International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2015): 78.96 | Impact Factor (2015): 6.391

Compression of Remotely Sensed Images using Hadmard Transform and (AMBTC)

Dr. Ban Abd.Al-Razak[1], Aliaa Hussein Shukhair[2], Zena F. Rasheed[3]

[1]Department of Astronomy and Space, College of Science, University of Baghdad

[2]Department of Astronomy and Space, College of Science, University of Baghdad

[3]Department of Petroleum Engineering, College of Engineering, University of Baghdad

Abstract: ***The fast technology improvement in communication and information interchange systems, in the last few decades, has led a big problem. The problem a raised when modern machines produced enormous amount of data and the requirement for fast handling becomes necessary. Therefore, image coding techniques and data compression methods have recently appeared to solve the above mentioned problem. In order to reduce the volume of pictorial data which one may need to store or transmit, the research modifies a method for image data compression based on the two-component code; in this coding technique, the image is partitioned into regions of slowly varying intensity. The contours separating the regions are coded by hadmard transform, while the rest image regions are coded by (AMBTC).***

Keywords: compression, hadmard transform, AMBTC

1. Introduction

The science of remote sensing consists of the analysisand interpretation of measurements of electromagneticradiation (EMR) that is reflected from or emitted by atarget and observed or recorded from a vantage point byan observer or instrument that is not in contact with thetarget. Earth observation by remote sensing (EO) is theinterpretation and understanding of measurements madeby airborne or satellite-borne instruments of EMR thatis reflected from or emitted by objects on the Earth'sland, ocean or ice surfaces or within the atmosphere,together with the establishment of relationships betweenthese measurements and the nature and distribution ofphenomena on the Earth's surface or within the atmosphere. [1].

A digital image is a rectangular array of dots, or pixels, arranged in m rows and n columns. A digital image is represented by a two-dimensional array of pixels, which are arranged in rows and columns. Hence, a digital image can be presented as M×N array [2].

$$f(x,y)=\begin{bmatrix} f(0,0) & f(0,1) & \cdots & f(0,N-1) \\ f(1,0) & f(1,1) & \cdots & f(1,N-1) \\ \vdots & \vdots & \ddots & \vdots \\ f(M-1,0) & f(M-1,1) & \cdots & f(M-1,N-1) \end{bmatrix}$$

Where f (0, 0) gives the pixel of the left top corner of the array that represents the image and f (M- 1, N-1) represents the right bottom corner of the array. A Grey-scale image, also referred to as a monochrome image contains the values ranging from 0 to 255, where 0 is black, 255 is white and values in between are shades of grey. In color images, each pixel of the array is constructed by combining three different channels (RGB) which are R=red, G=green and B=blue. Each channel represents a value from 0 to 255. In digital image, each pixel is stored in three bytes, while in a Grey image is represented by only one byte. Therefore, color images take three times the size of Gray images.

image coding is based on a two –components; the edge effect component coded by hadmard transform, while the rest region component will be coded by the absolute moment block truncation coding method. Also experimental result of applying hadmard transform and absolute moment block truncation coding (AMBTC) included in this paper.

2. Theoretical Concept of Edge Detection

Edge detection is a fundamental tool in computer vision. It is the front-end processing stage in object recognition and image understanding systems. The accuracy with which this task can be performed is a crucial factor in determining overall system performance. Since detecting primitive edges is an important task in early vision process. Many different edge detectors have been proposed. Most of them consist of two sub-processes: feature extraction and determination of edge points.

Edge detection is an important work for object recognition and is also an essential pre-processing step in image segmentation [3].

The edge detection of digital image is quite important foundation in the field of image analysis including image division, identification of objective region and so on.The goal of edge detection is to mark the points in a digital image at which the luminous intensity changes sharply. Sharp changes in image properties usually reflect important events and changes in objects included in the image. Edge detection preserves the important structural properties of an image and acts in a similar way as a filter do for irrelevant data [4].The purpose of detecting sharp changes in image brightness is to capture important events and changes in properties of the world. It can be shown that under rather general assumptions for an image formation model, discontinuities in image brightness are likely to correspond to [5]:

- discontinuities in depth
- discontinuities in surface orientation
- changes in material properties and
- Variations in scene illumination.

Four well-known edge- detection techniques i.e. Sobel,Lablacian, Roberts, Prewitt operator and Canny operator.in this work we adopted Canny operator .

Canny operator

The Canny-Deriche detector was derived from similar mathematical criteria as the Canny edge detector, although starting from a discrete viewpoint and then leading to a set of recursive filters for image smoothing instead of exponential filters or Gaussian filters[6].

The Process of Canny edge detection algorithm can be broken down to 5 different steps:

1) Apply Gaussian filter to smooth the image
2) Find the intensity gradients of the image
3) Apply non-maximum suppression
4) Apply double threshold to determine potential edges
5) Track edge by hysteresis.

Image compression

The development of multimedia and digital imaging has led to high quantity of data required to represent modern imagery. This requires high capacity for storage, and long time for transmission over computer networks, Image data compression addresses the problem of reducing the amount of data required to represent a digital image.Image compression techniques reduce the number of bits required to represent an image by taking advantage of these redundancies. An inverse process called decompression (decoding) is applied to the compressed data to get the reconstructed image [7]. Image compression methods aim at representing an approximation of original images with as few bits as possible while controlling the quality of these representations. Now-a-days, image compression techniques are very common in a wide area of researches. Compression methods can be either reversible (i.e. Lossless) or irreversible (lossy), depending whether images are exactly reconstructed after decoding or some distortion is introduced.

Fig.(1), represents the block diagram of image compression, in which $f(x,y)$ is a digital image. It is compressed using a suitable compression algorithm. Then it is transmitted over a channel to the receiver end. At the receiver end the data is retrieved from the storage device for decompression. The decompressed data will yield $f(x,y)$. here $f(x,y)$ is original image and $f^{\wedge}(x,y)$ is the reconstructed image [8].

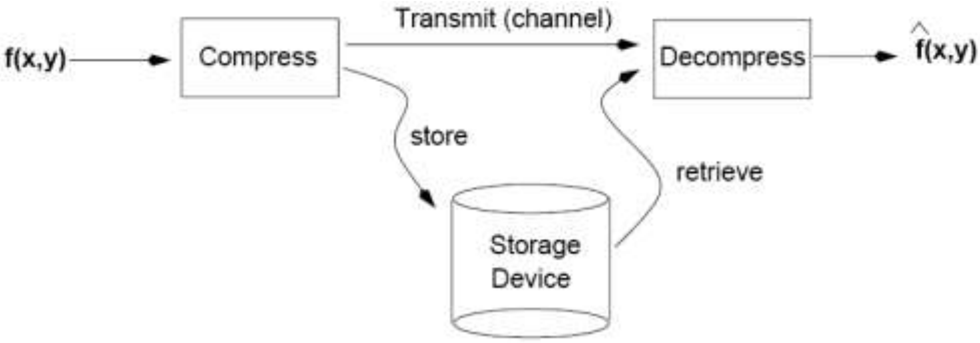


Figure 1: Block Diagram of Image Compression [8]

The Hadmard Transform : -

Unlike the DFT, the elements of the basis vectors of the Hadamard transform take only the binary values ± 1 and are, therefore, well suited for digital signal processing. The Hadamard transform matrices, H_n is N×N matrix, where N $=2^n$,n = 1,2,3. These matrices can be easily generated from the core matrix :

$$H_1 = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \quad (1)$$

And the kronecker product recursion

$$H = H_{n-1} \otimes H_1 = H_1 \otimes H_{n-1} = \frac{1}{\sqrt{2}}\begin{pmatrix} H_{n-1} & H_{n-1} \\ H_{n-1} & -H_{n-1} \end{pmatrix} \quad (2)$$

As an example, for n=3, the Hadamard matrix becomes :

$$H_3 = H_1 \otimes H_2$$
$$H_2 = H_1 \otimes H_1$$

Which gives : -

$$H_3 = \frac{1}{\sqrt{8}} \begin{array}{cc} \left[\begin{array}{cccc|cccc} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ \hline 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \end{array}\right] & \begin{array}{c} 0 \\ 7 \\ 3 \\ 4 \\ 1 \\ 6 \\ 2 \\ 5 \end{array} \end{array}$$

The basis vectors of the Hadamard transform can also be generated by sampling a class of function called the Walsh functions. These functions also take only the binary values ± 1 and form a complete orthonormal basis for square integral function. For this reason the Hadamard transform just defined is also called the Walsh-Hadamard transform [9].

The number of zero crossings of a Walsh function or the number of transitions in a basis vector of the Hadamard transform is called its sequence. The Hadamard transform of an N.×1 vector U is written as

$$\boldsymbol{V = HU}$$

And the inverse transform is given by

$$U = H^{-1}V$$

where $H^{-1} = H$

Where H = Hn , n = $\text{Log}_2 2^N$. in series form the transform pair becomes

$$V(K) = \frac{1}{\sqrt{N}} \sum_{m=0}^{N-1} U(m)(-1)^{b(k,m)} \qquad 0 \le k \le N-1 \tag{3}$$

$$U(m) = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} V(k)(-1)^{b(k,m)} \qquad 0 \le m \le N-1 \tag{4}$$

Where

$$b(k,m) = \sum_{i=0}^{n-1} k_i m_i \qquad k_i, m_i = 0,1$$

And $\{K_i\}$,$\{m_i\}$ are the binary representations of K and m, that is

$$k_n = k_0 + 2k_1 \ldots\ldots\ldots\ldots + 2^{n-1} k_{n-1} \tag{5}$$

$$m_n = m_0 + 2m_1 + \ldots\ldots\ldots\ldots + 2^{2-1} m_{n-1} \tag{6}$$

This transform will be adopted in our work.

Absolute Moment Block Truncation Coding (AMBTC)

The concept of absolute moment block truncation coding (AMBTC) was introduced by Lema and Mitchell in 1984, and the goal of AMBTC is to preserve the mean and the First absolute central moment of image blocks [10]. AMBTC involves less number of computations, compared to BTC as standard deviation involves more multiplications. AMBTC yields high quality of reconstructed images [11]. 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 [8]. An example for AMBTC encoding procedures are shown in Fig.(2). Fig.(2) (a) shows an image block of 4× 4 pixels. To encode this block, the mean of this block is calculate as x = 108. The value x is then taken as a threshold to generate a bit plane B, as shown in Fig.(2) (b). Pixels with values higher than or equal to x have a corresponding bit valued 1 stored in the bit plane. Otherwise, bit valued 0 is stored. Two rounded quantization levels a = 84 and b = 13 (low range (L), high rang (h)) . Once the quantization levels and the bit plane have been calculated, the compressed code of this image block comes out as a trio of [84,132, (1000 1100 1100 1110)]. All image blocks are encoded with the same manner, until all the blocks are processed.

120	86	97	93
128	124	62	84
137	142	69	94
142	150	112	88

(a) Original image block image

1	0	0	0
1	1	0	0
1	1	0	0
1	1	1	0

(b) AMBTC bit plane

132	84	84	84
132	132	84	84
132	132	84	84
132	132	132	84

(c) Reconstructed image blocks

Figure 2: Example for AMBTC encoding and decoding procedures

To decode an AMBTC-compressed image, the decoder reconstructs image blocks from the compressed code. To recover an image block, if a value i b in B is 0, then the Corresponding pixel is reconstructed by the quantization level a, Otherwise, reconstructed by the quantization level b. By recovering pixels in all image blocks, the whole compressed image can be reconstructed. Fig.(3-5)(c) shows an example of AMBTC decoded image block using the trio [84,132, (1000 1100 1100 1110) [17].

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 we have 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 also the mean of the higher range and lower range.
- Based on these means calculated 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 no overlapping blocks. The size of each non overlapping block may be *(4 x 4)* or *(8 x 8)*, etc. Calculate the average gray level of the block *(4x4)* as (for easy understanding we made the compression for *4 x 4* block. The same procedure is followed for block of any size):

$$\boldsymbol{X_H = \frac{1}{k} \sum_{X_i \ge mean}^{n} X_i} \tag{7}$$

$$\boldsymbol{X_L = \frac{1}{k} \sum_{X_i < mean}^{n} X_i} \tag{8}$$

Where k is the number of pixels whose gray level is greater than x. A binary block, denoted by b, is also used to represent the pixels. We can use "1" to represent a pixel whose gray level is greater than or equal to $x$and "0" to represent a pixel whose gray level is less than x. In the decoder, an image block is reconstructed by replacing the `1"s with X_H and the "0""s by X_L In the AMBTC, we need 16 bits to code the bit plane which is same as in the BTC. But, AMBTC requires less computation than BTC. But the result remains the same [12].

Image compression quality measurement

Image quality is measured using peak signal-to-noise ratio (PSNR) [13] as most common objective measure. Whenever the value of PSNR is high this implies good compression because it means high Signal to Noise. In other words, the signal represents the original image while the noise represents the error in reconstruction. So, a compression scheme having a high PSNR can be recognized as a better one [64].Therefore, certain performance measures have been established to compare different compression algorithms. Practically, let us denote the pixels of the original image by Pi and the pixels of the reconstructed image by Qi ($where\ 1 \leq i \leq n$),we first define the mean square error (MSE) between the two images as:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(P_i - Q_i)^2 \qquad (9)$$

It is the mathematical mean of the differences in the values for the pixels between the original and the reconstructed images. The root mean square error (RMSE) is defined

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(P_i - Q_i)^2} \qquad (10)$$

as the square root of the MSE [93]:
Hence, the PSNR can be defined as:

$$PSNR = 20\log_{10}\frac{max|P_i|}{RMSE} \qquad (11)$$

Also, the compression ratio may be defined as the ratio between the reconstructed image to the original image. Therefore, suppose P and Q are two units of a set of data representing the same information. The compression ratio, CR, is denoted as,

$$CR = \frac{Q_i}{P_i} \qquad (12)$$

The quality of a compression method can be characterized by considering a set of features which describe the usefulness of the method. These features include the bit rate, which gives the average number of bits per stored pixel of the image. Bit rate is the principal parameter of a compression technique because it measures the efficiency of the technique. Another feature is the ability to preserve the information content. A compression technique is lossless if the decoded image is exactly the same as the original one. Otherwise the technique is lossy. Most of the image compression techniques are of the latter type.

The third feature of importance in is the processing speed of the compression and decompression algorithms. In on-line applications the response times are often critical factors. In the extreme case a space efficient compression algorithm is useless if its processing time causes an intolerable delay in the image processing application. In some cases one can tolerate longercompression time if the compression operation can be done as a background task. However, fast decompression is desirable [14].

3. Experimental Result

In a new image compression technique, the image is partitioned into regions of slowly varying intensity, and the edge (contours) separating the regions. The actual performance of this technique depends highly on the edge detection algorithm used. For coding purposes, the edge detection algorithm should have the property that the contour (or edge) should be smooth so that they can be coded efficiently. At high compression ratios, SIC yields a better subjective image quality than block transform coding (BTC) methods because the objectionable block distortion is avoided. For this reason, this technique is ideal for every low bit rate coding and progressive transmission of images and is considered a possible candidate for inclusion in the upcoming MPEG – 4 standards.

In the following sections we will explained our experimental result of applying image data compression techniques

4. Edge Detection

The first step in most image processing application is to detect sharp transitions called edges, and then connect these edges to outline the desired boundaries. In short, extracting edge information and constructing boundaries by connecting edges and line segments can be considered as the essential process in most pictorial analysis and pattern recognition problems, since subsequent measurements of shape size (area or land perimeters) and texture can then be taken.

5. Image Block Subdividing Step

The next step in our coding algorithm is to group the binary edge image elements, which obtained from the first step into similar size blocks, each is considered as to be an individual image region. In our present work, different size (N×N) block are used; i.e. 4×4, 8×8, 16×16. Then we used a suitable threshold value to compare with the number of edge point in each block to decide if this block is an edge region or not and as follows:

1) No. Of edge. Point ≥ Th, then the block is edge region.
2) No. Of edge. Point ≤ Th, then the block is not edge.

Accordingly, experiments showed that larger threshold value results in an important loss of edge information, which must be avoided.

6. The Encoding Process

Up to this point, the segmented image regions are classified into two types, these are; edge region or rest region. Our major aim was directed toward the development of a coding scheme based on segmenting the digital image. In our present work, the encoding process is adopted by used two compression image data techniques to encode the same digital image, so we adopt the hadmard transform coding on edge region, while the absolute moment block truncation coding (ambtc) method is used to encode the rest region. The two compression techniques mentioned above are implemented on the original image, so the binary (edge detected) image is used to decide if this region is edge region or rest region only.

7. Decoding Procedures

In the decoder, the coding processes should be inverted in order to recover the original image. We shall demonstrate and conclude the obtained results by the mentioned encoding algorithm in the following figures

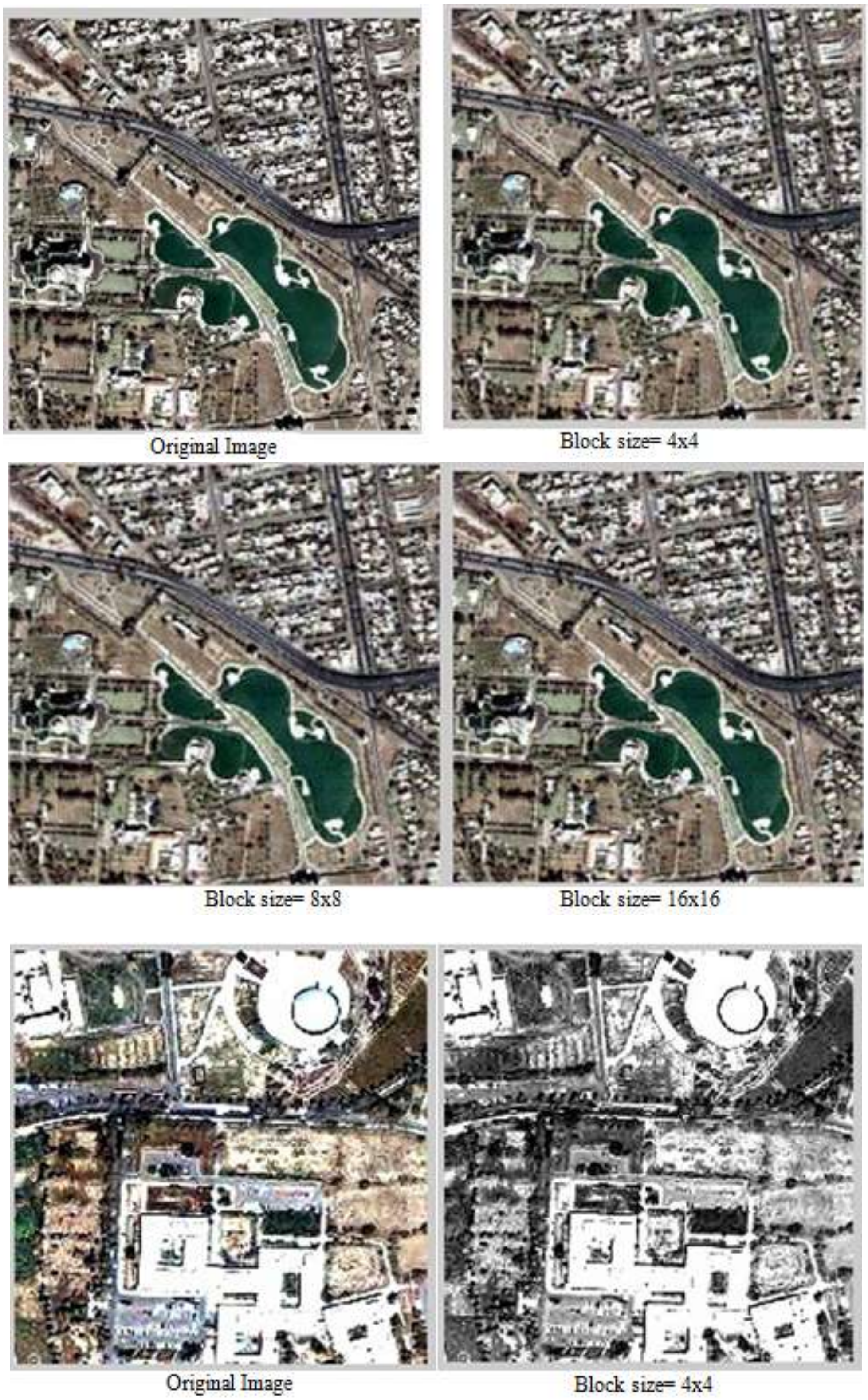
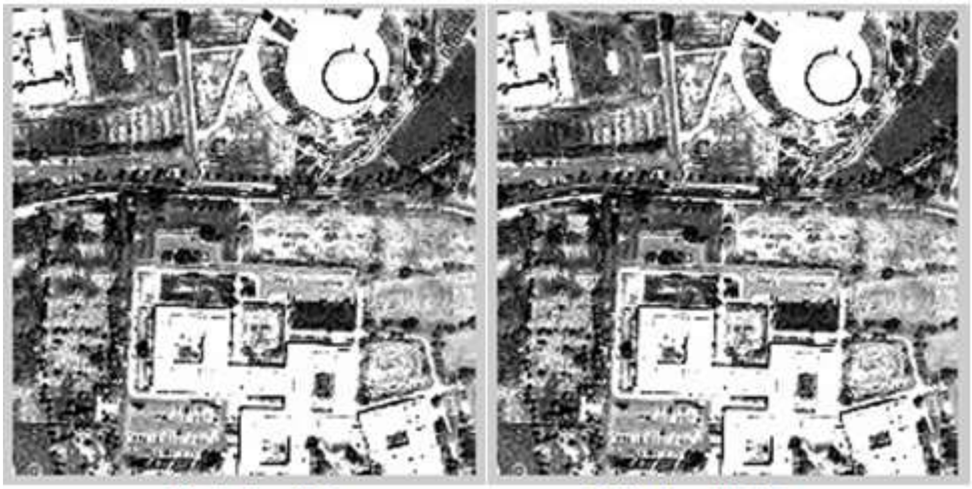


Figure 3: Compression Results of Baghdad Image (2m-wv-2) by Using a newTechnique



Block size= 8x8 Block size= 16x16

Figure 4: Compression Results of Baghdad-1m-ikonas Image by Using a newTechnique

Table 1: New Technique Results

Image	Block Size = 4*4			Block Size = 8*8			Block Size = 16*16		
	CR	MSE	PSNR	CR	MSE	PSNR	CR	MSE	PSNR
Baghdad- 2m-wv-2	9.479	34.652	17.435	18.958	17.326	29.476	37.917	8.663	41.518
Baghdad-1m-ikonas	27.027	207.834	9.655	54.055	103.917	21.697	108.111	51.958	33.738

8. Conclusion

The objective of this thesis was to present a wide coverage of digital image processing techniques for analysing information in digital images. The digital processing techniques which were cover in this thesis ranged from extracting edge information to image data compression including edge encoding by one compression technique and rest image region encoding by another compression technique.

The approach followed in this work was to regard the image as being composed of regions, which have, edges, the edge points of each region forming a closed contour in two dimensions. The coding strategy was to determine the correct contour points, isolating the region enclosed by each contour.

We developed a new hybrid coding technique which combined two coding techniques and was applied to test images, these coding techniques were the absolute moment Block Truncation method used to coded the rest regions in the image, while, used the hadmard transform to coded the edge regions. Different results have been obtained when the threshold values, used in determine the type of region is changed. This method yields higher compression ratios,. The obtained result showed that; implementing our coding techniques yields higher compression ratios and, at the same time, better quantitative and qualitative test measures on colored images. This is because the distortion obtained with each of these techniques is, normally, distributed randomly through color bands (RGB).The results of the decoded images, showedin table (1), a good image qualities for large block size, with better value of compression ratio. It's clear from the obtained result that the values of (CR) and (PSNR) depend on the block size and preoperational with it .the result showed that the larger block size the higher (CR) and (PSNR).so block size (4×4) gives small values of (CR) and (PSNR). While using block size (16×16) gives high values of (CR) and (PSNR).in other words: the values of (CR) and (PSNR) rise up when we use larger block sizes.

References

[1] Mather, Paul M., and Magaly Koch. Computer processing of remotely-sensed images: an introduction. John Wiley & Sons, 2011.

[2] Jassim, Firas A., and Hind E. Qassim. "Five modulus method for image compression." *arXiv preprint arXiv:1211.4591* (2012).

[3] M. I. Rajab, M. S. Woolfson, and S. P. Morgan,“ Application of region-based segmentation and neural network edge detection to skin lesions,” Computerized Medical Imaging and Graphics,vol.28,no.1-2, pp.61–68,January 2004.

[4] Wang, Dongyun, and Huiqin Liu. "Edge detection of cord fabric defects image based on an improved morphological erosion detection methods." In *Natural Computation (ICNC), 2010 Sixth International Conference on*, vol. 8, pp. 3943-3947.IEEE, 2010.

[5] R. Ramachandran, R. Manivannan, R. Ramachandiran, N. Balachandar” A Real Time Automatic License Plate Recognition Using Optical Character Recognition” *International Journal of Engineering And Computer Science I Volume 4 Issue 1*, pp: 9789-9796, January 2015.

[6] Zhang, Xuemei, and Brian A. Wandell. "A spatial extension of CIELAB for digital color-image reproduction."*Journal of the Society for Information Display* 5, no. 1,pp: 61-63, 1997.

[7] Sonal, Dinesh Kumar. "A study of various image compression techniques."COIT, RIMT-IET.Hisar (2007).

[8] Shashikumar.S, ArpanaParakale, BharamgondaMadhuriMahavir, and BharamuUllagaddi. “Image Compression using Absolute Moment Block

Truncation Coding." *International journal of advanced scientific and technical research*,vol.3 (3), 2013, pp: 97-102.

[9] A. K. Jain, P. M. Farrele, and V. R. Al Gazi, "Image Data Compression" In Digital Image Processing Technique, New York Academic Press, 1984.

[10] PRAMEELAMMA, M., V. SRIDHAR, G. NAGENDRA, V. VARAPRASAD, and P. ANILKUMA"AMBTC-CompressedImagesforLosslessSteganography." *International Journal of Advanced Research in Computer Science and Electronics Engineering (IJARCSEE)* 1, no. 6 (2012): pp-156.

[11] Arbat, Poonam A., and P. D. Gawande. "Improved Adaptive Block Truncation Coding For Image Compression in LCD Overdrive." *International Journal of Recent Technology and Engineering* 2, no. 4 (2013).

[12] Yang, Chen-Kuei, Ja-Chen Lin, and Wen-Hsiang Tsai. "Color image compression by moment-preserving and block truncation coding techniques." Communications, IEEE Transactions on 45, no. 12 (1997): 1513-1516.

[13] Eskicioglu, Ahmet M., and Paul S. Fisher. "Image quality measures and their performance." Communications, IEEE Transactions on 43, no. 12 (1995): 2959-2965.

[14] Fränti, Pasi, Olli Nevalainen, and TimoKaukoranta. "Compression of digital images by block truncation coding: a survey." *The Computer Journal* 37, no. 4 (1994): 308-332.