Augmentation of Face Recognition and Retrieval (AFRR) Under Various Illuminations Using Block Truncation Coding

Dr.S.Prasath
Assistant Professor, Department of Computer Science,
Erode Arts & Science College (Autonomous), Erode, Tamil Nadu, India.
Email: softprasaths@gmail.com

Abstract— **A simple and efficient Block Truncation Coding (BTC) method to compress two dimensional color shape without loss of data as well as quality of the picture. BTC is one of the lossy image compression algorithms but in this research work has proposed on extended version of image compression using BTC. The given color RGB value is converted into gray scale image. The features are extracted and constructed in matrix form. Then the matrix form is divided into block. Finally compressed block table is generated. The proposed algorithm is tested and implemented on various parameters. The experiments and shows that proposed method provides better recognition rate when compared with the existing methods then experiments are carried out using MATLAB.**

Keywords- BTC, RGB, SV, MV, GSV, CB, CR, MSE, PSNR.

1. INTRODUCTION

Today Information Technology plays a vital role in every field of human life. Digital image Processing is rapidly growing area for last few decades and involved in various fields of research as well in human life also. Present, digital imaging is popularly technology and used in many industrial solutions concerning. In this process industry several on-line measurement systems are based on the digital imaging. As a result of this rapid development, the amount of image data has increased rapidly and leads to a complexity in data storage and management. Consequently the size of different kinds of image databases in the industry has increased significantly. Therefore managing of these databases for further processing becomes a tedious one. Hence, lot of techniques and methods has evolved to overcome those issues.

2. RELATED WORKS

Aditya Kumar et.al [1] presented the improved BTC algorithm, namely Enhanced Block Truncation Coding was presented in this research study. Several gray scale images are used to appraise the coding efficiency and performance of this proposed algorithm with existing image compression algorithm. In the EBTC algorithm, here bit plane is encoded to improve the compression ratio and also improve the image quality. Simulation results show that bit rate and PSNR of this proposed EBTC are better than BTC as well as the ABTC in terms of subject visual quality and significantly improves the compression ratio of BTC.

Gaganpreet Kaur et al. [2] analyzed lossy and lossless image compression techniques. They studied and discussed all the techniques. Finally, concluded lossy compression techniques provided high compression ratio than lossless compression scheme. Lossy compression is used for more compression ratio and Lossless compression is used when the original image and reconstructed image are to be identical.

Prabhakar et al. [3] proposed the efficient medical image compression as a combined approach of SPIHT and Lempel-Ziv-Welch coding. This combined approach was applied for both color and gray scale images. They concluded that compressed image with good compression ratio and good PSNR value. The proposed method was comparatively better than the existing algorithms regarding PSNR and Compression ratio.

Jun Wang et. al [4] presented an improved motion adaptive codec. The IMAC contains two approaches to improve the motion adaptive codec overdrive. The first was an advanced hybrid image codec for the efficient reduction of the image data stored in frame memory. The second was an advanced motion adaptive selector to reduce the overdrive error.

Lucas Hui [5] proposed a method of designing a minimum mean square error quantizer for the block truncation coding algorithm for image compression is presented. Based on this MMSE quantizer, an adaptive block truncation coding algorithm is designed. This method tries to optimize the quantization output levels based on the locality of the image pixel blocks and significantly improved reconstructed image quality was achieved at 2 b/pixel. The ABTC method is very efficient as compared to the standard BTC and the AMBTC methods.

Lema et. al [6] presented the Color image data compression using absolute moment block truncation coding scheme was implemented. This compression technique reduces the computational complexity and achieves the optimum minimum mean square error and PSNR. It was an improvised version of BTC, obtained by preserving absolute moments. AMBTC is an encoding technique that preserves the spatial details of digital images while achieving a reasonable compression ratio.

Meftah et. al [7] proposed a method called the Improved Adaptive Block Truncation Coding based on Adaptive Block Truncation Coding. The feature of inter-pixel redundancy is exploited to reduce the bit-rate further by retaining the quality of the reconstructed images.

Pravin et al. [8] presented comparative of various image compression techniques to assess the progress made in the field of image compression effects on different images for different applications. They reviewed about the image compression, need of compression, its principles, and types of compression and different algorithms used for image compression.

Delphi et. al [9] proposed to improve the transmission rate and storage space the compression of image is very necessary. For compression of image many tools and methods are applied. They are developed block truncation coding (BTC) was easy to implement. The bit rate obtained by BTC is 2 bpp but the quality of reconstructed image was very low.

Kaspar Raj et. al [10] proposed three and two stage still gray scale image compressor based on BTC. They have employed a combination of four techniques to reduce the bit rate. They are quad tree segmentation, bit plane omission, bit plane coding using 32 visual patterns and interpolative bit plane coding. The experimental results show that the proposed schemes achieve an average bit rate of 0.46bits per pixel for standard gray scale images with an average PSNR value of 30.25, which is better than the results from the exiting similar methods based on BTC.

Somasundaram et. al [11] proposed an extended version of Absolute Moment Block Truncation Coding to compress images. Generally the elements of a bitplane used in the variants of Block Truncation Coding are of size 1 bit. But it has been extended to two bits in the proposed method.

Somasundram et. al [12] proposed method the feature of inter-pixel correlation was exploited to further reduce the requirement of bits to store a block. The proposed method gives very good performance in terms of bit-rate and PSNR values when compared to the conventional BTC. The inter-pixel redundancy feature is incorporated in the conventional BTC method and this has lead to further reduction in the bit rate and improvement in the PSNR of the reconstructed images. The EBTC method was tried with different threshold values in categorizing the low and high detail blocks. As the threshold value increases, the PSNR was improved and the bit-rate was reduced. The technique was tried with different images and yielded better results with less computational effort.

Vimala et. al [13] proposed a method called the Improved Adaptive Block Truncation Coding based on Adaptive Block Truncation Coding. The feature of inter-pixel redundancy is exploited to reduce the bit-rate further by retaining the quality of the reconstructed images. The proposed method outperforms the existing BTC based methods both in terms of bit-rate and PSNR values.

3. METHODOLOGY

In the existing system an image compression and decompression using Block truncation code algorithm has been developed for a grayscale image. The below sample example clearly shows the importance of compression. An image has 1024×1024 pixel/24 bit, without compression, would require 3 MB of storage and 7 minutes for transmission, utilizing a high speed, 64 Kbits/sec. If the image is compressed at a 10:1 compression ratio, the storage requirement is reduced to 300 KB and the transmission time drop to less than 6 sec.

3.1.1 Block Truncation Coding (BTC)

Block Truncation Coding (BTC) is a well-known compression scheme proposed in 1979 for the grayscale images. It was also called the moment-preserving block truncation because it preserves the first and second moments of each image block. The amount of image data grows day by day. Large storage and bandwidth are needed to store and transmit the images, which is quite costly. Hence methods to compress the image data are essentially now-a-days. Block Truncation Coding is a simple technique which involves less computational complexity. BTC is a recent technique used for compression of monochrome image data. It is one-bit adaptive moment-preserving quantizer that preserves certain statistical moments of small blocks of the input image in the quantized output. The original algorithm of BTC preserves the standard mean and the standard deviation. The statistical overheads mean and the Standard deviation are to be coded as part of the block. Various methods have been proposed during last twenty years for image compression such BTC and Absolute Moment Block Truncation Coding (AMBTC). AMBTC preserves the higher mean and lower mean of the blocks and use this quantity to quantize output. AMBTC provides enhanced image quality than image compression using BTC.

3.1.2 Standard BTC

Block Truncation Coding (BTC) works by dividing the image into as well as small sub blocks of size 4 x 4 pixels and then reducing the number of gray levels within each block. This reduction is performed by a quantizer that adapts to the local image statistics. The basic form of BTC divides the whole image into N blocks and codes each block using a two-level quantizer. The two levels are selected using the mean and standard deviation of the gray levels within the block and are preserved. Each pixel value within the block is then compared with the mean and then is assigned to one of the two levels.

3.1.3 Absolute Moment Block Truncation Coding (AMBTC)

In this technique image compression is done using absolute moment block truncation coding. It is an improved version of BTC, preserves absolute moments rather than standard moments, here also a digitized image is divided into blocks of n x n pixels. Each block is quantized in such a way that each resulting block has the same sample mean and the same sample first absolute central moment of each original block. A. Image Compression Using AMBTC.AMBTC has been extensively used in signal compression because of its simple computation and better mean squared error (MSE) performance. It has the advantages of preserving single pixel and edges having low computational complexity. The original algorithm preserves the block mean and the block standard deviation. Other choices of the moments result either in less MSE or less computational complexity. In AMBTC algorithm similar to BTC there are four separate steps while coding a single block of size n x n. They are quad tree segmentation, bit plane omission, bit plane coding using 32 visual patterns and interpolative bit plane coding. In this work, used compressed the image using AMBTC algorithm.

- Original image is segmented into blocks of size 4x4 or 8x8 for processing.
- Find the mean of each block and also find levels from the mean which has higher range and lower range.
- Based on these calculated mean value for each block, bit plane is calculated.
- Image block is reconstructed by replacing the '1's with higher mean and '0's with lower mean. At the encoder side an image is divided into non-overlapping blocks.

The size of each non overlapping block may be (4 x 4) or (8 x 8), etc. and then it is calculate the average gray level of the block (4x4) as (for easy understanding we made the compression for 4 x 4 block).

3.2 Multilevel Block Truncation Coding (MLBTC)

Block Truncation Coding has been used. It has been implemented on four levels are,

3.2.1 BTC Level 1

A face image is taken from the database and the average intensity value of each color plane of the image is calculated. The color space considered in this algorithm is the RGB color space. So the average intensity value of each of the RGB plane of a face image is calculated. The further discussion is done using the Red plane of an image. The same has to be carried out for the Blue and Green color space. After obtaining the average value intensity value of the Red color plane of the face image, each pixel is compared with the mean value and the image is divided into two regions: Upper Red and Lower Red. The average intensity values of these regions is calculated and stored in the feature vector as UR and LR. Thus, after repeating this procedure for the Blue and the Green color space our feature vector has six elements: Upper Red, Lower Red, Upper Green, Lower Green, Upper Blue and Lower Blue.

3.2.2 BTC Level 2

At level two the values Upper Red and Lower Red are extracted from the feature vector of BTC level 1 and using these values, the Red plane of face image is now divided into four regions. These are Upper-Upper Red, Upper-Lower Red, Lower-Upper Red and Lower-Lower Red. The average intensity values in these four regions is calculated and stored in the feature vectors. The above process is reiterated for the Blue and Green color spaces of the face image. Thus the feature vector at this level has 12 elements, 4 elements for each plane.

3.2.3 BTC Level 3 and BTC Level 4

Using the procedures described in the Levels 1 & 2, the face images are further divided into more regions in each of the color space. These regions are depicted in figure1. The average intensity value at these regions are calculated and stored in the feature vector. The feature vector has 24 elements at BTC-level 3 and 48 elements at BTC-level 4. The feature vectors obtained in BTC-levels 1,2,3,4 are used for comparison with the database images set.

3.3 BTC-INTERMEDIATE-4 & BTC-INTERMEDIATE-9

To calculate the feature vector in this algorithm, Block Truncation Coding has been used. It has been implemented on four levels which are explained below.

3.3.1 BTC-Intermediate-4 Level 1

In BTC-Intermediate-4, the given image is partitioned into four non overlapping equal parts. Each part of the image is then considered individually. After this , BTC is applied on each part of the given image .For each part the average intensity value of each of the RGB plane is calculated independently and which helps in dividing the RGB plane into upper and lower clusters respectively. The 6 elements (UR, LR, UG, LG, UB and LB) for an image. Further we calculate the mean value of all the four feature vectors of the four parts of the given image to obtain the feature vector of the entire image. This is the basic implementation of BTC-Intermediate-4 method applied on BTC based face recognition level 1 and hence we call it as BTC-Intermediate-4 Level 1.

3.3.2 BTC-Intermediate-4 Level 2

In level 2 implementation of BTC based face recognition of BTC-Intermediate-4 get 12 elements for each feature vector of the four parts of the given image. Thus to obtain the feature vector of the image, the mean value of all the four feature vectors is calculated.

3.3.3 BTC-Intermediate-4 Level 3 and BTC-Intermediate-4 Level 4

Similarly in level 3 and level 4 implementation of BTC based face recognition of BTC-Intermediate-4 get 24 elements and 48 elements respectively. Thus the mean value is calculated at both the levels in order to obtain the feature vector of the entire image.

3.3.4 BTC-Intermediate-9

The implementation of BTC-Intermediate-9 is same as that of BTC-Intermediate-9, but in this method the image is divided into nine non overlapping equal parts. Similarly to obtain the feature vector of the given image the mean value of all the feature vectors of the nine parts is calculated. The implementation of BTC-Intermediate-9 is also done up to 4 levels.

3.4 PROPOSED AUGMENTATION OF FACE RECOGNITION AND RETERIVAL (AFRR)

The main objective of this proposed system is to improve the quality of the image and reduce the size. This research proposes an image compression technique using a Improved Block Truncation Coding (IBTC) to compress the image. The following figure shows that block diagram of proposed system.

Initially, all the images are preprocessed. In color based approaches certain region of interest is cropped from the images in order to avoid processing unnecessary detail present in the face. During training stage features are extracted from every image and are stored separately in the database. While testing a search image features are extracted for that image and are matched against all the images in the database. The dissimilarity measure used to match feature is chi-square. The performance of the proposed approach is studied using color features separately and combined.

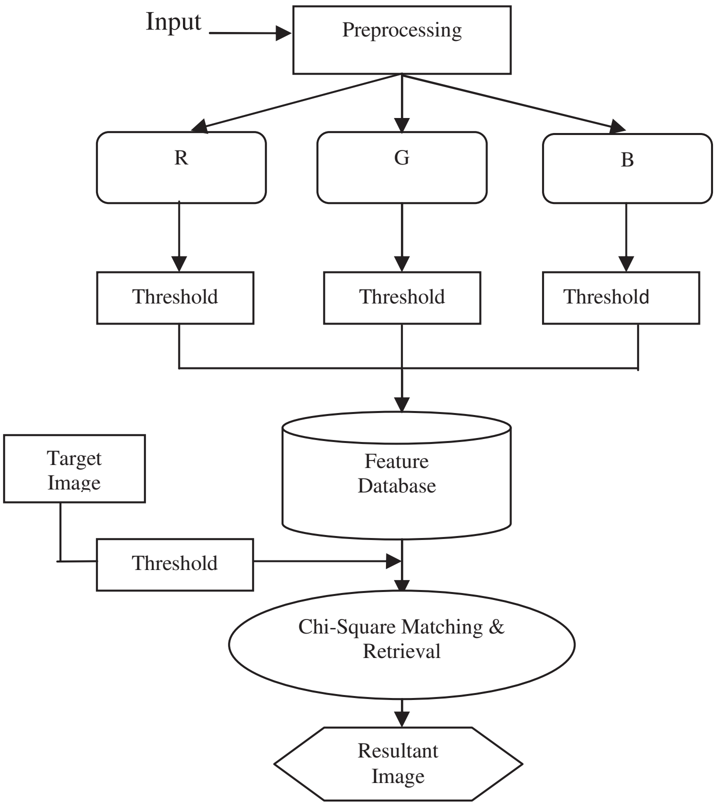

Fig.3 Proposed Face recognition process

3.4.1 Feature Extraction

The feature vector at each BTC level for the query image and database set is extracted. Here three binary bitmaps values will be computed. If a pixel in each component (R, G and B) for threshold.

The feature vectors obtained in each level of BTC are used to compare with the database images. The comparison of similarity measures is done by Chi-square statistic defined as equation (2).

$$x^2\,(I,T) = \sum \frac{(I_i - T_i)^2}{(I_i - T_i)} \qquad \text{--->}\ (2)$$

Where, I_i is the ith feature value of the gallery set image and T_i is the ith feature value of the search one. The dissimilarity among the images for color features using weighted Chi-square defined in equation (3).

$$x_w^2(I,T) = w_i \sum_{i,j} \frac{(I_{i,j} - T_{i,j})^2}{(I_{i,j} + T_{i,j})} \qquad \text{--->}\ (3)$$

In the above formula $I_{i,j}$ and $T_{i,j}$ are the jth feature of ith region in the input database image and search image respectively. wi is the weight of ith region. Then normalize dissimilarity measures for color features individually for the images in the training set .

Where, NF is the normalized dissimilarity measure for every image in the input database image $x^2_w(I, T)$ is the dissimilarity measure, max x^2_w (I, T) and min x^2_w (I, T) are the maximum and minimum distance measures among the input database images respectively.

4. SIMILARITY AND PERFORMANCE MEASURES

4.1 False Acceptance Rate (FAR)

False Acceptance Rate (FAR) and Genuine Acceptance Rate (GAR) are standard performance evaluation parameters of face recognition system. The False acceptance rate (FAR) is the measure of the likelihood that the biometric security system will incorrectly accept an access attempt by an unauthorized user. A system's FAR typically is stated as the ratio of the number of false acceptances divided by the number of identification attempts.

FAR = (False Claims Accepted/Total Claims) X 100 ---> (4)

4.2 Genuine Acceptance Rate (GAR)

The Genuine Acceptance Rate (GAR) is evaluated by subtracting the FAR values from 100.

GAR=100-FAR (in percentage) ---> (5)

5. ALGORITHM

The process of the face images is take place in two phases and defined as algorithm I and algorithm II. In algorithm I results in the feature extraction and in the retrieval of face. In algorithm II the face matching is take place and the GAR and FAR has estimated and are as follows.

ALGORITHM I

// To Calculate threshold value //

Step 1 : *Select an input image I of size M x N (Even row and column)*

Step 2 : *Convert gray scale values into matrix format.*

Step 3 : A*pply sorting method for an array by using step 2.*

Step 4 : *Find out the middle gray scale values of lower range and upper range.*

Step 5 : *Find out the average value of middle gray scale values and take whole number in sorted array and also known as threshold value.*

Step6 : *Convert binary matrix by using threshold value.*

Step 7 : *Apply the weighted Chi-square to target image and input image by using equation (3).*

Step 8: *Repeat step 2 to step 6 for all images in the database.*

Step9: *Stop*

ALGORITHM II

Input : Face Image
Output : Feature set

//Generation of Face Image Feature set //

Step 1: Select an input image I of size M x N
Step 2: Divide the input image I into non-overlapping blocks with size w×w .
Step 3: Color Features in the form of vector in the form of histogram are computed for input images and are stored in a database.
Step 4: For every search image determine Color features vector for the image.
Step 5: The certain region in the face have more importance, compute the dissimilarity among the images for color features using weighted Chi-square by using equation (3).
Step 6: Compute normalize dissimilarity measures for color features individually for the images in the training set by applying equation (4).
Step 7: Then calculate normalized measures for every image in the input database images.
Step 8: The input database image which give up least dissimilarity measure with the target image is considered as the recognized one.
Step 9: Repeat step 5 to step 6 for all images in the database.
Step 10: Stop

ALGORITHM III

Input : Target Image
Output : Face Image Matching

// Face Image Matching//

Step 1: Select the target image L of size M x N and divide into m x n block.
Step 2: Repeat step 5 to step 6 as in the algorithm II.
Step 3: Compute the Chi-square distance between the target image and the image set for matching using the equation 3 and 4.
Step 4: Compute the GAR and FAR using the equation 4 and equation 5.
Step 5: Stop

6. EXPERIMENTATION AND RESULTS

The proposed AFRR feature extraction is experimented with the images collected from the standard database. Therefore all the images are rotated in such a way that a line connecting iris centers lies in a horizontal line and the following experimental setup, parameter settings are used. This database is created by Dr Libor consisting of 1000 images (each with 180 pixels by 200 pixels), corresponding to 100 persons in 10 poses each, including both males and females. All the images are captured against a dark background, little variation in illumination. The subjects are next to fixed distance from the camera are taken with sequence of images. The ten poses of Face database are shown in figure 4.

False Acceptance Rate (FAR) and Genuine Acceptance Rate (GAR) are standard performance evaluation parameters of face recognition system. The false acceptance rate is the measure of the likelihood that the biometric security system will incorrectly accept an access attempt by an unauthorized user. A system FAR typically is stated as the ratio of the number of false acceptances divided by the number of identification attempts. The average FAR and GAR of all queries in respective face databases are considered for performance ranking of BTC levels. For optimal performance the FAR values must be less and accordingly the GAR values must be high for each successive levels of BTC.

Fig.4 Sample Images from Face Database

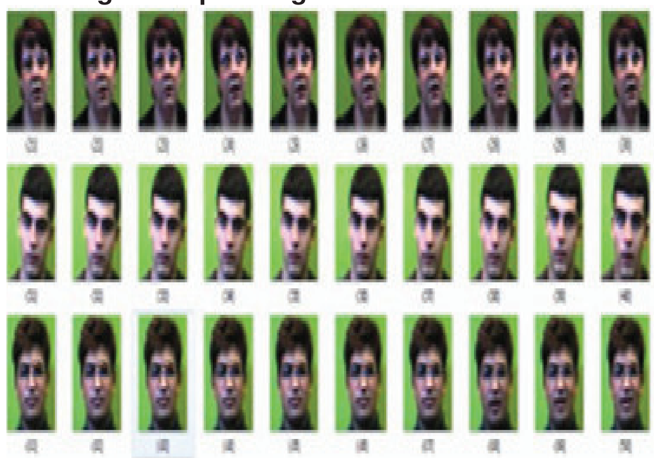

Table.1 GAR values at different BTC levels MLBTC Intermediate-4, MLBTC-Intermediate-9 and AFRR

Name	Recognition in %			
	BTC Intermediate-4	BTC Intermediate-9	MLBTC	AFRR
BTC-Level 1	95.66	96.94	97.04	98.2
BTC-Level 2	97.40	97.55	98.01	98.56
BTC-Level 3	97.48	97.65	98.23	99.02
BTC-Level 4	97.62	97.84	98.39	99.15

Table.2 FAR values at different BTC levels MLBTC Intermediate-4 , MLBTC-Intermediate-9 and AFRR

Name	Recognition in %			
	BTC Intermediate -4	BTC Intermediate-9	MLBTC	AFRR
BTC-Level 1	0.043	0.030	0.0296	0.018
BTC-Level 2	0.026	0.024	0.0199	0.014
BTC-Level 3	0.025	0.023	0.0177	0.017
BTC-Level 4	0.023	0.021	0.0161	0.008

From the below Table.1 gives the experimental results the AFRR produces higher recognition accuracy of 99.15% for faces partially occluded with spectacles. This shows that AFRR is more suited than the other tested BTC in recognizing faces partially occluded with spectacles. The total claims, false rejections and GAR values of the four BTC levels based face recognition methods tested on face database are shown. False rejections refer to number of relevant images which were rejected. Thus BTC-level 4 based face recognition technique gives us the least false number of rejections and highest GAR value indicating best performance. From the below Table.2 gives the experimental results the AFRR produces higher recognition accuracy of for faces partially occluded with spectacles. This shows that AFRR is more suited than the other tested BTC in recognizing faces partially occluded with spectacles. The total claims, false rejections and GAR values of the four BTC levels based face recognition methods tested on face database are shown. False rejections refer to number of relevant images which were rejected. Thus BTC-level 4 based face recognition technique gives us the least false number of rejections and highest GAR value indicating best performance. From the below figure 4 shows the pictorial representation of the performance evaluated. By analyzing the obtained results the JPG produced the best results.

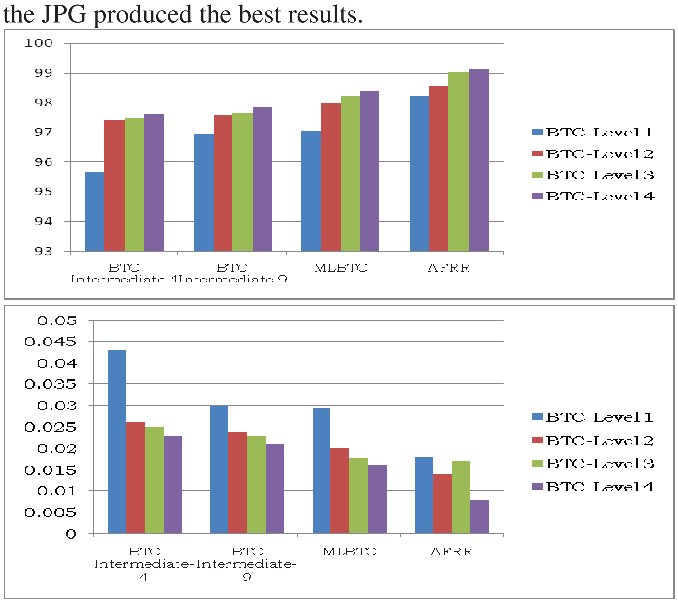

Fig 5. Performance Evaluation

7. CONCLUSION

Digital Image Processing is on emerging field in the area of computer science and other application area. Digital Image Processing compression plays a vital on role of all field. A new image compression scheme based on BTC and maximum-minimum pixels value has been proposed. The face recognition of color features capture the face and geometrical features describe the shape details of the facial components. But when they are applied alone face recognition may not produce good results. This paper proposes to mixture of two features to enhance the accuracy of face recognition. The performance of color feature extraction methods and geometrical methods are investigated independently and combined together when compared existing methods.

8. REFERENCES

[1] Aditya Kumar,Pardeep Singh, "Enhanced Block Truncation Coding for Gray Scale Image" Int. J. Comp. Tech. Appl., Vol 2 (3), 525-529,2012-ISSN:2229-6093.

[2] Gaganpreet Kaur and Manjinder Kaur "A Survey of Lossless and Lossy Image Compression Techniques". International Journal of Advanced Research in Computer Science. Volume 3, Issue 2, February 2013: ISSN: 2277 128X.

[3] G. Prabhakar and B. Ramasubramanian, "An Integrated an Efficient Approach for Enhanced Medical Image Compression using SPIHT and LZW Coding", International Journal of Scientific & Engineering Research Volume 4, Issue 2, February-2013.

[4] Jun Wang and Jong-Wha Chong, Member, IEEE "High Performance Overdrive Using Improved Motion Adaptive Codec in LCD" IEEE Transactions on Consumer Electronics, Vol. 55, No. 1, FEBRUARY 2009.

[5] Lucas Hui, "An Adaptive Block Truncation Coding Algorithm for Image Compression", IEEE, PP.2233-2235,1990.

[6] Maximo D.Lema, O.Robert Mitchell, "Absolute Moment Block Truncation Coding and its Application to Color Images", IEEE Transactions on Communications, VOL. COM-32, NO. 10, October 1984.

[7] Meftah M. Almrabet , Amer R. Zerek, Allaoua Chaoui Ali A. Akash "Image compression using block truncation coding" IJ-STA, Volume 3, No 2, December 2009.

[8] Pravin B. Pokle and Dr. Narendra.G. Bawane, "Comparative Study of Various Image Compression Techniques", International Journal of Engineering Research, Volume 4, Issue 5, May-2013.

[9] Rupinder Kaur, Nisha Kaushal, "Comparative Analysis of various Compression Methods for Medical Images". National Institute of Technical Teachers' Training and Research, Panjab University Chandigarh, 2007.

[10] K.Somasundaram and I.Kaspar Raj, "Low Computational Image Compression Scheme based on Absolute Moment Block Truncation Coding", World Academy of Science, Engineering and Technology, Vol. 19, pp. 166-171, 2006.

[11] K.Somasundaram, S.Vimala "Multi-Level Coding Efficiency with Improved Quality for Image Compression based on AMBTC" International Journal of Information Sciences and Techniques (IJIST) Vol.2, No.2, March 2012.

[12] K.Somasundram and S.Vimala, "Efficient Block Truncation Coding", International Journal on Computer Science and Engineering, Vol. 02, No. 06, 2010, 2163-2166.

[13] S.Vimala, P. Uma, B. Abidha "Improved Adaptive block truncation coding for image compression" international journal of computer application (0975-8887) vol 19-No.7, April 2011.

[14] Developed by Dr. Libor Spacek. Available Online at: http://cswww.essex.ac.uk/mv/otherprojects.html.

[15] Rafael C. Gonzalez and Richard Eugene Woods "Digital Image Processing", 3rd edition, Prentice Hall.