Performance Comparison of Face Recognition Using DCT Against Face Recognition Using Vector Quantization Algorithms LBG, KPE, KMCG, KFCG

Dr. H. B. Kekre hbkekre@yahoo.com
Senior Professor, MPSTME, SVKM's
NMIMS University Mumbai,
400-056, India.

Dr. Tanuja K. Sarode tanuja_0123@yahoo.com
Assistant Professor, TSEC, Bandra (W),
Mumbai, 400-050, India.

Prachi J. Natu prachi.natu@yahoo.com
Assistant Professor, GVAIET, Shelu,
Karjat, 410201, India

Shachi J. Natu shachi_natu@yahoo.com
Lecturer, TSEC, Bandra (W),
Mumbai, 400-050, India.

Abstract

In this paper, a novel face recognition system using Vector quantization (VQ) technique is proposed. Four different VQ algorithms namely LBG, KPE, KMCG and KFCG are used to generate codebooks of desired size. Euclidean distance is used as similarity measure to compare the feature vector of test image with that of trainee images. Proposed algorithms are tested on two different databases. One is Georgia Tech Face Database which contains color JPEG images, all are of different size. Another database used for experimental purpose is Indian Face Database. It contains color bitmap images. Using above VQ techniques, codebooks of different size are generated and recognition rate is calculated for each codebook size. This recognition rate is compared with the one obtained by applying DCT on image and LBG-VQ algorithm which is used as benchmark in vector quantization. Results show that KFCG outperforms DCT and other three VQ techniques and gives better recognition rate up to 85.4% for Georgia Tech Face Database and 90.66% for Indian Face Database. As no Euclidean distance computations are involved in KMCG and KFCG, they require less time to generate the codebook as compared to LBG and KPE

Keywords: Face Recognition, DCT, VQ, KFCG, LBG

1. INTRODUCTION

Information is an important asset and can be in any form. Widespread use of an internet is making access and transmission of the information much easier. Particularly, the ability to access the information in an efficient manner has become a crucial issue. Hence some security measures must be provided so that only authorized users can access the intended information. Fingerprint recognition, speech recognition, iris recognition etc. are different recognition techniques that can be used for this purpose. But each of them has its own drawbacks. Special equipments to capture the input for recognition system and willingness of an individual to undergo testing procedure can hinder the implementation of these security measures. Face recognition is one of the important techniques and has become an interesting research topic due to its enormously commercial and law enforcement applications [1]. Face recognition is challenging because it is a real world problem. The human face is a complex, natural object that tends not to have easily (automatically) identified edges and features. Because of this, it is difficult to develop a mathematical model of the face that can be used as prior knowledge when analyzing a particular image. A human face is complex object with features that can vary over the time due to aging. Other factors that influence face recognition include shape, pose, poor lighting, occlusion, sunglasses, long hair, or other objects partially covering the subject's face and low resolution images [1].

Face recognition as a security measure has an advantage that it does not require any extensive equipment to capture the image of an individual and also explicit co-operation from an individual is not expected [10, 34]. Face recognition can also be used in human- computer interaction. A face recognition system is a process of automatically identifying or verifying a person from a digital image. In face recognition applications, original input image is of high dimension. Feature extraction is the most important step in face recognition, which reduces high dimensional image data to low dimensional feature vectors. A face recognition system can operate in following two modes:

Verification: A one to one comparison of a captured biometric with a stored template to verify that the individual is who he claims to be.

Identification: A one to many comparisons of the captured biometric against a biometric database in attempt to identify an unknown individual. The identification only succeeds in identifying the individual if the comparison of the biometric sample to a template in the database falls within a previously set threshold [15].

In this paper, the goal is to present effective face recognition system by using vector quantization technique. Vector Quantization (VQ) is simpler method for data compression and has been successfully used in various applications involving VQ based encoding and VQ based recognition. VQ consists of three steps Codebook design, Encoding and Decoding.

The rest of the paper is organized as follows: in section 2 we present related work carried out in the field of face recognition. In section 3 we discuss vector quantization. In section 4 our proposed approach is presented. Section 5 elaborates the experiment conducted. Results are tabulated in section 6. Conclusion has been outlined in section 7.

2. RELATED WORK

In face recognition system feature extraction plays an important role to create more robust systems [1].

Various techniques are proposed till now to extract feature vector from the image. PCA [15], Wavelet analysis, LDA [16], EBGM [15] are to name the few. Principle Component Analysis (PCA) approach is used to reduce the dimension of data by means of data compression basics and reveals the most effective low dimensional structure of facial patterns [17]. This reduction in dimension removes information that is not useful and precisely decomposes the face structure into orthogonal components known as eigenfaces. Each face image is represented as a feature vector which is stored in a 1D array. A distance between the respective feature vectors of images is compared to find the match [18]. In LDA samples of unknown classes are classified based on training samples of known classes. Whereas nonlinear characteristics of face are considered in EBGM. DCT [26] has been used as a feature extraction step in various studies on face recognition. Other transforms like Walsh-Hadamard transform [30, 31, 32], Wavelet transform also have been proposed [9, 24, 25, 28, 35].

For Recognizing objects from large image databases, histogram based methods were proposed in last decade. Initially, this idea was based on color histograms that were launched by Swain and Ballard [19]. Following this idea numerous developments were made by different people, exploiting this idea, such as texture histograms for 2D object recognition suggested by Gimelfarb and Jain [20], shape-index histograms for range image recognition proposed by Dorai and Jain [21] and relational histograms used by Huet and Hancock [22] for line-pattern recognition. Similarly, one dimensional (ID) and two dimensional (2D) histograms are also proposed with diverse variations like ID shape index histogram, 2D maximum and minimum curvature histogram, 2D mean and Gaussian curvature histogram and 2D shape index and curvedness histogram in [23]. Based on vector quantization algorithm, VQ histogram method is proposed by kotni [1].
Closest distance between histograms of different face images can be used for recognition purposes. Different distance measures may affect the recognition rate [22]. Euclidean distance can be used as it produces stable and satisfactory results [23].

2.1 Feature Extraction Using 2D-DCT on the Image

DCT technique that we have applied is as follows:

1) Read the color image from the database.
2) Convert the image into grayscale and resize it to size 128x128.
3) Apply 2D-DCT on this preprocessed image to obtain feature vector of an image.
4) Repeat steps 1 through 3 for each database image.
5) Read query image.
6) Repeat steps 2 and 3 for each query image to obtain its feature vector.
7) Calculate Euclidean distance of query image with each image in the database using equation (1).

$$ED = \sqrt{\sum_{i=1}^{n} (Vpi - Vqi)^2} \quad \text{...} \quad (1)$$

where V_{pi} and V_{qi} be the feature vectors of image P and Q respectively. Minimum Euclidean distance gives the closest matching image from the database. Accuracy of this algorithm is discussed in result section.

3. Vector Quantization [2, 7, 8]

VQ can be defined as a mapping function that maps k dimensional vector space to a finite set CB= $\{C_1, C_2, C_3, \ldots, C_N\}$. The set CB is called codebook consisting of N number of codevectors and each codevector $C_i = \{c_{i1}, c_{i2}, c_{i3}, \ldots\ldots, c_{ik}\}$ is of dimension k. The key to VQ is the good codebook. There are various codebook generation algorithms available in literature [7].

Vector quantization is composed of three operations: Codebook design, Encoding and Decoding. The input to the encoder is a vector and output is the index of the codevector that shows closest matching with that of input vector. In this case the closest match is found by evaluating the Euclidean distance between the input vector and each codevector in the codebook. Once the closest codevector is found, the index of that codevector is sent through a channel. When the decoder receives the index of the codevector, it replaces the index with the associated codevector [3]. Various techniques to generate the codebook are available. Codebook can be generated in spatial domain by clustering algorithms or using transform domain techniques [11].

The method most commonly used to generate codebook is the Linde-Buzo-Gray (LBG) algorithm [2, 7, 8].

3.1 Linde-Buzo-Gray (LBG) Algorithm [2,3,4]

Initially entire training vector of an image acts as one cluster. Centroid of this cluster is calculated. This is the first codevector (CV). Constant error is added to CV which gives two new vectors V1 and V2. Euclidian distance of all training vectors with V1 and V2 is calculated. Training vectors having minimum Euclidian distance with V1 than V2 are grouped into one cluster and remaining are grouped into another cluster. Now centroids of these two clusters are calculated which are nothing but the two new codevectors. These vectors overwrite the value CV in the codebook. Thus two new clusters are formed and codebook size has become two. In second iteration above procedure is repeated for two new clusters. It generates four new clusters. Then this procedure is repeated for each new cluster till codebook of desired size is generated [2].

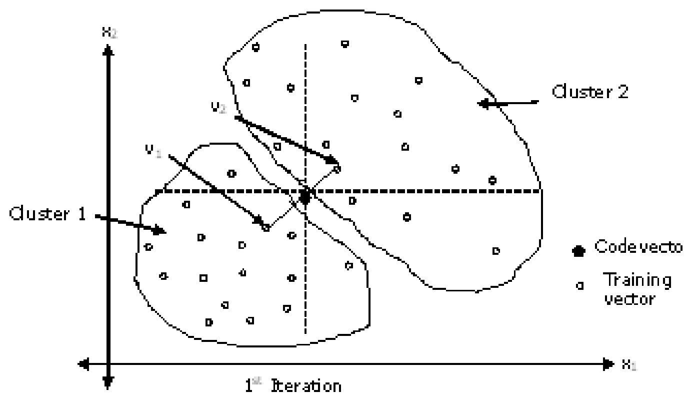

FIGURE 1: LBG for two dimensional case

3.2 Kekre's Proportionate Error Algorithm (KPE) [6, 33]

This algorithm is same as LBG, but the difference is, instead of adding constant error to codevector, we add proportionate error. Number of Euclidean distance computations required in LBG and KPE are 2N and numbers of comparisons are also 2N. Here N is the total number of training vectors in each iteration to generate clusters.

3.3 Kekre's Median Codebook Generation Algorithm (KMCG) [6, 29]

In this algorithm the training vector matrix T is sorted with respect to first column of the training vector. The median of training vector is selected and put into the codebook. Now codebook size is one. Matrix T is divided into two equal parts. Each part is sorted with respect to second column of the matrix. Median of each sorted part is chosen and written into the codebook. Codebook size is now increased to two. Each half obtained in previous step is again divided into two half parts. Thus four parts are obtained. Each of these parts is sorted with respect to next column number i.e. with respect to third column. Median of each cluster is chosen and written into the codebook. This gives codebook of size four. This process is repeated till the codebook of desired size is generated. As Euclidian distance computation is not required while generating the codebook, this algorithm generates codebook in a faster way.

3.4 Kekre's Fast Codebook Generation Algorithm (KFCG) [5, 27, 30]

KFCG does not use Euclidean distance to generate the codebook. Let T is the initial training vector. T contains m training vectors each of size k. T= {X1, X2,..., Xm} and X1= $\{x_{11}, x_{12}, \ldots\ldots x_{1k}\}$.

Initial codevector C1 is computed as centroid of training vector T. In first iteration, first element of each training vector of T is compared with the first element of codevector. i.e. if $x_{i1} < C1_1$ then put X_i in cluster 1 else in cluster 2.

In second iteration, split cluster1 into two by comparing x_{i2} with second element of the centroid of cluster 1 and cluster 2 into two by comparing x_{i2} with second element of the centroid of cluster 2. Repeat this procedure till codebook of desired size is generated.

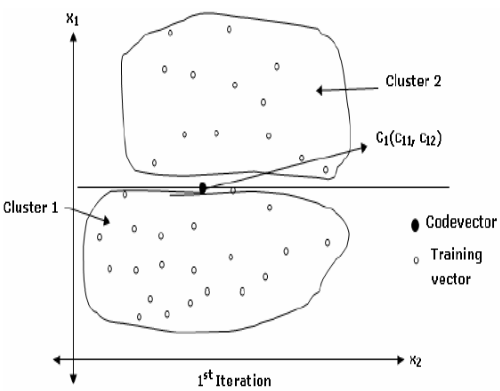

FIGURE 2: KFCG for two dimensional case with first iteration

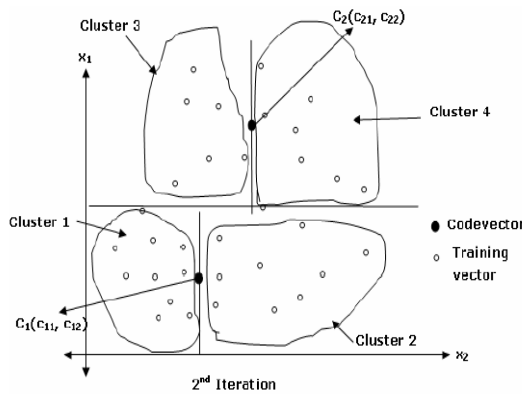

FIGURE 3: KFCG for two dimensional case with second iteration

3.5 Proposed Method

Figure 4 shows steps involved in proposed method.

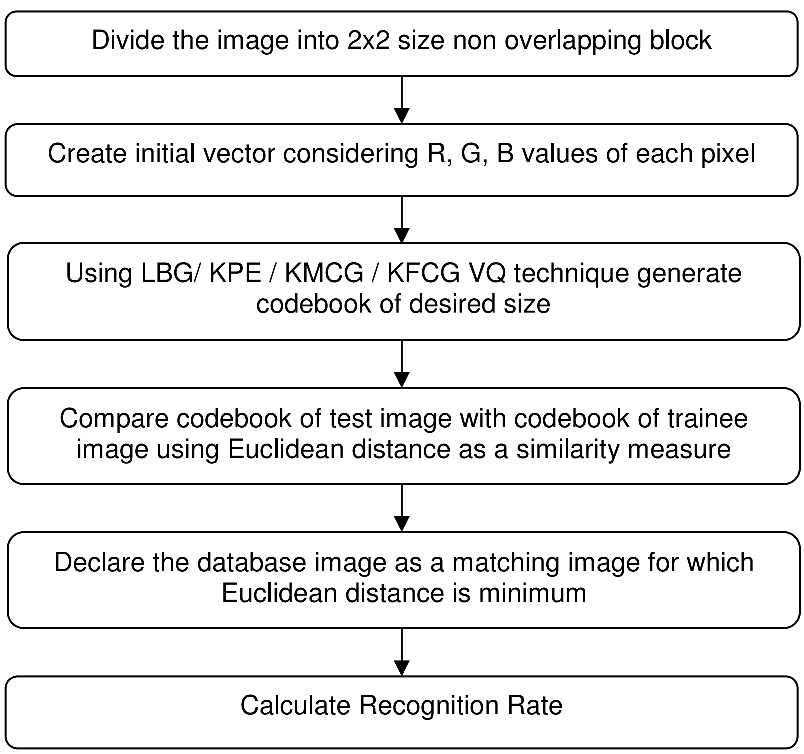

FIGURE 4: Flowchart for face recognition system using proposed VQ techniques

3.6 Query execution

Here the codebook of query image is generated by applying proposed VQ technique. This codebook is compared with other codebooks in codebook database using direct euclidean distance as similarity measure which is given in equation (1). Codebooks of size 4, 8, 16, 32, 64, 128, 256, 512 are generated using LGB, KPE, KMCG and KFCG algorithms. Recognition rate in each case is calculated and compared.

Recognition rate is calculated as:

$$(\%)\ \text{Recognition Rate} = \frac{\text{Number of images correctly recognized} \ \ X \ \ 100}{\text{Total number of images tested}} \quad \text{……….(2)}$$

4. Experiments and Results

In implementation of these algorithms, we have used two different databases, Georgia Tech Face Database and Indian Face Database. No preprocessing is done on the images in the database. These algorithms were implemented in MATLAB 7.0 on Intel Core 2 Duo Processor (2.0 GHz), 3GB RAM on Windows XP.

4.1 Georgia Tech Face Database

It contains total 750 images, 15 images each for 50 individuals. The images are in JPEG format and all are of different sizes. The images in the database show variations in illumination conditions, facial expression, and appearance. The images show frontal and/or tilted faces with different facial expressions, lighting conditions and scale. Out of 15 images per person, 8 images were taken as trainee images and remaining 7 images were taken as test images. Overall 350 images are test images and 400 images are trainee images. For second case, numbers of trainee images were increased to 10 images per person so that 250 images were used as test images and same methodology was applied. Sample images from Georgia Tech Face Database are shown below in figure 5.

FIGURE 5: Sample images from Georgia Tech Face Database

Steps mentioned in Section 2.1 are applied on the images in case 1 and case 2 to obtain the results of 2D-DCT. These results are mentioned in result section in 5.3. Steps mentioned in Section 3.5 are applied for each image in the database. For every image, codebooks of size 4x12, 8x12, 16x12, 32x12, 64x12, 128x12, 256x12, and 512x12 are generated using LBG, KPE, KMCG and KFCG algorithms. These codebooks are feature vectors of the corresponding images and are stored in the database. Feature vector of test image is compared with feature vectors of all trainee images using Euclidean distance as a measure of similarity to obtain the closest recognized image. After giving query image as input, the closest match is sent as output. Recognition rate is calculated for different codebook size using all four VQ algorithms.

4.2 Indian Face Database

Local database is generated to test efficiency of algorithms. This database has been prepared using Indian faces. The faces have been selected from long video clips where the object is asked to move the face with different angle and expressions. The shooting has been carried out at different locations with wide difference in illumination. From these video clips various frames have been selected for each object by taking care that variation of expression and angle are included. This database contains images of 50 persons and is stored in bitmap format. There are 10 color images of each person. Database contains images with rotation of face at different angles, different facial expressions and with considerable difference in illumination. Images are taken at different locations. Out of 10 images per person, 3 images were used in testing set and 7 in training set. In second case 2 test images were considered and 8 trainee images were taken. Sample faces from Indian Face Database are shown low. For this database also, codebooks of size 4x12, 8x12, …, 512x12 are generated using LBG, KPE, KMCG and KFCG algorithms and recognition rate obtained for these algorithms are compared. Sample images from Indian Face Database are shown below in figure 6. Steps mentioned in Section 2.1 and Section 3.5 are followed for Indian Face Database also.

FIGURE 6: Sample images from Indian Face Database

4.3 Results obtained by applying 2D-DCT on the image:

Technique Used	(%) Recognition Rate			
	Georgia Tech Face Database		Indian Face Database	
2D-DCT	400 Trainee Images	500 Trainee Images	350 Trainee Images	400 Trainee Images
	70%	70%	88.66%	90%

Table 1: Comparison of (%) Recognition Rate for different number of trainee images, by applying DCT on the image.

Table 1 shows percentage accuracy obtained by applying 2D- DCT on the gray scale resized image. Images from two different databases are used. It is observed that, for Georgia tech Face Database 70% accuracy is obtained when test sets of 350 and 250 images are used. For Indian Face Database, accuracy of 88.66% is obtained with 150 test images and 90% accuracy is obtained with 100 test images.

4.4 Results obtained by applying VQ Algorithms:

4.4.1 Results obtained for Georgia Tech Face Database:

VQ Technique	Codebook Size							
	4x12	8x12	16x12	32x12	64x12	128x12	256x12	512x12
LBG	73.42	79.14	**79.14**	77.14	76.85	72.28	70.57	64.57
KPE	73.42	**79.42**	78.57	77.14	76.85	74.85	72.85	64
KMCG	43.71	48.85	52.57	56.28	66.29	64.86	76.57	**78.57**
KFCG	78.28	**78.57**	74.57	75.71	76	76.86	76.86	76.28

Table 2: Comparison of (%) Recognition Rate for different codebook sizes by applying different VQ Techniques on Georgia Tech Face Database with 400 trainee images

Table 2 shows comparison of LBG, KPE, KMCG and KFCG algorithms. For each of these algorithms codebooks of size 4, 8,16,32,64,128,256 and 512 are generated. From Table 2 it is clear that, for different VQ techniques, codebook size that gives maximum recognition rate is different. Optimal codebook size for LBG is 16x12 whereas for KPE and KFCG it is 8x12. Due to overclustering, recognition rate decreases when codebook size is increased in LBG, KPE, and KFCG. For KMCG, recognition rate goes on increasing as codebook size is increased.

VQ Technique	Codebook Size							
	4x12	8x12	16x12	32x12	64x12	128x12	256x12	512x12
LBG	78.4	**84**	83.2	83.2	83.2	79.2	76.4	69.6
KPE	79.2	84	**84.6**	84	84	80.8	76.8	70
KMCG	51.6	48	66.8	69.2	75.6	80.4	84	**84.4**
KFCG	82.8	84	82.4	83.2	82.8	84	**85.4**	82.8

Table 3: Comparison of (%) Recognition Rate for different codebook sizes by applying different VQ Techniques on Georgia Tech Face Database with 500 trainee images.

By changing the number of trainee images for the same database, recognition rate is calculated. Here 5 test images and 10 trainee images per individual are used. Table 3 shows that if number of trainee images in the database is increased, the recognition rate also changes.

4.4.2 Results obtained for Indian Face Database:

VQ Technique	Codebook Size							
	4x12	8x12	16x12	32x12	64x12	128x12	256x12	512x12
LBG	66.66	78.66	84	86	86	**86.66**	82	58.66
KPE	66	76.66	77.33	80	**82**	76	67.33	53.33
KMCG	43.33	53.33	68.66	76	82.66	83.33	86	**87.33**
KFCG	79.33	80.66	84.66	86	86.66	**90.66**	84	77.33

Table 4: Comparison of (%) Recognition Rate for different codebook sizes by applying different VQ Techniques on Indian Face Database with 350 trainee images.

Table 4 shows the comparison of recognition rates for Indian Face Database. Here out of 10 images per person, 7 images are used as trainee images and 3 images are used for testing. It shows that KFCG outperforms all three VQ techniques in terms of recognition rate, giving 90.66% accuracy.

VQ Technique	Codebook Size							
	4x12	8x12	16x12	32x12	64x12	128x12	256x12	512x12
LBG	68	81	86	**88**	87	87	80	61
KPE	65	77	**78**	71	62	54	36	12
KMCG	45	55	64	77	80	82	**86**	**86**
KFCG	80	82	88	89	89	**90**	84	76

Table 5: Comparison of (%) Recognition Rate for different codebook sizes by applying different VQ Techniques on Indian Face Database with 400 trainee images

Table 5 shows comparison of recognition rate when 8 images per person in Indian Face Database are considered as trainee images. In this case also KFCG shows highest recognition rate than other VQ algorithms.

4.5 Complexity Analysis

Let M be the total number of training vectors,
k be the vector dimension,
N be the codebook size,
1 CPU unit is required for addition of 8 bit numbers
1 CPU unit for comparison
For multiplication/division of two 8 bit number 8 CPU units are required
To compute one squared Euclidean distance (ED) of k dimensional vector k multiplications and 2k-1 additions are required and hence 8k + 2k -1 CPU units are needed.
To compute Centroid for M training vectors of k dimension, M-1 additions and k divisions are required and therefore M-1 + 8k CPU units are needed.

Let $P = \log_2 N$

Complexity Parameters	LBG	KPE	KFCG	KMCG
Total Comparisons	2MP	2MP	MP	$M \sum_{i=0}^{P-1} \log_2 (M/2^i)$
Total No. of ED	2MP	2MP	0	0

Table 6: Comparison of LBG, KPE, KFCG and KMCG algorithm with respect to total number of Comparisons and total number of ED computations required.

Table 6 shows comparison of all four VQ algorithms in terms of two complexity parameters: total number of comparisons required and total number of Euclidean Distance computations required. It is observed that, numbers of comparisons required in KFCG are exactly half of the number of comparisons required in LBG and KPE and number of Euclidean Distance computations in KMCG and KFCG are zero. Hence generation of codebooks using KFCG and KMCG is faster.

Parameter	Algorithm applied on Georgia Tech Face Database with 400 Trainee Images				
	DCT	**LBG**	**KPE**	**KMCG**	**KFCG**
Codebook size for maximum recognition rate	-	16x12	8x12	512x12	8x12
Recognition Rate (%)	70	79.14	**79.42**	78.57	78.57

Table 7: Comparison of Recognition Rate by applying DCT and different VQ techniques on Georgia tech Face Database with 400 trainee images.

Table 7 shows the comparison of highest recognition rates obtained using VQ algorithms with the recognition rate obtained by applying 2D DCT technique on Georgia Tech Face Database with 400 trainee images. It is observed that, maximum rate of 79.42% is obtained for codebook size 8x12 using KPE algorithm. For same codebook size KFCG gives 78.57% accuracy. DCT gives 70% rate for same set of database.

Parameter	Algorithm applied on Georgia Tech Face Database with 500 Trainee Images				
	DCT	LBG	KPE	KMCG	KFCG
Codebook size for maximum recognition rate	-	8x12	16x12	512x12	256x12
Recognition Rate (%)	70	84	84.6	84.4	**85.4**

Table 8: Comparison of Recognition Rate by applying DCT and different VQ techniques on Georgia tech Face Database with 500 trainee images.

Similarly, from table 8 it is observed that, KFCG gives highest recognition rate of 85.4% for 256x12 size codebook with 500 trainee image set of Georgia Tech Face Database. DCT gives 70% for same case and peak value obtained by LBG, KPE and KMCG is 84%, 84.6% and 84.4% respectively.

Parameter	Algorithm applied on Indian Face Database with 350 Trainee Images				
	DCT	LBG	KPE	KMCG	KFCG
Codebook size for maximum recognition rate	-	128x12	64x12	512x12	128x12
Recognition Rate (%)	88.66	86.66	82	87.33	**90.66**

Table 9: Comparison of Recognition Rate by applying DCT and different VQ techniques on Indian Face Database with 350 trainee images.

Table 9 shows comparison of accuracy for Indian Face Database with 350 trainee images and 150 test images. DCT gives 88.66% accuracy. KFCG gives highest accuracy of 90.66% with 128x12 codebook size. KMCG gives maximum rate of 87.33%.

Parameter	Algorithm applied on Indian Face Database with 400 Trainee Images				
	DCT	LBG	KPE	KMCG	KFCG
Codebook size for maximum recognition rate	-	32x12	16x12	256x12	128x12
Recognition Rate (%)	90	88	78	86	**90**

Table 10: Comparison of Recognition Rate by applying DCT and different VQ techniques on Indian Face Database with 400 trainee images.

Result for second set of Indian Face Database is presented in table 10. Here KFCG gives maximum 90% recognition rate for 128x12 size codebook.

According to table 11, maximum accuracy obtained by KMCG is for codebook size 512x12. Using KFCG algorithm, maximum recognition rate is obtained at codebook size 256x12. Number of CPU units required by KMCG and KFCG are 368639 and 112723 respectively. Here also, KFCG requires least number of CPU units to perform all calculations.

Parameter	Algorithm applied on Georgia Tech Face Database with 500 Trainee Images with image size 128x128				
	DCT	LBG	KPE	KMCG	KFCG
Codebook size giving the maximum recognition rate	0	8x12	16x12	512x12	256x12
Number of multiplications required	4210688	29599	393420	6144	3084
Number of additions required	4194303	614579	803187	12287	55283
Number of comparisons required	0	24576	32768	307200	32768
Number of CPU units required	37879807	875947	3983315	368639	112723

Table 11: Comparison of number of calculations and CPU units required for applying DCT and different VQ techniques on Georgia Tech Face Database with 500 trainee images.

Though the size of codebook in case of LBG and KPE is smaller for maximum Recognition Rate than the codebook size in case of KFCG, the numbers of CPU units required by these two algorithms are higher than the CPU units required by KFCG.

5. CONCLUSION

In this paper, we have discussed face recognition using DCT and vector quantization. LBG, KPE, KMCG and KFCG-VQ algorithms are used to generate codebook of an image. No preprocessing is done on the images and their performance is compared with that of DCT technique in terms of recognition rate and number of CPU units required. DCT gives maximum accuracy of 70% for Georgia Tech face database and 90% for Indian face database. For Georgia Tech face database KFCG gives maximum accuracy of 85.4%. LBG gives 83.2%, KMCG gives 84.4% and KPE gives 84.6% accuracy. For Indian face database KFCG gives the best performance with the accuracy of 90.66%. LBG gives 88%, KMCG gives 87.33% and KPE gives 82% accuracy. Thus KFCG outperforms DCT algorithm and other VQ techniques namely LBG, KPE and KMCG. Numbers of CPU units required in KFCG are almost 336 times less than that of required in DCT and KFCG gives highest recognition rate. Also the CPU units required by KFCG are 7.77 times less than CPU units needed by LBG, 35 times less than that of needed by KPE and 3.27 times less than the CPU units required by KMCG. This is because the numbers of comparisons are reduced to half in KFCG as compared to LBG and KPE and no Euclidean distance computations are required in KMCG and KFCG which makes their execution much faster.

6. REFERENCES

[1] K. kotani, C. Qiu and T. Ohmi. “Face Recognition Using Vector Quantization Histogram Method”. International Conference on Image Processing, 2002

[2] H.B.Kekre, K. Shah, T. K. Sarode, S. Thepade. “Performance Comparison of Vector Quantization Technique-KFCG with LBG, Existing Orthogonal Transforms and PCA For Face Recognition”. International Journal of Information Retrieval, II(I):64-71, 2009

[3] Y. Linde, A. Buzo, R. M. Gray. “An algorithm for vector quantizer design”. IEEE Transaction on Communication, COM-28(1):84-95, 1980

[4] A. Gersho, R.M. Gray. “Vector Quantization and Signal Compression”, Kluwer Academic Publishers, Boston, 1991.

[5] H.B. Kekre, T. Sarode.” An Efficient Fast Algorithm to Generate Codebook for Vector Quantization”. First International conference on Emerging Trends In Engineering and Technology (ICETET), 2008

[6] H.B. Kekre, T. Sarode. “Fast Codebook Search Algorithm For Vector quantization Using Sorting technique”. ACM International Conference on Advances in Computing, Communication And Control (ICAC3) 2009

[7] Y. Linde, A. Buzo, R. M. Gray. “An algorithm for vector quantizer design”. *IEEE* Trans. Commun., COM-28(1):84-95, 1980

[8] C. chan, L. Po. “A Complexity reduction Technique For Image Vector Quantization”. IEEE Transactions on Image Processing, 1(3):312-321, 1992

[9] H. K. Ekenel, R. Stiefelhagen. “Analysis of Local Appearance Based Face recognition: Effects of Feature Selection and Feature Normalization”. International Conference on Computer vision and Pattern Recognition Workshop, 2006

[10] S. Lin. “An Introduction to Face Recognition Technology”. Informing Science Special Issue on Multimedia Informing Technologies- Part 2, 3(1): 2000

[11] J. Choi, Y. Chung, K. Kim, J. Yoo. "Face Recognition using Energy Probability in DCT Domain".IEEE International Conference on Multimedia and Expo, 2006

[12] H. B. Kekre, Ms. Tanuja K. Sarode, S. D. Thepade. “Image Retrieval using Color-Texture Features from DCT on VQ Codevectors obtained by Kekre’s Fast Codebook Generation”. ICGST-International Journal on Graphics, Vision and Image Processing (GVIP), 9(5):1-8, 2009. Available at http:// www.icgst.com/gvip/Volume9/Issue5/ P1150921752.html.

[13] “Georgia Tech Face Database”. Available at: http://www.face-rec.org/databases.

[14]”Face Recognition”. Available at: http://www.biometrics.gov/Documents/FaceRec.pdf

[15] P. N. Belhumeur, J. P. Hespanha, D. J. Kriegman. “Eigenfaces vs. fisherfaces: Recognition using class specific linear projection”. IEEE Transactions on Pattern Analysis and Machine Intelligence, 19(7):711–720, 1997

[16] M. Begum, N. Nahar, K. Fatimah, M. K.Hasan and M. A. Rahaman. “An Efficient Algorithm for Codebook Design in Transform Vector Quantization”. WSCG, 2003

[17] F. Basit, M. Y. Javed and U. Qayyum. “Face Recognition Using Processed Histogram and Phase-only Correlation (POC)”. International conference on Emerging Technologies (ICET), 2007

[18] M. J. Swain and D. H. Ballard, "Indexing via color histogram", In Proceedings of third international conference on Computer Vision (ICCV), Osaka, Japan, 1990.

[19] G. L. Gimel'farb, A. K. Jain. "On retrieving textured images from an image database". Pattern Recognition, 29(9):1461-1483, 1996

[20] C. Dorai, A. K. Jain, Cosmos. "A representation scheme for free form surfaces". IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), 19(10):1115-1130, 1997

[21] B. Huet, E. R. Hancock. "Line pattern retrieval using relational histograms". IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), 21(12):1363-1370, 1999

[22] Y. Li, E. R. Hancock. "Face Recognition using Shading-based Curvature Attributes". IEEE Proceedings ofthe 17th International Conference on Pattern Recognition (ICPR'04) 1051-4651/04

[23] H.B.Kekre, S. D. Thepade. "Improving the Performance of Image Retrieval using Partial Coefficients of Transformed Image". International Journal of Information Retrieval (IJIR), Serials Publications, 2(1):72-79, 2009

[24] H.B.Kekre, T. Sarode, S. D. Thepade. "DCT Applied to Row Mean and Column Vectors in Fingerprint Identification". In Proceedings of International Conference on Computer Networks and Security (ICCNS), VIT, Pune, 2008

[25] H.B.Kekre, S. D. Thepade, A. Athawale, A. Shah, P. Verlekar, S. Shirke. "Energy Compaction and Image Splitting for Image Retrieval using Kekre Transform over Row and Column Feature Vectors". International Journal of Computer Science and Network Security (IJCSNS), 10(1): 2010, (ISSN: 1738-7906) Available at www.IJCSNS.org.

[26] H.B.Kekre, S. D. Thepade, A. Athawale, A. Shah, P. Verlekar, S. Shirke. "Performance Evaluation of Image Retrieval using Energy Compaction and Image Tiling over DCT Row Mean and DCT Column Mean". Springer-International Conference on Contours of Computing Technology (Thinkquest-2010), Babasaheb Gawde Institute of Technology, Mumbai, 2010

[27] H.B.Kekre, T. K. Sarode, S. D. Thepade, V. Suryavanshi. "Improved Texture Feature Based Image Retrieval using Kekre's Fast Codebook Generation Algorithm". Springer-International Conference on Contours of Computing Technology (Thinkquest-2010), Babasaheb Gawde Institute of Technology, Mumbai, 2010

[28] H. B. Kekre, T. K. Sarode, S. D. Thepade. "Image Retrieval by Kekre's Transform Applied on Each Row of Walsh Transformed VQ Codebook". (Invited), ACM-International Conference and Workshop on Emerging Trends in Technology (ICWET 2010),Thakur College of Engg. And Tech., Mumbai, 2010

[29] H. B. Kekre, T. Sarode, S. D. Thepade. "Color-Texture Feature based Image Retrieval using DCT applied on Kekre's Median Codebook". International Journal on Imaging (IJI), 2(A09):55-65, 2009.Available at www.ceser.res.in/iji.html (ISSN: 0974-0627).

[30] H. B. Kekre, Ms. T. K. Sarode, S. D. Thepade. "Image Retrieval using Color-Texture Features from DCT on VQ Codevectors obtained by Kekre's Fast Codebook Generation". ICGST-International Journal on Graphics, Vision and Image Processing (GVIP), 9(5):1-8, 2009. Available at: http://www.icgst.com/gvip/ Volume9/Issue5/P1150921752.html.

[31] H. B. Kekre, S. Thepade, A. Maloo. "Image Retrieval using Fractional Coefficients of Transformed Image using DCT and Walsh Transform". International Journal of Engineering Science and Technology, 2(4): 362-371, 2010

[32] H. B. Kekre, S. Thepade, A. Maloo. "Performance Comparison of Image Retrieval Using Fractional Coefficients of Transformed Image Using DCT, Walsh, Haar and Kekre's Transform". CSC-International Journal of Image processing (IJIP), 4(2)142-155, 2010

[33] H. B. Kekre, T. Sarode. "Two Level Vector Quantization Method for Codebook Generation using Kekre's Proportionate Error Algorithm". CSC-International Journal of Image Processing, 4(1):1-10, 2010

[34] H. B. Kekre, S. Thepade, A. Maloo. "Eigenvectors of Covariance Matrix using Row Mean and Column Mean Sequences for Face Recognition". CSC-International Journal of Biometrics and Bioinformatics (IJBB), 4(2):42-50, 2010

[35] H. B. Kekre, T. Sarode, S.i Natu, P. Natu. "Performance Comparison Of 2-D DCT On Full/Block Spectrogram And 1-D DCT On Row Mean Of Spectrogram For Speaker Identification". CSC International Journal of Biometrics and Bioinformatics (IJBB), 4(3).