IJEEE, Vol. 1, Issue 1 (Jan-Feb 2014) e-ISSN: 1694-2310 | p-ISSN: 1694-2426

DESIGN OF IMAGE COMPRESSION ALGORITHM USING MATLAB

Abhishek Thakur[1], Rajesh Kumar[2], Amandeep Bath[3], Jitender Sharma[4]
[1,2,3,4]Electronics & Communication Department, IGCE, Punjab, India
[1]abhithakur25@gmail.com, [2]errajeshkumar2002@gmail.com,
[3]amandeep_batth@rediffmail.com, [4]er_jitender2007@yahoo.co.in

Abstract- This paper gives the idea of recent developments in the field of image security and improvements in image security. Images are used in many applications and to provide image security using image encryption and authentication. Image encryption techniques scramble the pixels of the image and decrease the correlation among the pixels, such that the encrypted image cannot be accessed by unauthorized user. This study proposes a method for encrypting the sender's messages using new algorithm called chaotic encryption method. This key will be used for encrypting and decrypting the messages which are transmitted between two sides. Chaotic encryption technique is the new way of cryptography. Emphasizing the image security, this paper is to design the enhanced secure algorithm which uses chaotic encryption method to ensure improved security and reliability.

Index Terms- Cryptography, Cryptography key, RGB image, LSB, image key.

I. INTRODUCTION

In telegraphy Morse code, is a data compression technique using shorter codewords for letters invented in 1838. In late 1940s information theory is invented. In 1949 Claude Shannon and Robert Fano develop a way to assign code words based on probabilities of blocks. Later an optimal method for doing this was developed by David Huffman in 1951. Early implementations were done in hardware, with specific choices of codewords being made as compromises between error correction and compression. In the mid-1970s, Huffman encoding for dynamically updating codewords based on the actual data encountered is invented. Pointer based encoding is invented in 1977 by Abraham Lempel and Jacob Ziv. LZW algorithm is invented in the mid-1980s, which became the method for most general purpose compression systems. It was used in programs, as well as in hardware devices such as modems [1]. In the early 1990s, lossy compression method is widely used. Today image compression standards include: FAX CCITT 3 (run-length encoding, with codewords determined by Huffman coding from a definite distribution of run lengths); BMP (run-length encoding, etc.); JPEG (lossy discrete cosine transform, then Huffman or arithmetic coding); GIF (LZW); TIFF (FAX, JPEG, GIF, etc.)[2]. Typical compression ratios currently achieved for text are around 3:1, and for photographic images around 2:1 lossless, and for line diagrams and text images around 3:1, and 20:1 lossy. Images on internet grow very fast and hence each server needs to store high volume of data about the images and images are one of the most important data about information. As a result servers have a high volume of images with them and require a huge hard disk space and transmission bandwidth to store these images. Most of the time transmission bandwidth is not sufficient for storing all the image data. Image compression is the process of encoding information using fewer bits. Compression is useful because it helps to reduce the consumption of expensive resources, such as transmission bandwidth or hard disk space. Compression scheme for image may require expensive hardware for the image to be decompressed fast enough to be viewed as its being decompressed [3]. Compression schemes therefore involves trade-offs among various factors, including the degree of compression, the amount of distortion introduced, and the computational resources required to compress and uncompress the data. Image compression is the application of Data compression. Image compression reduce the size in bytes of a graphics file without degrading the quality of the image. The purpose of reduction in file size allows more images to be stored in a given amount of disk or memory space. It also reduces the time required for images to be sent over the Internet or downloaded from Web pages. There are several different ways in which image files can be compressed. For Internet use, the two most common compressed graphic image formats are the JPEG format and the GIF format. The JPEG method is more often used for photographs, while the GIF method is commonly used for line art and other images in which geometric shapes are relatively simple [4-6].

Other techniques for image compression include the use of fractals and wavelets. These methods have not gained widespread acceptance for use on the Internet as of this writing. However, both methods offer promise because they offer higher compression ratios than the JPEG or GIF methods for some types of images. Another new method that may in time replace the GIF format is the PNG format [7]. With a compression ratio of 32:1, the space, bandwidth and transmission time requirements can be reduced by the factor of 32, with acceptable quality. Image compression and coding techniques explore three types of redundancies:

A. Coding redundancy:

Coding redundancy is present when less than optimal code words are used. This type of coding is always reversible and usually implemented using look up tables (LUTs) [8]. Examples of image coding schemes that explore coding redundancy are the Huffman codes and the arithmetic coding technique.

B. Inter pixel redundancy:

This type of redundancy sometimes called spatial redundancy, inter frame redundancy, or geometric redundancy. This redundancy can be explored in several ways, one of which is by predicting a pixel value based on the values of its neighboring pixels. In order to do so, the original 2-D array of pixels is usually mapped into a different format, e.g., an array of differences between adjacent pixels. If the original image pixels can be reconstructed from the transformed data set the mapping is said to be reversible. Examples of compression techniques that explore the interpixel redundancy include: Constant Area Coding (CAC), (1-D or 2-D) Run Length Encoding (RLE) techniques, and many predictive coding algorithms such as Differential Pulse Code Modulation (DPCM) [9].

C. Psycho visual redundancy:

Many experiments on the psychophysical aspects of human vision have proven that the human eye does not respond with equal sensitivity to all incoming visual information; some pieces of information are more important than others. The knowledge of which particular types of information are more or less relevant to the final human user have led to image and video compression techniques that aim at eliminating or reducing any amount of data that is psycho visually redundant. The end result of applying these techniques is a compressed image file, whose size and quality are smaller than the original information, but whose resulting quality is still acceptable for the application at hand. The loss of quality that ensues as a byproduct of such techniques is frequently called quantization, as to indicate that a wider range of input values is normally mapped into a narrower range of output values thorough an irreversible process [10]. In order to establish the nature and extent of information loss, different fidelity criteria (some objective such as root mean square (RMS) error, some subjective, such as pair wise comparison of two images encoded with different quality settings) can be used. Most of the image coding algorithms in use today exploit this type of redundancy, such as the Discrete Cosine Transform (DCT)-based algorithm at the heart of the JPEG encoding standard.

II. TECHNIQUES USED FOR IMAGE COMPRESSION

A. Image Compression Model:

Image compression system is composed of two distinct functional components: an encoder and a decoder. The encoder performs the complementary operation of compression, and the decoder performs the complementary operation of decompression. Both compression and decompression operations can be performed in software, as in the case in web browsers and many commercial image editing programs, or in a combination of hardware and firmware, as in commercial DVD players [11]. A codec is a device or program that is capable of both encoding and decoding.

Input image f(x, y) is fed into the encoder, which creates a compressed representation of the input. This representation is stored for latter use, or transmitted for storage and use at a remote location. When the compressed representation is presentation is presented to its complementary decoder, a reconstructed output image f^(x, y) is generated. In general, f^(x,y) may or may not be a replica of f(x,y). If it is, the compression system is called error free, lossless or information preserving. If not, the reconstructed output image is distorted and the compression system is referred to as lossy.

B. The encoding or compression process:

The encoder is designed to remove the redundancies. In the first stage of the encoding process, a mapper transforms f(x, y) into a format designed to reduce spatial and temporal redundancy. This operation generally is reversible and may or may not reduce directly the amount of data required to represent the image. The run length coding is an example of the mapping that normally yields compression in the first step of encoding process. The quantizer reduces the accuracy of the mapper's output in accordance with per-established fidelity criterion. The goal is to keep irrelevant information out of the compressed representation. This operation is irreversible. It must be omitted when error-free compression is desired. In the third and final stage of the encoding process, the symbol coder generates a fixed or variable length code to represent the quantizer output and maps the output in accordance with the code [12]. In many cases, a variable –length code is used. The shortest code words are assigned to the most frequently occurring quantizer output values- thus minimizing coding redundancy. This operation is irreversible. Upon its completion, the input image has been processed for the removal of each of the three redundancies (Coding, Interpixel , Psycho visual).

C. The Decoding or decompression process:

The decoder contains only two components: a symbol decoder and an inverse mapper. They perform, in reverse order, the inverse operation of encoder's symbol encoder and mapper. Because quantization results in irreversible information loss, an inverse quantizer block is not included in the general decoder model [13].

D. Image Compression Algorithms:

Image compression can be lossy or lossless. Lossless compression is sometimes preferred for artificial images such as technical drawings, icons or comics. This is because lossy compression methods, especially when used at low bit rates, introduce compression artifacts. Lossless compression methods may also be preferred for high value content, such as medical imagery or image scans made for archival purposes. Lossy methods are especially suitable for natural images such as photos in applications where minor (sometimes imperceptible) loss of fidelity is acceptable to achieve a substantial reduction in bit rate [14-15].

E. Various Lossy Compression Methods are:

1) Cartesian Perceptual Compression: Also known as CPC
2) DjVu

3) Fractal compression
4) HAM, hardware compression of color information used in Amiga computers
5) ICER, used by the Mars Rovers: related to JPEG 2000 in its use of wavelets
6) JPEG
7) JPEG 2000, JPEG's successor format that uses wavelets.
8) JBIG2
9) PGF, Progressive Graphics File (lossless or lossy compression) Wavelet compression.

F. Various Loss-Less Compression Method are:

1) Run-length encoding – used as default method in PCX and as one of possible in BMP, TGA, TIFF
2) Entropy coding
3) Adaptive dictionary algorithms such as LZW – used in GIF and TIFF
4) Deflation – used in PNG, MNG and TIFF.

G. The steps involved in compressing and decompressing of image are:

1) Specifying the Rate (bits available) and Distortion (tolerable error) parameters for the target image.
2) Dividing the image data into various classes, based on their importance.
3) Dividing the available bit budget among these classes, such that the distortion is a minimum.
4) Quantize each class separately using the bit allocation information derived in step 3.
5) Encode each class separately using an entropy coder and write to the file.
6) Reconstructing the image from the compressed data is usually a faster process than compression. The steps involved are step 7 to step 9.
7) Read in the quantized data from the file, using an entropy decoder. (Reverse of step 5).
8) Dequantized the data. (Reverse of step 4).
9) Rebuild the image. (Reverse of step 2).

H. Calculation of image compression:

There are various types of terms that are used in calculation of image compression. Some are listed below:

a. Cartesian Perceptual Compression: Also known as CPC
b. Compression Ratio = Uncompressed Size/Compressed Size
c. Space Saving = [1- compressed Size/Uncompressed Size]
d. Data Rate Savings = [1- compressed data rate/Uncompressed data rate]

The PSNR is most commonly used as a measure of quality of reconstruction in image compression etc. It is most easily defined via the mean squared error (MSE) which for two $m\times n$ monochrome images I and K where one of the images is considered a noisy approximation of the other is defined as:

$$MSE = \frac{1}{MN}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}\left\| I(i,j) - K(i,j) \right\|^2$$

The PSNR is defined as:

$$PSNR = 10\log_{10}\left(\frac{MAX_1^2}{MSE}\right) = 20\log_{10}\left(\frac{MAX_1}{\sqrt{MSE}}\right)$$

Here, MAX_i is the maximum possible pixel value of the image. When the pixels are represented using 8 bits per sample, this is 255. More generally, when samples are represented using linear PCM with B bits per sample, MAX_I is 2^B-1.

Signal-to-noise ratio is a term for the power ratio between a signal (meaningful information) and the background noise:

$$SNR = \frac{P_{Signal}}{P_{Noise}} = \left(\frac{A_{Signal}}{A_{Sinal}}\right)^2$$

If the signal and the noise are measured across the same impedance then the SNR can be obtained by calculating 20 times the base-10 logarithm of the amplitude ratio:

$$SNR(db) = 10\log_{10}\left(\frac{P_{Signal}}{P_{Noise}}\right) = 20\log_{10}\left(\frac{A_{Signal}}{A_{Noise}}\right)$$

The MSE of an estimator θ′with respect to the estimated parameter θ is defined as:

MSE (θ′)=E((θ′-θ)2)

The MSE can be written as the sum of the variance and the squared bias of the estimator:

MSE (θ′)=Var(θ′)+(Bias(θ′,θ))2

In a statistical model where the estimand is unknown, the MSE is a random variable whose value must be estimated. This is usually done by the sample mean, with θ_j being realizations of the estimator θ′of size *n*.

$$MSE'(\theta') = \frac{1}{n}\sum_{j=1}^{n}(\theta_j - \theta)^2$$

III. COMPRESSION TECHNOLOGIES

A. All Block Truncation Coding:

Block Truncation Coding (BTC) is a well known compression scheme proposed in 1979 for the greyscale. It was also called the moment preserving block truncation because it preserves the first and second moments of each image block. It is a lossy image compression technique. It 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. The truncated block of the BTC is the one-bit output of the quantizer for every pixel in the block [1].

B. Wavelet Transform:

A wavelet image compression system can be created by selecting a type of wavelet function, quantizer, and statistical coder. The first step in wavelet compression are performing a discrete wavelet Transformation (DWT), after that quantization of the wavelet space image sub bands, and then encoding these sub bands. Wavelet images are not compressed images; rather it is quantization and encoding stages that do the image compression. Image decompression, or reconstruction, is achieved by carrying out the above steps in reverse and inverse order [4].

C. EZW Algorithm:

In embedded coding, the coded bits are ordered in accordance with their importance and all lower rate codes are provided at the beginning of the bit stream. Using an embedded code, the encoder can terminate the encoding process at any stage, so as to exactly satisfy the target bit-rate specified by the channel. To achieve this, the encoder can maintain a bit count and truncate the bit-stream, whenever the target bit rate is achieved. Although the embedded coding used in EZW is more general and sophisticated than the simple bit-plane coding, in spirit, it can be compared with the latter, where the encoding commences with the most significant bit plane and progressively continues with the next most significant bit-plane and so on. If target bit-rate is achieved before the less significant bit planes are added to the bit-stream, there will be reconstruction error at the receiver, but the “significance ordering” of the embedded bit stream helps in reducing the reconstruction error at the given target bit rate [2].

D. Fractal Image Compression and Decompression:

Fractal compression is a lossy image compression method using fractals to achieve high levels of compression. The method is best suited for photographs of natural scenes (trees, mountains, ferns, clouds).

The fractal compression technique relies on the fact that in certain images, parts of the image resemble other parts of the same image. Fractal algorithms convert these parts, or more precisely, geometric shapes into mathematical data called "fractal codes" which are used to recreate the encoded image. Fractal compression differs from pixel-based compression schemes such as JPEG, GIF and MPEG since no pixels are saved. Once an image has been converted into fractal code its relationship to a specific resolution has been lost. The image can be recreated to fill any screen size without the introduction of image artifacts that occurs in pixel based compression schemes.

IV. RESULTS

Matlab approach has been implemented for compressing images. Matlab is capable of handling simultaneous multiple matrix of images. In this firstly given RGB image is divided into three matrices which are compressed individually and then these are merged together to construct final compressed image. Image processing tool box of matlab is used to perform all compression task.

Matlab based environment ensures high scalability, availability and reliability through redundancy mechanisms. Hence matlab computing proves to be an appropriate platform for compressing images by various techniques. Matlab provides High-level language for technical computing and Development environment for managing code, files and data. It is an Interactive tool for iterative exploration, design and problem solving. It provides Mathematical functions for linear algebra, statistics, Fourier analysis, filtering, optimization, numerical integration and Tools for building custom graphical user interfaces.

In this study compressing of images is taken by splitting image matrix into R,G,B matrix which are compressed individual and then these are merged for construction of compressed image. It is an efficient approach for performing compression for large size images of different format. Time and space is reduced to great extent with the help of Matlab computing.

An RGB image, sometimes referred to as a true color image, is stored as an m-by-n-by-3 data array that defines red, green and blue color components for each individual pixel. RGB images do not use a palette.

The colour of each pixel is determined by the combination of the red, green, and blue intensities stored in each colour plane at the pixel's location. Graphics file formats store RGB images as 24-bit images, where the red, green, and blue components are 8 bits each. This yields a potential of 16 million colours.

A. Block Truncation Coding:

In table 1 we take the block size of 8*8, 64*64 and perform block truncation coding. These results are shown in figure 1.

Table 1: Block Truncation Coding

Block Size	8*8	64*64
MSE	3.5077e+003	3.2613e+003
PSNR	12.6805	12.9969

a)Original Image b) 8*8 Block Truncation c) 64*64 Block Truncation

Fig. 1: Block Truncation Coding

B. Wavelet Compression:

In table 2 Wavelet Compression encode time, MSE and PSNR is shown with Decomposition Level 3 and 7. Figure 2 shows results for Wavelet Compression.

Table 2: Wavelet Compression

Decom.	4	7

Level		
Encode time	35.188 Sec.	35.359 Sec.
MSE	3.5609e+003	3.6379e+003
PSNR	12.6152	12.5224

a) Original b) Level 4 c) Level 7

Fig. 2: Wavelet Compression

C. Embedded Zerotree:

In table 3 Embedded Zerotree encode time, decode time, MSE and PSNR is shown with decomposition level 4 and 6. Figure 3 shows the results for Embedded Zerotree.

Table 3: Embedded Zerotree

Decom. Level	4	6
Encode time	237.7180 Sec.	243.8910 Sec.
Decode time	119.9070 Sec.	123.4360 Sec.
MSE	3.4228e+003	3.4228e+003
PSNR	12.7870	12.7870

a) Original b) Level 4 c) Level 6

Fig. 3: Embedded Zerotree

D. Fractal Image Compression:

In table 4 Fractal Image Compression encode time, decode time, MSE and PSNR is shown with increase in block size of 10 and 25. Figure 4 shows the results for Fractal Image Compression.

Table 4: Embedded Zerotree

Decom. Level	4	25
Encode time	300.0610 Sec.	302.310 Sec.
Decode time	41.0320 Sec.	40.703 Sec.
MSE	3.5205e+003	2.9751e+003
PSNR	12.6648	13.3957

a) Original b) Block Size 2 c) Block Size 25

Fig. 4: Embedded Zerotree

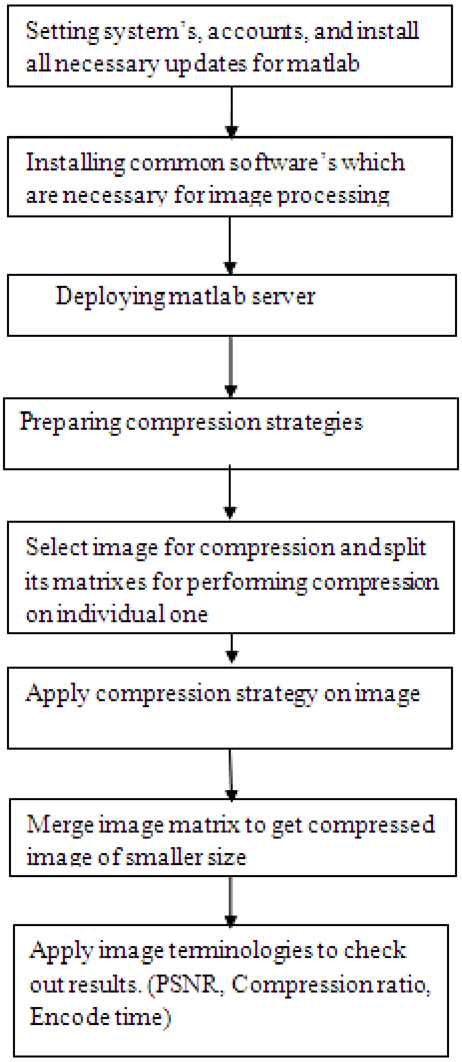

Fig. 5 Flowchart for implementation of compressing image in matlab Environment

Different types of image are taken for discussing their results. In our study four different types of images are taken:

a. Face image

Encode Process Time as Second of red =95.9850
Decode Process Time as Second of red =13.5620
Encode Process Time as Second of green =98.1410

Decode Process Time as Second of green =13.7970
Encode Process Time as Second of blue =96.9690
Decode Process Time as Second of blue =13.6560
Total encode time =291.0950
Total decode time =41.0150
Cr =6.2036
Encode_time =291.095seconds
The MSE performance is 56.72 dB
The psnr performance is 30.59 dB

Fig. 6 Face Image

b. Scene image:
Encode Process Time as Second of red =97.8600
Decode Process Time as Second of red =13.5630
Encode Process Time as Second of green =95.1560
Decode Process Time as Second of green =13.4220
Encode Process Time as Second of blue =95.9370
Decode Process Time as Second of blue =13.4690
Total encode time =288.9530
Total decode time =40.4540
Cr =1.0033
Encode time =288.953seconds
The MSE performance is 10.65 dB
The psnr performance is 37.86 dB

Fig. 7 Scene image

V. CONCLUSIONS

Images play an important role in our lives. They are used in many applications. In our study we have applied different type of compression technique on different type of images for a PSNR value and compression ratio. In this paper, the problem of compressing an image in Matlab environment has been taken. A matlab compression approach has been implemented for compressing images.

After analysis we have found that, scene Images Wavelet provides the better result but compression ratio is high in case of BTC and visual quality is better of BTC. For face Images BTC perform the most compression. FIC provide almost same result as compare to BTC, but fractal image compression take a long time for compression, so we can use BTC as compare to FIC when time in main concern. We analyzed the PSNR obtained of compressed after each compression technique and decides which technique can provide maximum PSNR for a particular image.

We have done PSNR measures, but should also use objective and subjective picture quality measures. The objective measures such as PSNR and MSE do not correlate well with subjective quality measures. Therefore, we should use PQS as an objective measure that has good correlation to subjective measurements. After this we will have an optimal system having best compression ratio with best image quality.

REFERENCES

[1]. A. Kumar, P. Singh, "Comparative study of Block Truncation Coding with EBTC", International Journal of Advances in Computer Networks and its Security, pp: 395-405, 2011.
[2]. D. Mohammed, F. Abou-Chadi, "Image Compression Using Block Truncating Code", JSAT, pp 9 – 1, 2011.
[3]. Q. Guan, "Research of Image Compression Based on Embedded Zero-tree Wavelet Transform", IEEE, pp: 591-595, 2011.
[4]. S. A. Abdul, k. B. A. Karim," Wavelet transform and fast Fourier transform for signal compression: a comparative study", 2011 international conference on electronic devices, systems and applications, IEEE, pp: 280-285, 2011.
[5]. P. aggarwal, B. Rani, "Performance Comparison of Image Compression Using Wavelets", IJCSC, IEEE, pp: 97-100, 2010.
[6]. Chunlei Jiang, Shuxin Yin,"A Hybrid Image Compression Algorithm Based on Human Visual System" International Conference on Computer Application and System Modeling, IEEE, pp: 170-173, 2010.
[7]. L. Hengjian, W. Jizhi, W. Yinglong, T. Min, X. Shujiang, "A flexible and secure image compression coding algorithm", International Conference on Future Information Technology and Management Engineering, IEEE, pp: 376-379, 2010.
[8]. S.P. Raja, A. Suruliandi, "Analysis of Efficient Wavelet based Image Compression Techniques", Second International conference on Computing, Communication and Networking Technologies, IEEE, pp: 1-6, 2010.
[9]. Shukla, M. Alwani, A. K. Tiwari, "A survey on lossless image compression methods" 2nd international conference on computer engineering and technology, IEEE vol 6, pp: 136-141, 2010.
[10]. X. Wu, W. Sun "Data Hiding in Block Truncation Coding", International Conference on Computational Intelligence and Security, IEEE, pp: 406-410, 2010.
[11]. T. Y. Mei, T. J. Bo, "A Study of Image Compression Technique Based on Wavelet Transform", Fourth International Conference on Genetic and Evolutionary Computing 2010 IEEE, 2010.
[12]. J. Bhattacharjee, "A Comparative study of Discrete Cosine Transformation, Haar Transformation, Simultaneous Encryption and Compression Techniques, International Conference on Digital Image Processing", ICDIP, 2009 Proceedings of the International Conference on Digital Image Processing, Pages 279-283,2009.
[13]. B. Rani, R. K. Bansal and S. Bansal , "Comparison of JPEG and SPIHT Image Compression Algorithms using Objective Quality Measures" , Multimedia, Signal Processing and Communication Technologies, IMPACT 2009, IEEE, 2009, Page no 90-93, 2009.
[14]. V. P. Lineswala , J. N. patel, "JPEG image compression and transmission over wireless channel", International Conference on Advances in Computing, Control, and Telecommunication Technologies, IEEE, pp: 643-645, 2009.
[15]. M. Grossber, I. Gladkova, S. Gottipat , M. Rabinowitz, P. Alabi, T. George1, and A.Pacheco, "A Comparative Study of Lossless Compression Algorithms on Multi-Spectral Imager Data", Data Compression Conference, IEEE, pp:447, 2009.

AUTHORS

First Author– Abhishek Thakur: M. Tech. in Electronics and Communication Engineering from Punjab Technical University, MBA in Information Technology from Symbiosis Pune, M.H. Bachelor in Engineering (B.E.- Electronics) from

Shivaji University Kolhapur, M.H. Five years of work experience in teaching and one year of work experience in industry. Area of interest: Digital Image and Speech Processing, Antenna Design and Wireless Communication. International Publication: 7, National Conferences and Publication: 6, Book Published: 4 (Microprocessor and Assembly Language Programming, Microprocessor and Microcontroller, Digital Communication and Wireless Communication). Working with Indo Global College of Engineering Abhipur, Mohali, P.B. since 2011.
Email address: abhithakur25@gmail.com

Second Author – Rajesh Kumar is working as Associate Professor at Indo Global College of Engineering, Mohali, Punjab. He is pursuing Ph.D from NIT, Hamirpur, H.P. and has completed his M.Tech from GNE, Ludhiana, India. He completed his B.Tech from HCTM, Kaithal, India. He has 11 years of academic experience. He has authored many research papers in reputed International Journals, International and National conferences. His areas of interest are VLSI, Microelectronics and Image & Speech Processing.

Third Author – Amandeep Batth: M. Tech. in Electronics and Communication Engineering from

Punjab Technical University, MBA in Human Resource Management from Punjab Technical University , Bachelor in Technology (B-Tech.) from Punjab Technical University . Six years of work experience in teaching. Area of interest: Antenna Design and Wireless Communication. International Publication: 1, National Conferences and Publication: 4. Working with Indo Global College of Engineering Abhipur, Mohali, P.B. since 2008.
Email address: amandeep_batth@rediffmail.com

Forth Author – Jitender Sharma: M. Tech. in Electronics and Communication Engineering from Mullana University, Ambala, Bachelor in Technology (B-Tech.)from Punjab Technical University . Five years of work experience in teaching. Area of interest:, Antenna Design and Wireless Communication. International Publication: 1 National Conferences and Publication:6 and Wireless Communication). Working with Indo Global college since 2008.
E-mail address:er_jitender2007@yahoo.co.in