A FRAMEWORK FOR FACE RECOGNITION USING ADAPTIVE BINNING AND ADABOOST TECHNIQUES

Srinivasan A[1]

[1]Department of Computer Science and Engineering, Misrimal Navajee Munoth Jain Engineering College, Chennai, Tamilnadu, India
asrini30@gmail.com

ABSTRACT

In this paper, a novel framework for face recognition is developed by using adaptive binning and adaboost technique. Adaptive binning is an efficient classifier technique to classify the object and the results are represented in Histogram Gabor Phase Pattern [HGPP]. The resultant HGPP is again applied with an adaboost classification technique to improve the efficiency of the pattern by further reducing the computational complexity. This new framework is experimentally verified with FERET and found that the recognition rate of the system is improved. The main feature of this system is a unified model for assessing all the probe sets of the face images and best results are thus achieved.

KEYWORDS

Face Recognition, Adaptive Binning, Adaboost, Classifiers, Histogram

1. INTRODUCTION

Face recognition is a natural and straightforward biometric method used by us to identify one another. Face recognition is a recognition process that analyzes facial characteristics of a person [1, 2]. The recent interest in face recognition can be attributed to the use of latest techniques in security and surveillances and many other commercial interests. People look for more secure methods to protect their valuable information. Password authentication, card key authentication, and biometric authentication are the most commonly used authentication types.

Face detection is an essential tool for face recognition system [3]. Face detection locates and segments face regions from cluttered images obtained from still images. It has numerous applications such as surveillance, security control systems, content based image retrieval, video conferencing, and intelligent human computer interfaces. Most of the current face recognition systems presume that faces are readily available for processing. However, one can not get typical images with just faces. The corollary is that a system that will segment faces into cluttered images is needed. With such a portable system, one can ask the user to pose for the face identification task. In addition to creating a more cooperative target, one can also interact with the system in order to improve and monitor face detection. With a portable system, detection seems easier.

The task of face detection is seemingly trivial for the human brain, but it still remains a challenging and difficult problem to enable a computer or mobile phone or PDA to do the same. This is because the human face changes with respect to the internal factors such as facial expression, occlusion etc. And, it is also affected by the external factors such as scale, lightning conditions, contrast between faces, and background and orientation of faces.

DOI : 10.5121/ijma.2011.3107

1.1. Need for Face Detection

A facial recognition system is computer applications which automatically identify or verify a person from a digital image or a video frame from a video source [4]. One of the ways to do this is by comparing selected facial features of the image and a facial database. It is typically used in security systems and other biometrics such as fingerprint or eye iris recognition systems.

There are number of potential uses for facial recognition that are currently being developed. For example, the technology could be used as a security measure at ATM's, instead of using a bank card or personal identification number, the ATM would capture an image of your face, and compare it to your photo in the bank database to confirm your identity. The same concept could also be applied to computers, by using a webcam to capture a digital image of yourself and your face could replace your password as a means to log-in in the system.

A face recognition system has a lot of commercial, military, security and research applications. Some of them are

- Checking for criminal records.
- Enhancement of security by using surveillance cameras in conjunction with face recognition system.
- Knowing in advance, if some VIP is entering the hotel.
- Detection of a criminal at public place.
- Can be used in different areas of science for comparing an entity with a group of entities.
- Pattern Recognition.

1.1. Related Works

Many facial recognition algorithms identify faces by extracting landmarks, or features, from an image of the face. For example, an algorithm may analyze the relative position, size, shape of the eyes, nose, cheekbones, and jaw. These features are then used to search for other images with matching features. A probe image is then compared with the face data stored in the database. One of the earliest, successful systems is based on template matching techniques applied to a set of salient facial features.

Recognition algorithms can be divided into two main approaches, geometric, which looks for distinguishing features or photometric, which is a statistical approach that distil an image into values and comparing the values with templates to eliminate variances. Popular recognition algorithms include Principal Component Analysis [5], Linear Discriminate Analysis [6], Elastic Bunch Graph Matching Fisher faces [7], and the Hidden Markov model [8].

The rest of the paper presents the architecture and the design of the new framework. Section 2 presents the image processing features adopted in this framework. Section 3 explains the boosting techniques and its summary. Section 4 reveals the results and Section 5 concludes the paper with future work.

2. Face Recognition Framework

2.1. Gabor Wavelets

Gabor wavelet is used as Gabor filter [9]. A set of filtered images are obtained by convolving the given image with Gabor filters. Each of these images represents the image information at a certain frequency and orientation. From each filtered image, Gabor features can be calculated and used to retrieve images. Here, Gabor transformation is applied to the normalized faces using the equations shown below

$$\Psi_{u,v}(z) = \frac{\left(\|K_{u,v}\|\right)^2}{\sigma^2} e^{\frac{\left(-\|K_{u,v}\|\|Z\|\right)^2}{2\sigma^2}} \left[e^{iK_{u,v}Z} - e^{-\frac{\sigma^2}{2}} \right]$$

where $K_{u,v} = \begin{pmatrix} K_{ju} \\ K_{jv} \end{pmatrix} = \begin{pmatrix} K_v \cos\theta_u \\ K_v \sin\theta_u \end{pmatrix}$ and $K_v = \frac{f_{max}}{2^{v/2}}, \sigma = 2\pi,\ \theta_u = u\left(\frac{\pi}{8}\right)\ and\ Z = image\ position$

'v' is the frequency and 'u' is the orientation with v_{max} = 5 and u_{max} = 8, v = 0…..v_{max}-1, u = 0…. u_{max}–1. The Gabor transformation of a given image is defined as its convolution with the Gabor function: $G_{u,v}(z) = I(z) * \Psi_{u,v}(z)$ where $z = (x, y)$ denotes the image position, the symbol "*" denotes the convolution operator, and $G_{u,v}(z)$ is the convolution result corresponding to the Gabor kernel at scale and orientation . The Gabor wavelet coefficient is a complex function which can be rewritten as $G_{u,v}(z) = A_{u,v}(z).\exp\,(i\theta_{u,v}(z))$. The magnitude and phase part is represented accordingly by $A_{u,v}(z)$ and $\exp\,(i\theta_{u,v}(z)$.

2.2. Daugman's Method

Daugman's Method is used for phase quadrant demodulation coding of Gabor phase. The output of Gabor Wavelets is demodulated and each pixel in the resultant image is encoded to two bits [10]. This method is essential to split the Gabor Wavelets Pattern to Global Gabor Phase Pattern (GGPP0 and Local Gabor Phase Pattern (LGPP) . For separation the below equations are used.

$$P_{u,v}^{Re}(z) = \begin{Bmatrix} 0\ if\ Re(G_{u,v}(z)) > 0 \\ 1\ if\ Re\left(G_{u,v}(z)\right) \le 0 \end{Bmatrix} \text{ and } P_{u,v}^{Im}(z) = \begin{Bmatrix} 0\ if\ Im(G_{u,v}(z)) > 0 \\ 1\ if\ Im\left(G_{u,v}(z)\right) \le 0 \end{Bmatrix}$$

Here, the real and imaginary parts of Gabor coefficient are denoted as $Re(G_{u,v}(z))$. Daugman's encoding method can be reformulated by using equation shown below.

$$P_{u,v}^{Re}(z) = \begin{Bmatrix} 0\ if\ \theta_{u,v}\ (z) \epsilon\ I, IV \\ 1\ if\ \theta_{u,v}(z)\ \epsilon\ II, III \end{Bmatrix} \text{ and } P_{u,v}^{Im}(z) = \begin{Bmatrix} 0\ if\ \theta_{u,v}\ (z) \epsilon\ I, IV \\ 1\ if\ \theta_{u,v}\ (z) \epsilon\ II, III \end{Bmatrix}$$

Here, $\theta_{u,v}(z)$ is the Gabor phase angle for a pixel. Quadrant bit coding (QBC) assigns two bits for each pixel according to the quadrant in which the Gabor phase angle lies. QBC is relatively stable and it is actually the quantification of Gabor feature.

2.3. Global Gabor Phase Pattern

For a given frequency GGPP scheme computes one binary string for each pixel by concatenating the real and imaginary bit codes of different orientations. The GGPP value, $GGPP_v(Z_o)$for the frequency 'v' at the position Z_o in a given image is formulated as the combination of Daugman's Values by using functions shown below.

$$GGPP_v^{Re}(z_0) = \left[P_{0,v}^{Re}(z_0), P_{1,v}^{Re}(z_0) \dots\dots\dots P_{k,v}^{Re}(z_0)\right] \text{ and}$$
$$GGPP_v^{Im}(z_0) = \left[P_{0,v}^{Im}(z_0), P_{1,v}^{Im}(z_0) \dots\dots\dots P_{k,v}^{Im}(z_0)\right]$$

It is experimented with eight orientations with k = 0..7 which forms a byte, representing 256 different orientation modes. These binary values are converted to decimal values by using functions given below.

$$GGPP_v^{Re}(z_0) = \left[P_{0,v}^{Re}(z_0) * 2^k + P_{1,v}^{Re}(z_0) * 2^{k-1} \dots\dots\dots P_{k,v}^{Re}(z_0)\right]$$

$$GGPP_v^{Im}(z_0) = \left[P_{0,v}^{Im}(z_0)\ * 2^k + P_{1,v}^{Im}(z_0) * 2^{k-1} \dots\dots\dots P_{k,v}^{Im}(z_0)\right]$$

By using this encoding method, one can get two decimal numbers for each pixel corresponding to the real and imaginary GGPPs. Both of them range in [0, 255], and it is easy to visualize them as the grey-level images for a given frequency.

2.4. Local Gabor Phase Pattern

In this scheme, the local variations for each pixel are encoded. For each orientation and frequency the real and imaginary parts of Local Gabor Phase Pattern (LGPP) value is computed by using local XOR pattern (LXP) operator [10, 11]. LGGP actually encodes the sign difference of the central pixel from its neighbours. LGPP reveals the spots and flat area in the given images.

For each orientation 'u' and frequency 'v', the real and imaginary LGPP value at the position is computed by using the following equation named local XOR pattern (LXP) operator.

$$LGPP_{u,v}^{Re}(Z_0) = P_{u,v}^{Re}(Z_0) \oplus P_{u,v}^{Re}(Z_1), P_{u,v}^{Re}(Z_0) \oplus P_{u,v}^{Re}(Z_2), \dots \dots \dots \dots \dots P_{u,v}^{Re}(Z_0) \oplus P_{u,v}^{Re}(Z_7)$$
$$LGPP_{u,v}^{Im}(Z_0) = P_{u,v}^{Im}(Z_0) \oplus P_{u,v}^{Im}(Z_1), P_{u,v}^{Im}(Z_0) \oplus P_{u,v}^{Im}(Z_2), \dots \dots \dots \dots \dots P_{u,v}^{Im}(Z_0) \oplus P_{u,v}^{Im}(Z_7)$$

Thus from the definition of QBC each bit is computed as follows

$$P_{u,v}^{Re}(Z_o) \oplus P_{u,v}^{Re}(Z_i) = \begin{cases} 0, if\ Re(G_{u,v}(Z_0) * Re(G_{u,v}(Z_i) > 0 \\ 1, if\ Re(G_{u,v}(Z_0) * Re(G_{u,v}(Z_i) \le 0 \end{cases}$$

$$P_{u,v}^{Im}(Z_o) \oplus P_{u,v}^{Im}(Z_i) = \begin{cases} 0, if\ Im(G_{u,v}(Z_0) * Im(G_{u,v}(Z_i) > 0 \\ 1, if\ Im(G_{u,v}(Z_0) * Im(G_{u,v}(Z_i) \le 0 \end{cases}$$

LGPP actually encodes the sign difference of the central pixel from its neighbors. Therefore, LGPP can also reveal the spots to ("111111111"), flat areas to ("00000000"), for binary images. Similar to GGPP, eight neighbors provide 8 bits to form a byte for each pixel. Therefore, a decimal number ranging from 0 to 255 is computed.

2.5. Spatial Histogram

Object representation and feature extraction are essential to object detection. Specially, objects are modelled by their spatial histograms over local patches and class specific features are extracted. Spatial histograms consist of marginal distributions of an image over local patches and they can preserve texture and shape information of an object simultaneously. The obtained GGP and LGP patterns are relatively new and simple texture model serving very as powerful feature in classifying the images [12, 13]. They are invariant against any monotonic transformation of the gray scale and they use their neighbourhood intensities to calculate the 3x3 region central pixel value using the equation given below.

$$S_{(g_0,g_i)} = \begin{cases} 1, if\ g_i \ge g_0 \\ 0, if\ g_i \ge g_0 \end{cases} where\ i \le i \le 8$$

The signs of the eight differences are encoded into an 8-bit number to obtain LGPP value of the centre pixel by using the equation$\sum_{i=1}^{8} S(g_0, g_i)\ 2^{i-1}$. For any sample image, histogram-based pattern representation is computed as follows. First, variance normalization on the gray image to compensate the effect of different lighting conditions are applied. Then, the basic global or local binary pattern operator is used to transform the image into an GGPP or LGPP image and compute histogram of an image as representation finally. It is easy to prove that histogram, a global representation of image pattern, is invariant to translation and rotation [14].

2.6. Histogram of Gabor Phase Patterns

Baochang Zhang. et.al defined a descriptor called Histogram of Gabor phase pattern (HGPP) for robust face recognition [15]. This approach is based on the combination of the spatial histogram and the Gabor phase information encoding scheme, the Gabor phase pattern (GPP). Different from the learning-based face recognition methods, in HGPP, features are directly extracted without the training procedure. Two kinds of GPPs, GGPP and LGPP, are used to capture the phase variations derived from the orientation changing of Gabor wavelet and the relationships among local neighbours. Both are divided into small non overlapping rectangular regions, from which the local histograms are extracted and concatenated into a single extended histogram feature to capture the spatial information. The method captures both phase and magnitude information of Gabor transformation.

3. EXPERIMENTS

3.1. Block Diagram

The Figure 1 shows the entire block design of the new system with new methodology.

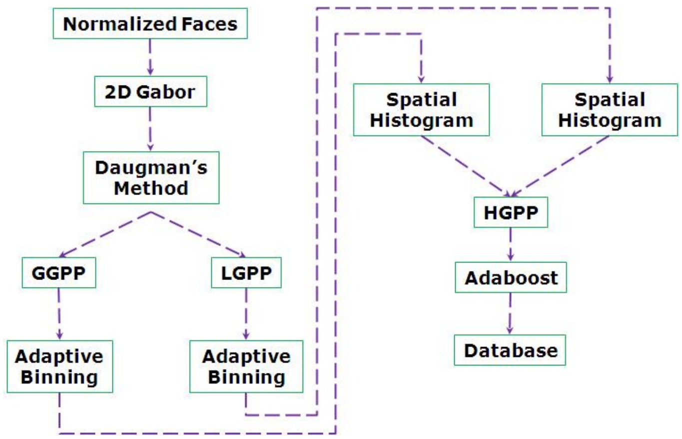


Figure 1: Block diagram for Adaboost

The Normalized face is given as input to the four different processes i) Gabor filters ii) daugman's method, iii) GGPP and iv) LGPP. For the Adaptive binning, the result obtained by GGPP and LGPP is given as the input. This method works by creating a bin of size 3x3 and the result value obtained by adaptive binning is given to Spatial Histogram and the output of spatial histogram is used to create HGPP. To further reduce the date set, a adaboost classifier is used.

3.2. Adaptive Binning

Adaptive binning is the simplest algorithm to reduce the feature space set obtained from two GGP's [14]. It is an attempt to adaptively bin a single image based on the number of pixels in each region. The basic method is to bin pixels in two dimensions by a factor of two, until the fractional Poisson error of the count in each bin becomes less than or equal to a threshold value. When the error is below this value, those pixels are not binned any further [16, 17].

3.2.1. Adaptive Binning Algorithm

Step 1: Put each pixel in a 'bin', which is a collection of pixels.
Step 2: The net count in the bin is defined by $s_i = c_i - n_i b$
Step 3: Fractional error in the bin is calculated as $\frac{\sigma(s_i)}{s_i} = \frac{\sqrt{c_i + n_i b}}{c_i - n_i b}$
Step 4: Find aveage mean count s_i/n
Fractional error <= threshold value bin processed, find average mean count else bin not yet processed
Step 5: Set Identification number for each processed bin
Step 6: Merge the neighboring bins.
Step 7: Repeat from Step 2 until a single bin is got

3.3. AdaBoost

Boosting is a general method for improving the accuracy of any given learning algorithm. Boosting refers to a general and provably effective method of producing a very accurate prediction rule by combining rough and moderately inaccurate rules of thumb. AdaBoost algorithm has undergone intense theoretical studies and empirical testing. The AdaBoost algorithm, introduced in 1995 by Freund and Schapire, solved many of the practical difficulties of the earlier boosting algorithms. So, it is used in this framework.

The concept of adaboost was first incorporated to HGPP, wherein quick classifications of features were done [18, 19]. The advantages of using a classifier are i) it is fast, simple and easy to program, ii) it has no parameters to tune (except for the number of rounds), iii) it requires no prior knowledge about the weak learner. So, it can be flexibly combined with any method for finding weak hypotheses. An AdaBoost classifier is a set of weak classifier and classification is done by increasing the weights of the incorrectly classified sets. A Final strong classifier is obtained by sum of weights of weak classifiers. The test image is checked with these function weights and if the resultant value is true for more than 50% then the image is classified as corresponding to the person's face and declared recognized.

In our algorithm, training sample for Adaboost learning is generated by subtracting test and trained HGPP features of face images. If the sample generated belongs to the same person, the generated sample is positive and vice versa. In an adaboost every iteration train a set of weak classifiers on each dimension of HGPP features. Hence a considerable amount of HGPP features are reduced, grouped, and classified. Adaboost uses CHI difference (i.e the subtraction of HI (HGPP feature for test image) form HP (HGPP feature for registered image) and the difference is given to a binary classifier for finding out whether a judgment is true or false. This is done for all features of HGPP and if more than half of the values are true then the face is identified as true and found recognized.

3.3.1. Adaboost Algorithm

Step 1: Based on the HGPP result both for the test and train set

$HGPP_i = [h_{i1}, h_{i2}, h_{i3}, \ldots\ldots h_{iQ}]$

where h_{ij} is a histogram, j=1,2,3………Q and Q is the number of HGPP file in a train set

Step 2: CHI distance is calculated between the test set and each image in a train set

Step 3: $HGPP_i - HGPP_j = [CHI(h_{i1},h_{j1}), CHI(h_{i2},h_{j2}), \ldots\ldots CHI(h_{iQ},h_{jQ})]$

Step 4: for i ranging from 1 to n

for j ranging from 1 to Q

CHI = ((Test[j] – train[i][j])^2 / Test[j] – train[i][j])

where n is the number of images in a train set

end for j

end for i

Step 5: if(CHI == 0)

Images are recognized

else

Images are not recognized

The algorithm takes training set and test set image as input. It takes the HGPP value of the test and train images and finds the CHI difference. When the CHI value equals to 0 then the image is recognized, otherwise not. The main advantage of using adaboost algorithm is its reduced computational time because it corresponds only to linear programming.

4. RESULTS AND ANALYSIS

A new framework is developed and presented by using a normalized image as an input. Gabor wavelets, which are directly related to Gabor filter, is a linear filter used for edge detection. A set of Gabor filers with different frequencies and orientations are helpful for extracting useful features from an image. In this system, frequency value is set to five and orientation to eight. Gabor filters has a real and an imaginary component representing orthogonal directions. As a result, 80 (40 real and 40 imaginary) different sets of Gabor filters in total are obtained from a single image. These filters are further processed to demodulate the image by using Daugman's method. And, all the eighty images are demodulated to obtain quantified Gabor feature. After quantifying the Gabor features using Daugman's method, global Gabor phase patterns are generated to form a byte to represent 256 different orientation modes. In GGPP, totally 10 (5 real and 5 imaginary) images are obtained. To encode the local variations in a pixel, LGPP is applied to all eighty Gabor features using local XOR pattern. For five frequency and eight orientations, the phase patterns obtained will be 90 "images" (five real GGPP, five imaginary GGPP, 40 real LGPPs and 40 imaginary LGPPs), with the same size as the original image. To reduce the size of phase patterns, binning is done by using adaptive binning technique. Each phase patterns are taken and binned (size 3x3) in such a manner that the count in each bin becomes less than or equal to the threshold value. To reserve the spatial information in the phase patterns, the GPP and LGP images are spatially divided into the non over-lapping rectangular regions, from which the spatial histograms are extracted. Then, all of these histograms are concatenated into a single extended histogram feature, the so-called HGPP. The HGPP feature is formulated as $HGPP = (H^{Re}_{GGPP}, H^{Im}_{GGPP}, H^{Re}_{LGPP}, H^{Im}_{LGPP})$

where $H^{Re}_{GGPP}, H^{Im}_{GGPP}$ are the sub regions of real and imaginary part of GGPP,

$H^{Re}_{LGPP}, H^{Im}_{LGPP}$ are the sub regions of real and imaginary part of LGPP.

Final HGPP representation is a local model which is robust to local distortions, caused by different imaging factors such as accessory and expression variations. To further boost the efficiency of the framework, another classification technique called Adaboost is used.

4.1. Input Image Database

The image database is developed into 3 sizes such as 64x64, 88x88 and 128x128. In each size of the image, these images are categorized into 15 parts as shown in Table 1.

Table 1: Classification of Image Database

Probe Set	Description
Aging	Aging of subject
Dup I	Duplicate I of Aging
Dup II	Subset of Dup I
fa	regular frontal image
fb	alternative frontal image, taken shortly after the corresponding fa image
fc	Illumination
pl	profile left
hl	half left - head turned about 67.5 degrees left
pr	profile right
hr	half right - head turned about 67.5 degrees right
ra	random image - head turned about 45 degree left
rb	random image - head turned about 15 degree left

rc	random image - head turned about 15 degree right
rd	random image - head turned about 45 degree right
Re	random image - head turned about 75 degree right

Table 2 gives the time taken for training the train set for three different sizes of images viz. 64x64, 88x88 and 128x128. It can be seen from the table that training time is increased according to image size and the time complexity increased with size.

Table 2: Training Time for 3 image sizes

Training Time	64x64	2.21 Minutes
	88x88	3.45 Minutes
	128x128	4.55 Minutes

The recognition rate and time taken to recognize an image are tabulated in Table 3 for Adaptive binning method and for probe sets which include aging, dup I, dup II, frontal images (fa), expression images (fb) and illuminated image (fc). It can be observed from the table that the processing time increases when the size of image increases and irrespective of image probe set, particularly for frontal images (fa), the processing time goes to peak values i.e. 80, 100 and 230 seconds for three different sized images. The Table 3.1 tabulates the recognition rate and time for probe sets which include images turned right and left (hr and hl), with their profile left and right (pl and pr), images turned randomly (ra, rb, rc, rd and re).

Table 3: Comparison chart for different image sizes and probe sets I for Adaptive Binning Method

Size	Probe Set ->	Aging	Dup I	Dup II	Fa	Fb	Fc
64x64	Recognition Rate [%]	92	98	97	99	100	99
	Processing Time[Sec.]	75	60	65	80	65	60
88x88	Recognition Rate [%]	93	98	98	100	98	99
	Processing Time[Sec.]	80	80	75	100	98	75
128x128	Recognition Rate [%]	91	97	98	98	98	99
	Processing Time[Sec.]	210	210	205	230	228	205

Table 3.1: Comparison chart for different image sizes and probe sets II for Adaptive Binning Method

Size	Probe Set ->	hr	hl	pr	pl	ra	rb	rc	rd	re
64x64	Recognition Rate [%]	99	98	98	99	98	98	97	99	99
	Processing Time[Sec.]	85	80	65	60	60	60	80	80	85
88x88	Recognition Rate [%]	99	99	99	99	97	95	98	99	98
	Processing Time[Sec.]	98	90	115	90	95	90	95	105	110
128x128	Recognition Rate [%]	97	98	96	98	94	99	98	97	99
	Processing Time[Sec.]	228	220	245	220	225	220	225	235	240

Table 4 tabulates the results of the proposed framework which uses Adaboost to further classify the data that are generated by the adaptive binning method. It is because of the act of classifying the images for the second time, the data sets are reduced and hence the processing time is also reduced i.e. computational cost is reduced, and at the same time the recognition rate is also increased. For example, with the probe set frontal images (fa) the efficiency of recognition is nearly 99% and processing time equals to five seconds when considering the three different sized images.

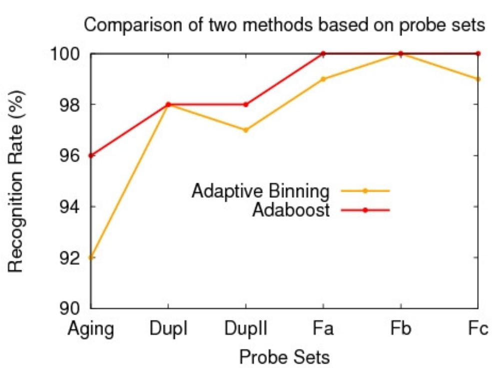
Table 4: Comparison chart for different image sizes and probe sets I for Adaboost Method

Size	Probe Set ->	Aging	Dup I	Dup II	Fa	Fb	Fc
64x64	Recognition Rate [%]	96	98	98	100	100	100
	Processing Time[Sec.]	5	3	2	3	3	2
88x88	Recognition Rate [%]	95	98	99	100	100	98
	Processing Time[Sec.]	8	5	4	5	5	4
128x128	Recognition Rate [%]	94	99	98	99	99	99
	Processing Time[Sec.]	9	9	8	9	9	8

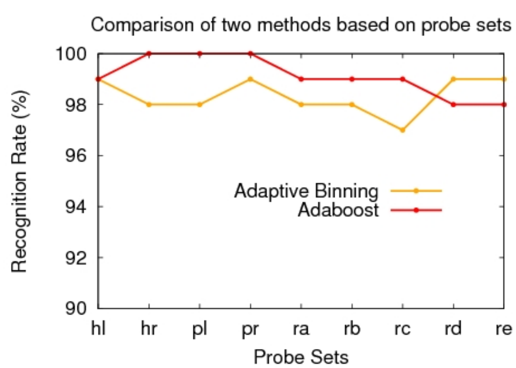
Table 4.1: Comparison chart for different image sizes and probe sets II for Adaboost Method

Size	Probe Set ->	hr	hl	pr	pl	ra	rb	rc	rd	re
64x64	Recognition Rate [%]	99	100	100	100	99	99	99	98	98
	Processing Time[Sec.]	3	2	2	3	2	3	3	3	2
88x88	Recognition Rate [%]	99	100	100	99	100	99	100	99	99
	Processing Time[Sec.]	5	4	4	5	4	5	5	5	4
128x128	Recognition Rate [%]	98	98	100	100	100	98	100	98	98
	Processing Time[Sec.]	9	8	8	9	8	9	9	9	8

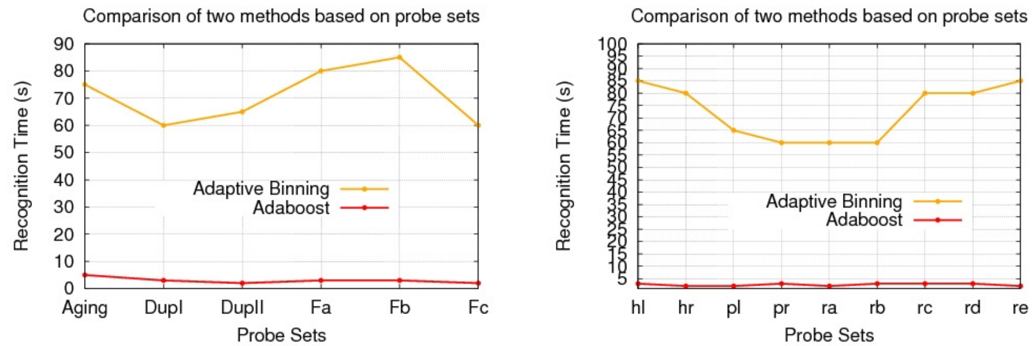
The Figure 2 shows the experimental results of adaptive binning and adaboost method for image size 64x64. Figure 2a compares the first 6 image probe sets vs. recognition rate, Figure 2b compares the rest of the image probe sets, whereas Figure 2c shows the results of processing time for first 6 image probe sets and the last Figure 2d shows the results of the rest of the probe sets. Similarly the Figure 3a, 3b, 3c and 3d shows the experimental results for 88x88 sized images and Figure 4a, 4b, 4c and 4d for 128x128 sized images.



a. Recognition rate vs. Probe Set I



b. Recognition rate vs. Probe Set I

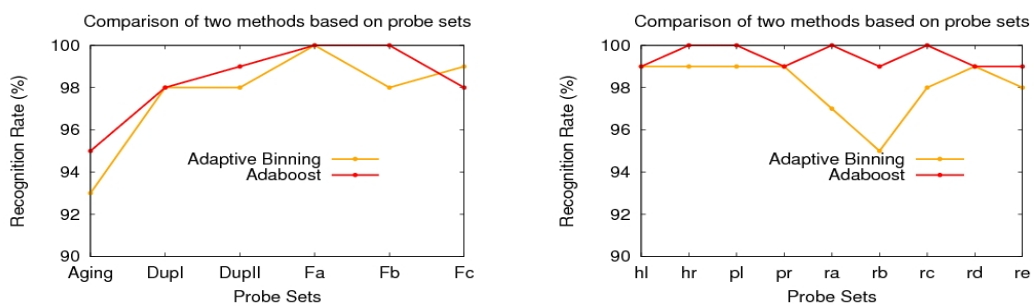


c. Recognition rate vs. Probe Set I   d. Recognition rate vs. Probe Set I

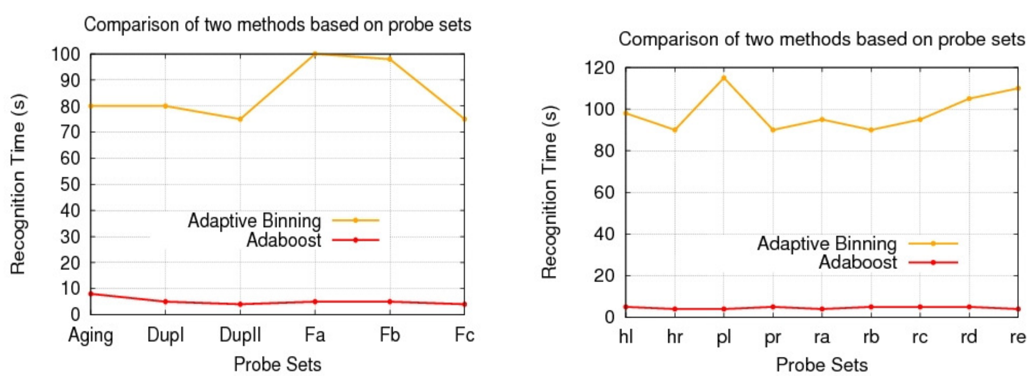
Fig: 2 Comparison chart for two methods for various 64x64 sized image probe sets.

The overall recognition rate increases linearly from aging, dup I, hl probe sets and reaches peak value for fa, fb, fc, hr, pl and pr. For dup II probe set, there is a slight decrease in recognition rate due to differences in dup II images. For probe sets ra, rb, rc, rd and re, the recognition rate drops because of face variations i.e. rotations of images. And, some features are dropped during pre processing stage, which result in decreased recognition rate.

Figure 2c and 2d show the processing time for two different techniques, because the data set are classified for the second time by using adaboost technique and the feature space is drastically reduced. This, results in reduced processing time, when compared with that in adaptive binning technique.

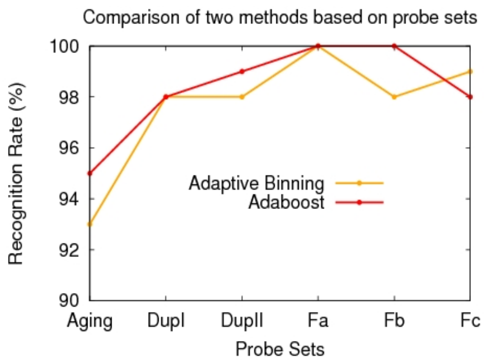


a. Recognition rate vs. Probe Set I   b. Recognition rate vs. Probe Set I

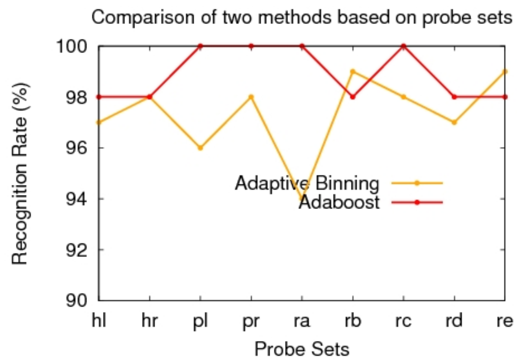


c. Recognition rate vs. Probe Set I   d. Recognition rate vs. Probe Set I

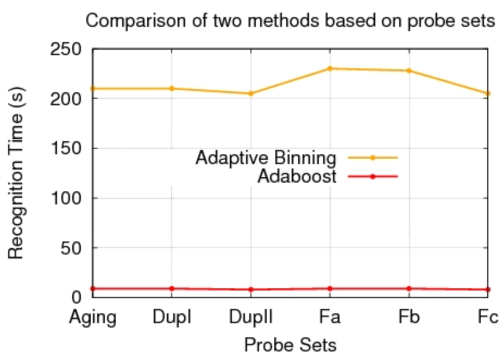
Fig: 3 Comparison chart for two methods for various 88x88 sized image probe sets.



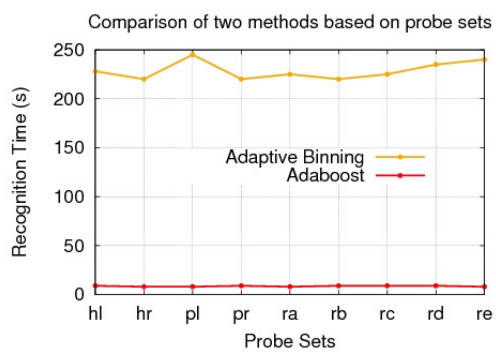
a. Recognition rate vs. Probe Set I



b. Recognition rate vs. Probe Set I



c. Recognition rate vs. Probe Set I



d. Recognition rate vs. Probe Set I

Fig: 4 Comparison chart for two methods for various 128x128 sized image probe sets.

A comparison of the Tables 3, 3.1, 4 and 4.1shows that the proposed new framework gives best efficiency and reduced computational cost for nearly all kinds of image probe sets. When the image size increases the efficiency increases. At the same time, processing time increases because larger data set are produced.

5. CONCLUSION

In this paper, a new framework for face recognition is developed by combining two classification algorithms: adaptive binning and adaboost. This system is tested with a larger database and the results show better identification of face with better efficiency. The feature space and execution time of this framework is reduced drastically compared with the other face recognition systems. The results stress the need for greater concentration on the probe sets fc, pr, ra, rb, rc, re and rd and also on the size of the probe set image, which plays an important role in obtaining good recognition rate. And, the system efficiency can be increased by adopting Edge Weighted Centroidal Voronoi Tessellation (EWCVT) technique, which can help in building more efficient recognition system with better representation of the images.

REFERENCES

[1] W. Zhao, R. Chellappa, A. Rosenfeld, P.J. Phillips (2003),"Face Recognition: A Literature Survey", ACM Computing Surveys, pp. 399-458.

[2] LinLin Shen and Li Bai (2005) "A review on Gabor wavelets for face recognition", Revision submitted, Pattern Analysis and Application, 2005.

[3] Zhimin Cao; Qi Yin; Xiaoou Tang; Jian Sun (2010), "Face recognition with learning-based descriptor", IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2010 doi: 10.1109/CVPR.2010.5539992,pp:2707 – 2714.

[4] Jie Chen, Ruiping Wang, Shengye Yan, Shiguang Shan, Xilin Chen, Wen Gao (2007) "Face Detection Based on Examples Resampling by Manifolds", IEEE Transactions on System, Man, and Cybernetics, Part A, 37(6), pp: 1017-1028.

[5] H. Moon, P.J. Phillips (2001), "Computational and Performance aspects of PCA-based Face Recognition Algorithms, Perception", Vol. 30, pp. 303-321.

[6] J. Lu, K.N. Plataniotis, A.N. Venetsanopoulos (2003), "Face Recognition Using LDA-Based Algorithms", IEEE Trans. on Neural Networks, Vol. 14, No. 1, January, pp. 195-200.

[7] L. Wiskott, J.-M. Fellous, N. Krueuger, C. von der Malsburg (1999), "Face Recognition by Elastic Bunch Graph Matching", Chapter 11 in Intelligent Biometric Techniques in Fingerprint and Face Recognition, eds. L.C. Jain et al., CRC Press, pp. 355-396.

[8] A.V. Nefian (2002), "Embedded Bayesian networks for face recognition", Proc. of the IEEE International Conference on Multimedia and Expo, Vol. 2, 26-29 August, Lusanne, Switzerland, pp. 133-136.

[9] Mian Zhou, Hong Wei and Stephen Maybank, (2006) "Gabor Wavelets and AdaBoost in Feature Selection for Face Recognition", Workshop in application of computer vision.

[10] Wenchao Zhang, Shiguang Shan, Xilin Chen, and Wen Gao (2007), "Local Gabor Binary Patterns Based on Mutual Information for Face Recognition", International Journal of Image and Graphics, 7(4) pp: 777-793.

[11] Yimo Guo, Zhengguang Xu, (2008) "Local Gabor Phase Difference Pattern for Face Recognition", International Conference on Pattern Recognition ICPR', pp 1-4.

[12] Hongming Zhang, Wen Gao, Xilin Chen, and Debin Zhao (2006) "Object Detection Using Spatial Histogram Features. Image and Vision Computing", 24(4) pp: 327-341..

[13] Dinu Coltuc,Philippe Bolon and Jean-Marc Chassery , (2006) "Exact Histogram Specification", IEEE Transaction on image processing, vol. 15,No.5,pp.1143-1152.

[14] Steven Diehl and Thomas S. Statlery, (2006) "Adaptive Binning of X-ray data with Weighted Voronoi Tessellations ",Monthly Notices of Royal Astronomical Society, vol. 368, No. 2, pp. 497-510(14).

[15] Baochang Zhang, Shiguang Shan, Xilin Chen & Wen Gao, (2007) "Histogram of Gabor Phase Patterns (HGPP): A Novel Object Representation Approach for Face Recognition", IEEE Transactions on Image Processing, vol. 16, No.1, pp 57-68.

[16] A.Srinivasan, R.S.Bhuvaneswaran, (2008) "Face Recognition System using HGPP and adaptive binning method", Int'l Conf Foundations of Computer Science FCS', pp 80-85.

[17] J.S.Sanders, and A.C.Fabian (2001), "Adaptive binning of X-Ray galaxy cluster images", Monthly Notices of the Royal Astronomical Society Journal, Volume 325, Issue 1, doi: 10.1046/j.1365-8711.2001.04410.x, pages 178–186, July 2001.

[18] Jianfu Chen, Xingming Zhang, Jinsheng Li, (2008) "Face verification based on Adaboost Learning for Histogram of Gabor Phase Patterns (HGPP) selection and samples synthesis with quotient image method" , Proceedings of the 4th international conference on Intelligent Computing: Advanced Intelligent Computing Theories and Applications - with Aspects of Theoretical and Methodological Issues; vol. 5226, pp 430 – 437.

[19] Mian Zhou, Hong Wei and Stephen Maybank, (2006) "Face Verification Using Gabor Wavelets and AdaBoost", International Conference on Pattern Recognition ICPR', pp 404-407.

Author

A.Srinivasan completed his ME, PhD in computer Science and Engineering at Madras Institute of Technology, Anna University, Chennai, India. He has finished his Post doctorate at Nan yang Technological University, Singapore. He has 18 years of Teaching and Research Experience in Computer Science and Engineering field and one year of Industrial Experience. At present, he has 6 PhD students working under him. He has published more than 40 Research publications in National and International journals and conferences. He is Editorial board member to Journal of Computer Science and Information Technology [JCSIT] and a Reviewer to Nine reputed International Journals in Computer Science and Engineering field. Currently he is working as Professor in Computer Science and Engineering Department, Misrimal Navajee Munoth Jain Engineering College, Anna University, Chennai, India. His fields of interests are Digital Image processing, Face Recognition and Distributed Systems.