New Lossless Image Compression Technique using Adaptive Block Size

I. El-Feghi , Z. Zubia and W. Elwalda

Abstract: - In this paper, we focus on lossless image compression technique that uses variable block size and simple arithmetic operations to achieve high compression ratios in spatial domain. The main idea of the proposal method is based on partitioning the image into several sub images that have interpixel relation. The second stage is to encode the sub images in a novel fashion which the distance between the first pixel of each and every other pixels in the block. Depending on the homogeneity of the block, the distance between the between related pixels will require only four bits at most. In the third stage, we create a histogram of all distances obtained to represent each sub image. These steps are repeated and the decision to continuous of compression is decided by the maximum histogram bin. The output string which represent will be recompressed by another similar method using the distance again. The method was tested on more than 100 color and grey images. Results obtained reached high ratios of 75% reduction in size especially in medical image due to grey level homogeneity.

Keywords— Image compression, spatial domain, lossless compression**,** homogeneity

.

I. INTRODUCTION

Beginning with modest initial attempts in roughly the 1960s, digital image processing has become a recognized field of science, as well as a broadly accepted methodology, to solve practical problems in many different kinds of human activities [1].

With the advance of the information age the need for mass information storage and retrieval has grown drastically. The capacity of commercial storage devices, however, has not kept pace with the proliferation of image data. Images are stored on computers as collections of bits representing pixels, or points forming the picture elements. Since the human eye can process large amounts of information, many pixels - some 8 million bits' worth - are required to store even moderate quality images. Although the storage cost per bit is not high in cost, the need to compress image to save storage and bandwidth is still needed. Image compression can play an important role in this one area. Storing the images in less memory leads to a direct reduction in cost. Another useful feature of image compression is the rapid transmission of data; less data requires less time to send. The main idea behind image compression is the removal of redundancy. Most data contains some amount of redundancy, which can sometimes be removed for storage and replaced for recovery, but this redundancy does not lead to high compression. Fortunately, the human eye is not sensitive a wide variety of information loss. That is, the image can be changed in many ways that are either not detectable by the human eye or do not contribute to "degradation" of the image. If these changes are made so that the data becomes highly redundant, then the data can be compressed when the redundancy can be detected[2][3].

Compression schemes can be divided into two major classes: lossless and lossy compression schemes. Data compressed using lossless compression schemes can be recovered exactly, while lossy compression introduces some loss of information in the reconstruction [4].

There are several methods lossy image compression such as wavelet coding [5][6], neural networks [7], vector quantization [8] and fractal coding [9].

Lossless image compression techniques, as their name implies, involve no loss of information. If an image was compressed in a lossless fashion, the original image can be recovered exactly from the compressed data. Lossless compression is generally used for applications that cannot tolerate any difference between the original and reconstructed data [10][11]. If data of any kind are to be processed or "enhanced" later to yield more information, it is important that the integrity be preserved. Because the price for this kind of error may be a human life, it makes sense to be very careful about using a compression scheme that can reconstruct the image a compressed version of the original[11].

Although lossy image compression techniques can achieve higher compression ratio but loss of some valuble information can occur. When it is important for many application domains such as medical imaging when losses are not acceptable the choice is lossless techniques. Discarding of small image details that might be an indication of pathology could alter a diagnosis, with severe human and legal consequences. In several medical applications, like coronary angiography, where one has to measure sub millimeter blood vessel diameters at the location of the stenosis, lossless technique is the preferred method [12][13].

I.El-Feghi is a professor at University of Tripoli, Tripoli-Libya, (Phone:+218-91-178-7206; e-mail idrise@ee.edu.ly).

Z. Zubia is with University of Sirit, Sirit, Libya (e-mail zszubi@su.edu.ly).

W. Elwada is with the Academy of Graduate Studies, Misurata Branch, Misurata, Libya (e-mail: wajeeh_77@yahoo.com).

The main objective of this paper is to compress color and grey images by reducing number of bits per pixel required to represent it and to decrease the transmission time for transmission of images.

The rest of the paper is organized as follows: Section -1 explains the need for compression, section-2 different types of data redundancies are explained,section-3 Methods of compressions are explained, In section-4 the proposed method of compression is done, section-5 the results are presented with explanation

II. NEED FOR IMAGE COMPRESSION

There are several advantages of image compression that can be summarized as follows:

- o It provides a considerable bandwidth savings related to sending smaller amount of data over the internet. This applies to still images, audio and video.
- o Considerable saving in storages space.
- o Provides a higher level of security and immunity against illegitimate monitoring.

To store a grey image of size 1024x1024, about 1Mb of disk space and this multiplied by 3 for color images. The same image will require about 14 minutes for transmission, utilizing a high speed, 32 Kbits/s.

To store these images, and make them available after transmission, compression techniques are needed. The idea behind the size reduction process is the removal of redundant data. 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 2 minutes.

III. REDUNDANCY IN DIGITAL IMAGES

Most digital images share the common characteristic of containing some sort of information redundancy which is that the neighboring pixels are correlated to each other[4]. It is important to take advantages of these redundancies to reduce the size of image.

A) Types of Redundancy

There are three types of redundancies in digital image basically:

a. Coding redundancy
b. Inter pixel Redundancy
c. Psycho visual redundancy

Coding redundancy can be found on the image when less than optimal cods are used. This type of coding is always reversible and usually implemented using look-up tables (LUTs). Examples of image coding schemes that explore this type of redundancy are the Huffman coding.

Interpixel redundancy or spatial redundancy is due to correlations between the pixels in the image. Psychovisual redundancy is due to data that is visually ignored by the human visual. Image compression techniques reduce the number of bits required to represent an image by taking advantage of these redundancies.

VI. PROPOSED METHOD

In most images, especially medical image as image shown in figure. 1, there are some pixels which are related or similar, or we can say that they are at close distance to each other. Instead of store the pixel using 8 bits/pixel, we can store the distance between the pixels.

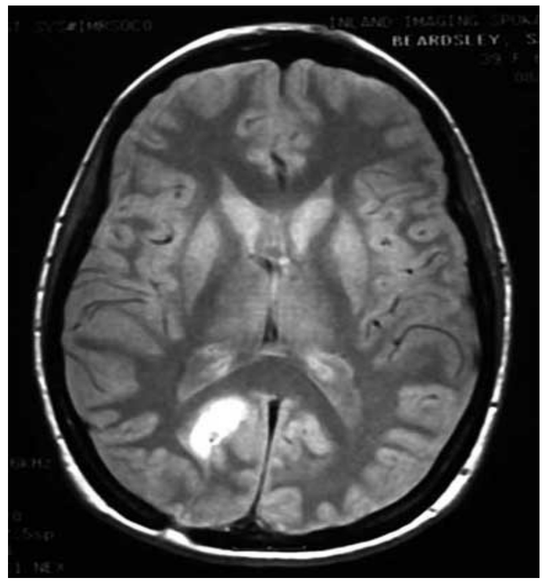

Figure 1. Sample medical image

To see the close distance between pixels in this image a block of size 5x5 is shown in figure.2

TABLE 1. sample 5x5 block

183	185	181	186	186
181	185	185	186	186
181	184	184	186	186
183	184	184	186	186
183	185	185	186	186

In the figure2, we can see related pixels, this 25 pixels needed 200 bit to store, (8 bit for each pixel) , but by using the distance we can reduce the storage size considerably by using on 4 bit pixel.

First step is portioning the image into small image called blocks, each block has related pixels, the maximum distance between pixels in the same block is 15 (4bit), then encode the block as shown in fig.3. The first sub-block will represent the block size. It can be up to 256 pixels which can be represented by 8 bits. The second sub-block contains the first pixel in the block which will also require 8bits.Next we represent the distance between pixel second and first pixels using one sign bit (0 for positive and 1 for negative) and 3 bits for the distance. This is applied for all pixels in the block. Fig. 4.show an example of how to encode 5 pixels block.

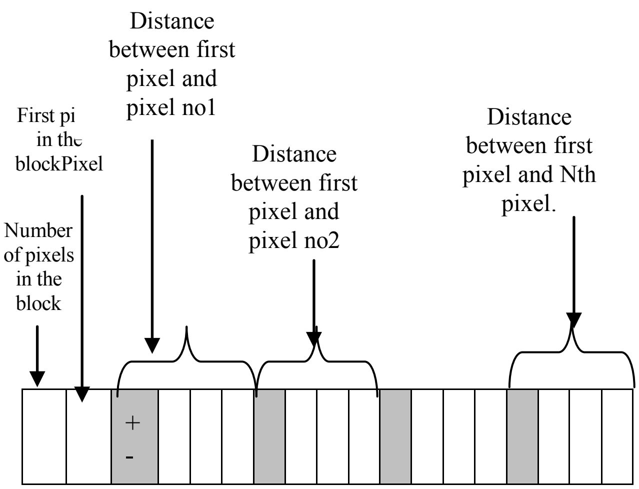

FIGURE 3 represent the pixels with proposal

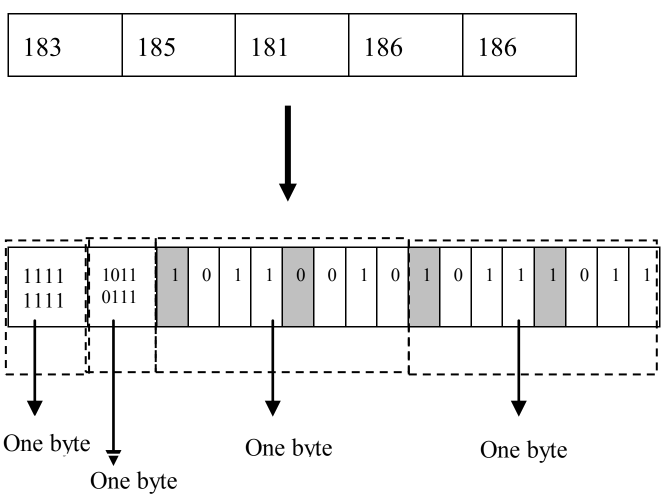

Figure 4. Example of encoded block of size 5x5

Now, suppose the first block in the image has 256 pixels that mean need 2048 bytes to store. While by using this encode we need only 129 bytes.

8 bits for number of pixels in the block + 8 bits for first pixel + 4*254= 8 + 8 + 1020=130 byte.

After we complete the first stage of compression, we do another compression stage. In this stage we will recompress the output string again. For example the produced string from last stage will look like this:

00011001 10110111 10101010 10111011 0100 1010 10101011 10110100 10011001 10111011 00001001 10011011 10110000 10101010 1011 1101

When converted to decimal it will look like:
25 183 170 187 74 171 180 ……………….. 189
The next step is to use the histogram bins shown in table 1 as a stopping criterion. If 30% of the data are gathered in any three bins then the compression stops. This indicates that the compression has reached its maximum rate.

TABLE 1. Assignment range used for stopping the compression algorithm.

Number	Range
1	0-15
2	16-31
3	32-47
4	48-63
5	64-79
6	80-95
7	96-111
8	112-127
9	128-143
10	144-159
11	160-175
12	176-191
13	192-207
14	208-239
15	249-255

If the stopping criterion is not fulfilled then we continue the compression in a different fashion as shown in fig. 5. In this stage we add one more byte to hold the number of compression runs. The next byte will hold the bin number of the area of the byte. One bit is used to indicate if the next number falls in the same area or in a different area (0 for same area and 1 for different area). The next bit is used to indicate the distance between the two numbers with 0 for positive and 1 for negative. As can be seen from figure 5 that this step will produce a considerable saving in size of the block.

This process is repeated until the stopping criterion is met.

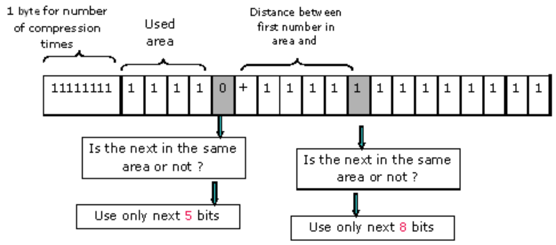

Figure.5 The second stage of compression

V. EXPERIMENTAL RESULTS

The proposed algorithm was tested on a large set of image of different models(colour, grey, binary) from different source (digital camera, x-ray, MRI). Sample of images used are shown in figures 6,7,8 and 9.

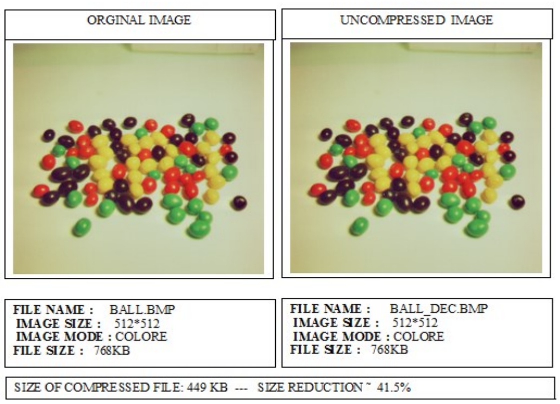

Figure 6. Ball color image of size 512x512

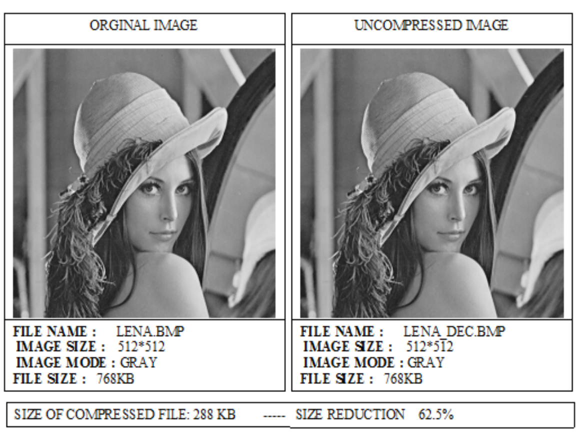

Figure 7. Grey image of size 512x512 pixels

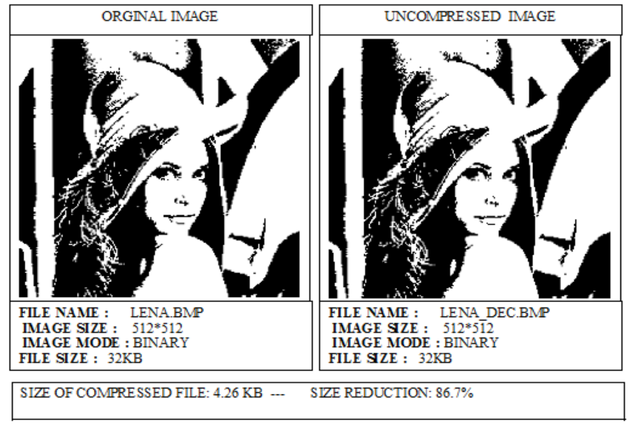

Figure 8. Binary image of size 512x512 pixels

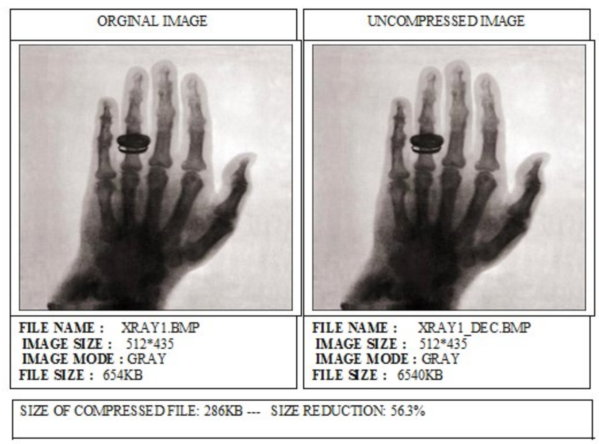

Figure 9. Medical image of size 512x512

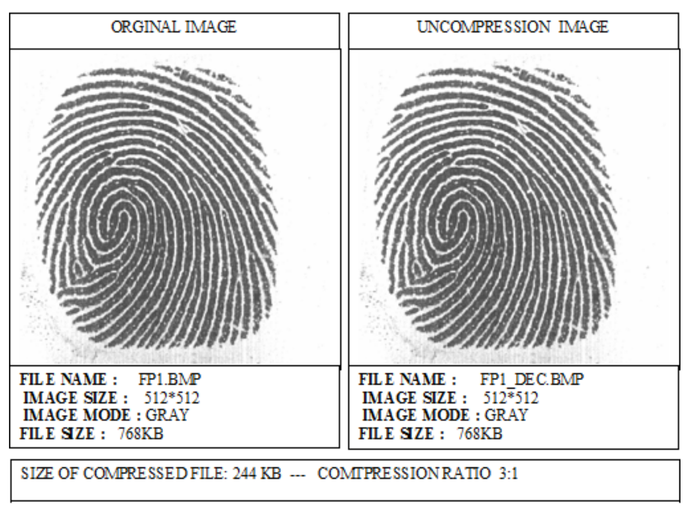

Figure 10. Fingerprints image compression details

TABLE 2. Average system results for color images

Image size	Compressing time	Uncompressing time	Compression ratio
1024	0.875s	0.775s	≈ 2:1
512	0.463s	0.585s	≈ 2:1
256	0.405s	0.425	≈ 2:1

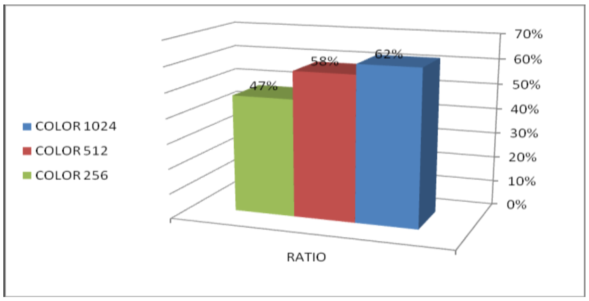

FIGURE 11 Average of compression ratio for color images

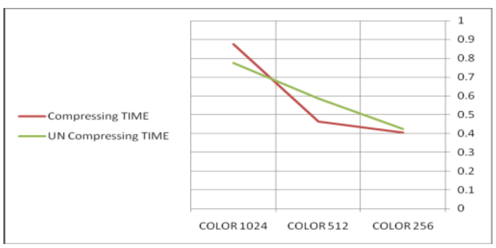

Figure 12. Average of compressing and Uncompressing Time for color images

TABLE 3. Average system results for gray images

Image size	Compressing time	Uncompressing time	Compression ratio
1024	1.241s	0.346s	≈ 4:1
512	0.352s	0.141s	≈ 3:1
256	0.089s	0.057s	≈ 3:1

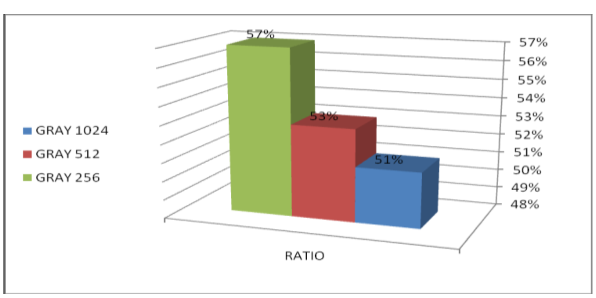

Figure 13. Average compression of grey images

TABLE 4. Average system results for binary images

Image size	Compressing time	Uncompressing time	Compression ratio
1024	1.560s	0.172s	≈ 38:1
512	0.038s	0.05s	≈ 7:1
256	0.169s	0.14	≈ 3:1

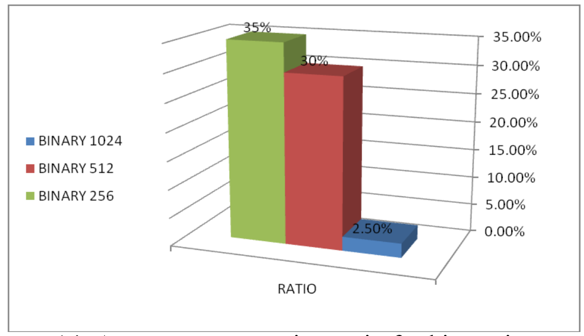

Figure 14. Average compression ratio for binary images

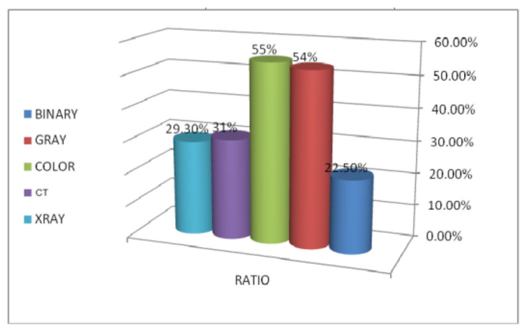

Figure 15. Average image size after compression

Figure 15 summarizes the results obtained for different types of image. It can be seen from fig.15 that best results are obtained in binary images due to the short distance between bits(0 or 1). The second highest results are obtained in medical images which is due to smooth distances in wide areas of the image. Over all most images colored and crowded grey images are reduced to about half of its original size without any loses in the reconstructed images. It is also worth mentioning that the that the higher data redundancy helps to achieve more compression.

VI. CONCLUSIONS

This paper presented a new compression algorithm based of interpixel relation. To enhance the compression further, we have utilized a different size histogram bins to gather similar pixels in close groups in closer histogram bins. The proposed algorithm was tested in a large set of images of different models. It was show that the proposed algorithm can achieve up to 78% reduction in size. The compression rates depend on the redundancy present in the image. Using this algorithm the result the decompressed image is same as that of the input image. This indicates that there is no loss of information during compression and image can be reconstructed exactly without any losses. The proposed algorithm is robust in time requirement since it uses only basic mathematical operations.

REFERENCES

[1] R. C .Gonzalez, and R. E. Woods., "Digital Image Processing", Prentice-Hall, New Jersey, 2002.

[2]. Dallwitz, M. J, " An introduction to computer images", TDWG Newsletter, vol. 7, pp. 1–3, 1992.

[3] A. K. Jain, "Fundamentals of Digital Image Processing", Prentice-Hall, Inc., USA, 1989.

[4] Said A and Pearlman WA," An Image Multiresolution Representation for Lossless and Lossy Compression*," IEEE Trans on Image Processing*, vol.5, pp. 1303-1310, 1996.

[5] Shaou-Gangmlaou, Shih-Tse chen, Shu-Nien chao, "Wavelet-based Lossy-to-Lossless Medical Image Compression using Dynamic VQ And SPIHT Coding" *, Biomedical Engineering Application ,Basis & Communication*, Vol. 15, no. 6, pp. 235-242, December 2003

[6] J. M. Shapiro, "Embedded image coding using zero trees of wavelet coefficients," *IEEE Trans. Signal Processing*, vol. 41, pp:3445-3462, 199

[7] R. D. Dony and S. Haykin, "Neural networks approaches to image compression," *Proc. IEEE*, vol. 83, pp: 288-303,1995

[8] P. C. Cosman, R. M. Gray, and M. Vetterli, "Vector quantization of image subbands: a survey," *IEEE Trans. Image Processing*, vol. 5, pp: 202-225, 1996

[9] Jean Cardinal." Fast fractal compression of greyscale images*," IEEE Transactions on Image Processing*, vol.10, no.1, 2001.

[9] Y. Fisher, *Fractal Compression: Theory and Application to Digital Images*, New York, Springer-Verlag, 1994.

[10] D. Clunie, "Lossless Compression of Grayscale Medical Images - effectiveness of Traditional and State of the Art Approaches,*" in Proc. SPIE (Medical Imaging),* vol. 3980, Feb. 2000.

[11] N. D. Memon and K. Sayood, "Lossless image compression: A comparative study," in *Proc. SPIE (Still-Image Compression)*, vol. 2418, pp. 8-20, Feb. 1995.

[12] C. Christopoulos, A. Skodras and T. Rbrahimi, " The JPEG 2000 Still Image Coding System: *An Overview", IEEE Trans.of Consumer Electronics*, vol.46, pp.1103–1127, 2000.

[13] R. P. Lewis, "President's page: developing standards for the digital age: the Dicom project," J. Amer. Coll. Cardiol., vol. 28, PP: 1631-1632, 1996.

Idirs El-Feghi Born in Misurata Libya 1959 Obtained BSc form University of Portland (1983) in Computer engineering, MSc. and PhD. from University of Windsor, Canada, (1999) and(2003) in Electrical and Computer engineering. Currently he is a professor at University of Tripoli, Tripoli-Libya. His area of interest include Image Processing, Artificial Intelligence, Optimization and Data Mining.

Zakaria Zubia is a professor at University of Sirit,Sirit, Libya at the Faculty of Information Technology. His area of interest includes Expert systems, distributed Computing and Data mining