4: M sub-histograms are concatenated into a single histogram thus representing the final M × T × S circular WLD histogram. The range of differential excitation is segmented into separate intervals to account for the variations in a given face image, and assigning optimal weights to these H_m segments further improves the performance of CWLD descriptor.

6.2.4*MULTI-SCALE CIRCULAR WLD:*

In Multi-scale analysis, CWLD descriptor is extracted with different values of P and R and the histograms obtained at different scales are concatenated. A face image is divided into non-overlapping local facial regions and MCWLD histogram is computed for each region. MCWLD histograms for every region are then concatenated to form the facial representation.

6.3MEMETIC OPTIMIZATION:

The combination of evolutionary algorithms with local search was named "memetic algorithms" (MAs). In the case of MA's, "memes" refer to the strategies (e.g., local refinement, perturbation, or constructive methods, etc.) that are employed to improve individuals. Optimizing large number of weights for best performance is a very challenging problem and requires a learning based technique. Memetic algorithm can be effectively used to optimize such large search spaces. From an optimization perspective, MAs have been found to be more efficient and effective than traditional evolutionary approaches such as GA.

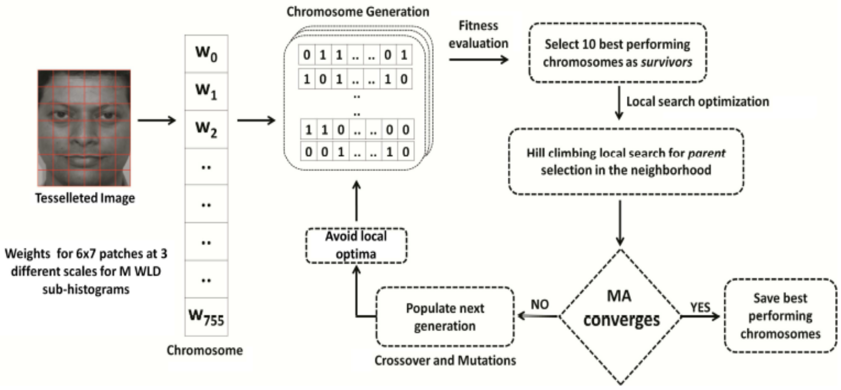
Tesselleted Image
Weights for 6x7 patches at 3 different scales for M WLD sub-histograms
w0
w1
w2
..
..
..
..
w755
Chromosome
Chromosome Generation
0 1 1 .. .. 0 1
1 0 1 .. .. 1 0
1 1 0 .. .. 0 0
0 0 1 .. .. 1 0
Fitness evaluation
Select 10 best performing chromosomes as survivors
Local search optimization
Hill climbing local search for parent selection in the neighborhood
MA converges
YES
Save best performing chromosomes
NO
Populate next generation
Crossover and Mutations
Avoid local optima

Fig.12. *Illustrating the steps involved in memetic optimization for assigning optimal weights to each tessellated face region.*

ALGORITHM: Memetic algorithm for weight optimization

Step 1: *Memetic Encoding*: A chromosome of length 42×3×6=756 is encoded where each unit in the chromosome is a real valued number representing the corresponding weight.

Step 2: *Initial Population*: A population of 100 chromosomes is generated starting with a seed chromosome.

Step 3: *Fitness Function*: Fitness is evaluated by performing recognition using the weights encoded by each chromosome. 10 best performing chromosomes from a population are selected as *survivors* to perform crossover and mutation.

Step 4: *Hill Climbing Local Search*: The *survivors* obtained in Step 3 are used to find better chromosomes in their local neighborhood and *parents* are selected.

Step 5: *Crossover and Mutation*: New population is generated from *parents* obtained after local search in Step 4. A set of uniform crossover operations is performed followed by mutation. To avoid local optima, adaptive mutation and random offspring generation techniques are used.

Step 6: Repeat Steps 3-5 till convergence criteria is satisfied.

6.4 SUMMARY:

The process of matching sketches with digital face images is as follows:
1) For a given sketch-digital image pair, the pre-processing technique is used to enhance the quality of face images.
2) Both sketches and digital face images are tessellated into non-overlapping local facial regions.
3) For each facial region, MCWLD histograms are computed at three different scales. The facial representation is obtained by concatenating MCWLD histograms for every facial region.
4) To match two MCWLD histograms, weighted chi-square distance measure is used where the weights are optimized using Memetic algorithm.
5) In identification mode, this procedure is applied for each gallery-probe pair and top matches are obtained.

VII. COMPARISONS

The performance of the Memetically optimized MCWLD is compared with existing algorithms designed for matching sketches with digital face images and two leading commercial face recognition systems (COTS-1, COTS2). Existing algorithms include SIFT, EU-CLBP+GA, and LFDA. Further, the performance gain due to multi-scale analysis and circular sampling is analyzed by comparing the performance of WLD, Multi-scale WLD (MWLD) algorithms with square sampling, and Multiscale circular WLD (MCWLD). The same weighting scheme is used in WLD, MWLD, and MCWLD algorithms. Further, to quantify the improvement due to memetic optimization of weights as compared to the weighting method proposed in WLD, the performance of the algorithm is compared with MCWLD. The preprocessing technique enhances the quality only when there are irregularities and noise in the input image and it does not alter good quality face images.

The details of comparison on the basis of accuracy and standard deviation of various methods is shown in table 2.

Results available are:

1) The proposed approach yields rank-1 accuracy of 93.16% which is slightly better than LFDA and is at least 2% better than MWLD, MCWLD, SIFT, and EUCLBP+GA. The proposed approach also outperforms the two commercial systems by at least 5%.
2) On comparing WLD with MWLD, it is observed that MWLD provides an improvement of about 1% on different viewed sketch databases due to multi-scale analysis. Further, compared to MWLD algorithm, the proposed MCWLD algorithm improves the rank-1 identification accuracy by 2.9%. It suggests that circular sampling method yields more discriminative representation of the face image as compared to square sampling.

Table 2: *Rank-1 identification accuracy of sketch to digital face image matching algorithms for matching viewed Sketches.*

ALGORITHM	RANK-1 IDENTIFICATION ACCURACY(%)	STANDARD DEVIATION(%)
COTS-1	80.14	0.78
COTS-2	79.24	0.86
WLD	84.37	0.88
MWLD	85.32	0.86
MCWLD	88.25	0.84
SIFT	85.86	1.01
EUCLBP+GA	88.75	0.87
LFDA	91.16	0.93
Proposed	93.16	0.96

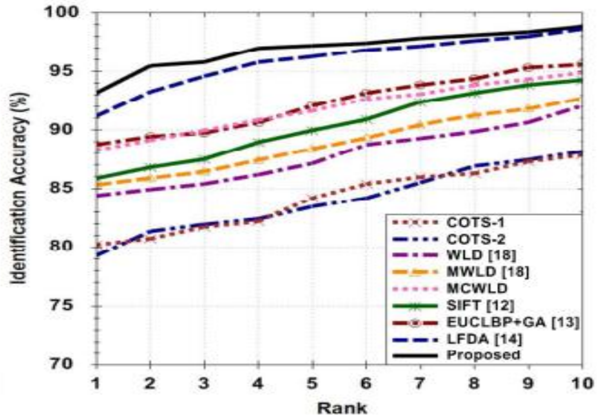

Fig.13. *CMC (Cumulative Match Characteristic) curves showing the performance of sketch to digital face image matching algorithms.*

Local descriptors have received attention in face recognition due to their robustness to scale, orientation, and speed. Local descriptors such as LBP are generally used as dense descriptors where texture features are computed for every pixel of the input face image. On the other hand, sparse descriptor such as Scale Invariant Feature Transform (SIFT) is based on interest point detection and computing the descriptor in the vicinity of detected interest points. SIFT is computed using the gradient and orientation of neighboring points sampled around every detected key point. The Weber's local descriptor (WLD) draws its motivation from both SIFT and LBP. It is similar to SIFT in computing histogram using gradient and orientation, and analogous to LBP in being computationally efficient and considering small neighborhood regions. However, WLD has some unique features that make it more efficient and robust as compared to SIFT and LBP. WLD computes the salient micro patterns in a relatively small neighborhood region with finer granularity. This allows it to encode more discriminative local micro patterns.WLD is optimized for matching sketches with digital face images by computing multi-scale descriptor in a circular manner is by MCWLD. The MCWLD and memetically optimized weighted chi-square distance outperforms existing algorithms such as SIFT, EUCLBP+GA, and LFDA by 2 − 5%. It also outperforms the two commercial face recognition systems by at least 9%.

VIII. Conclusions

Sketch to digital face matching is an important research challenge and is very pertinent to law enforcement agencies. Comprehensive analysis, including comparison with existing algorithms and two commercial face recognition systems, is performed using the viewed, semi forensic, and forensic sketch databases. Semi-forensic sketches are introduced to bridge the gap between viewed and forensic sketches. Analysis of results also suggests that local regions play an important role in matching sketch digital image pairs and is effectively encoded in MCWLD and memetically optimized weighted Chi-square distance. The results also show that the proposed algorithm is significantly better than existing approaches and commercial systems.

References

[1] Himanshu S .Bhatt, Samarth Bharadwaj, Richa Singh, Mayank Vatsa, “Memetically Optimized MCWLD for Matching Sketches with Digital Face Images.” Information Forensics and Security, IEEE Transactions on Volume:7 Issue:5,2012.

[2] B. Klare, L. Zhifeng, and A. Jain, “Matching forensic sketches to mug shot photos”, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 33, pp. 639–646, 2011.

[3] J. Chen, S. Shan, C. He, G. Zhao, M. Pietikainen, X. Chen, and W. Gao,“WLD - A robust local image descriptor”, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 32, no. 9, pp. 1705–1720, 2010.

[4] H. Bhatt, S. Bharadwaj, R. Singh, and M. Vatsa, “On matching sketches with digital face images”, in Proceedings of International Conference on Biometrics: Theory Applications and Systems, 2010.

[5] C. Geng and X. Jiang, “Face recognition using SIFT features”, in Proceedings of International Conference on Image Processing, 2009,pp. 3313–3316.

[6] T. Ahonen, A.Hadid, and M. Pietikainen, “Face description with local binary patterns: Application to face recognition”, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 28, no. 12, pp. 2037–2041, 2006.

[7] N. Krasnogor and J. Smith, “A tutorial for competent memetic algorithms: model, taxonomy, and design issues”, IEEE Transactions on Evolutionary Computation, vol. 9, no. 5, pp. 474 – 488, 2005.