IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 2, February 2017

Grayscale Image Compression using Bit Plane Slicing and Developed RLE Algorithms

Hassan K. Albahadily[1, 2], V. Yu. Tsviatkou[3], V.K. Kanapelka[3]

PhD, Dept. of Telecommunication and Network Devices, BSUIR, Minsk, Belarus[1]

Lecturer, Dept. of Computer Science, University of Mustansiriah, Baghdad, Iraq[2]

Professor, Dept. of Telecommunication and Network Devices, BSUIR, Minsk, Belarus[3]

Abstract: New suggested RLE compression algorithms to compress grayscale images with bitplane slicing technique to reduce the size of the encoded data by separating image into 8 binary layers, then use our modified RLE algorithms to compress bitplanes. Our modified algorithms designed perfectly to compress bitplane. The proposed methods achieved very good compression ratio especially with the MSB layer.

Keywords: image compression; bitplane; quantization.

I. INTRODUCTION

Data files frequently contain the same character repeated many times in a row or column. The digitized signals can also have runs of the same value, indicating that the signal is not changing, also images and music [1]. The Image is a two dimensional array of pixel intensities or can be considered as a discrete representation of data possessing both spatial (layout) and intensity (colour) information [2].
There is significant redundancy present in image signals; this redundancy is proportional to the amount of correlation among the image data samples [3].
Image compression addresses the problem of reducing the amount of data required to represent the image while keeping the resolution and the visual quality of the reconstructed image as close to the original image as possible, or it is a process intended to yield a compact representation of an image, thereby reducing the image storage/transmission requirements.
Image compression techniques reduce the number of bits required to represent an image by taking advantage of three basic data redundancies: the Coding redundancy which is present when less than optimal code words are used. The Interpixel redundancy results from correlations between the pixels of an image. The Psychovisual redundancy is due to data that is ignored by the human visual system (i.e. visually non-essential information) [4].
The goal of image compression is to represent an image signal with the smallest possible number of bits, thereby speeding up transmission, minimizing storage requirements, reduces the cost of data transmission and reduces the errors of transmission.
In the grayscale images-where the bit depth is 8bits- the pixel intensity is a decimal number between0 and 255 representing the grey level of the pixel intensity where thedecimal number 0 represents level black and decimal number 255 represents level white. and because each pixel's value is a 8bit decimal number in the range (0-255) so we can separate first bit from each pixel in the image to generate first layer which consist ofbinary numbers 0 and 1, and the generated image from this layer will be binary image consists of black and white only and we will separate second bit to generate second layer and so on.
Bit resolution defines the number of possible intensity/colour values that a pixelmay have and relates to the quantization of the image information. For instance a binaryimage has just two colours (black or white), a greyscale image commonly has 256different grey levels ranging from black to white whilst for a colour image it depends on the colour range in use (most commonly is 24 bit) [2].
The aim of this paper is to compress grayscale images using the bit plane technique and modified RLE algorithms by reducing the size of bits to represent the pixels' intensity of image, which allows storing the image with less number of bits.
Our technique is implemented using MATLAB2012 on WINDOWS7 Operating System.

II. RELATED WORK

Because the importance of image compression, several efforts involved in this field and implemented their new methods or enhanced algorithms. Some of recent related efforts can be mentioned here:
A method robust to bit error, and has the advantages of simple structure and easy software and hardware implementation[5], while a good method suggested for bit plane slicing and adaptive Huffman with LZW[6], or an efficient hybrid algorithm by sequence the bits in the bit plane with selected scan order and then encode the bits by the combination of RLE and modified Huffman coding scheme[7], or by combines adaptive quantization with an additional unit that selects only those bit planes which are visually sensitive and contain coarse information and then improving results by introducing DWT and adaptive

thresholding[8], also there are some efforts focusing on the idea of reducing some bits from the images using different values of quantization factor[9], or using the idea of discarding some bitplanes because they don't contribute too much information in image formation and doesn't degrade the visual effect of the image [4]. Also there is a good effort of improved compression algorithm by reducing the correlation and spatial redundancy between pixels [11], and an effort has developed a compression algorithms based on two decomposition approaches: bitplane decomposition and binary-tree decomposition with segmentation and arithmetic coding [12].
All of these efforts are valuable steps to enhance the compression process and achieving good results.

III. BITPLANE SLICING

The bitplane slicing is a fundamental technique of image processing in which the image is sliced into different planes(each layer contains sequences of only binary digits 0 or 1). It is ranges from plane1 which contains the least significant bit (LSB) to the last planeN which contains the most significant bit (MSB),where the number of layers depends on the bit depth of the image. The bit depth means how many bits need to represent the pixel's intensity. For example if the image is grayscale then the bit depth is 8bit and it will be separated into 8 layers, or into 24 layers if the image is coloured i.e. bit depth is 24bit.
The first bit in the binary number is the least significant bit (LSB) which is not very efficient and its value is very small (equals one if this bit=1) and will not effects on the pixels value. While the last bit is the most significant bit (MSB) which is efficient and contains the important weight of the pixel's value (equals 2N-1 if this bit=1).

Here is an example about decimal number which can be represented as a binary number, for example the decimal number 204 can be represented in the following binary form with the decimal weight for each bit.

MSB	**1**	1	0	0	1	1	0	**0**	**LSB**
Decimal weight	128	64	32	16	8	4	2	1	

Fig.1.Binary representation of decimal number

It is clear that the intensity value of each pixel can be represented by 8-bit binary vector (b8,b7, b6, b5, b4, b3, b2, b1) bk, where k is from 1 to 8 and each bk is either "0" or"1". In this case, the grayscale image may be considered as an overlay of eight bit-planes and each bit-plane can be thought as a two tone image and can be represented by a binary matrix [13][14].The formation of bit plane $BitPlane_k$ is given by the equation below [4]:

$$BitPlane_K = Reminder\left\{\frac{1}{2} floor\left[\frac{1}{2^{K-1}} Image\right]\right\} \qquad (1)$$

Or simply we can extract the K^{th} bit from all pixels to generate the $BitPlane_K$.

The spliced biplanes for Lena shown in the next figure:

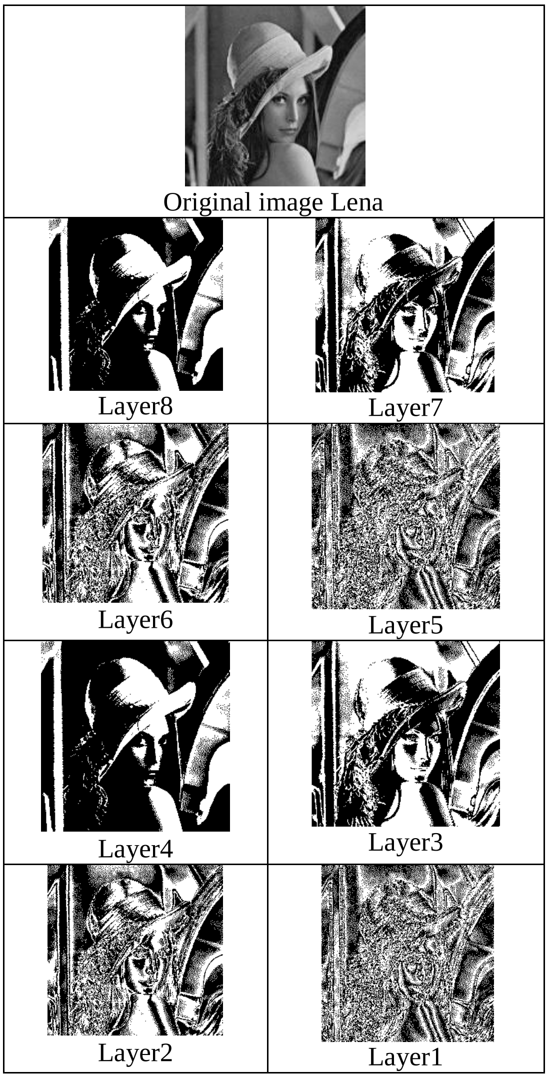

Fig. 2.bitplane slicing of image Lena

The bitplane decomposition is very useful for image compression. It allows some bi-level compression [12], i.e.it is used in some ways to compress images based on the idea of splitting the image into layers of binary values then either omit layers which are not highly effect on the image quality or by using the idea of similarity of elements in the bit plane which would be appears highly in the MSB layers, and by this way a long runs of similar values would result in very good compression rates [15][16]. Thus The RLE may be advantageously applied because the long runs in the bit planes which is the backbone of RLE [17]. This technique is very useful even if there are no repeated runs in the pixels, and by using bitplane slicing we will find some kind of repetition especially with last layer which contain MSB and achieving the highest compression ratio because it is contain frequently repeated runs.

The digital image is expressed in form of pixels which can be expressed further in terms of bitsas shown below for the image portion of Lena's eye

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319 5940
International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 2, February 2017

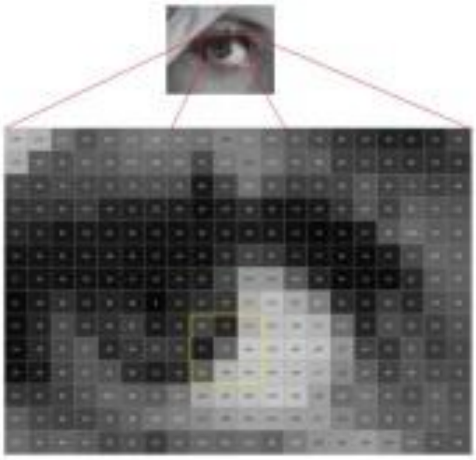

Fig. 3.Pixel representation for Lena

The selected area contains the following pixels values (we can see that there is no repeated value of neighboured pixels)

68	39	174
31	97	198
94	176	193

We can represent the selected area's pixels by binary numbers as follow:

01000100	00100111	10101110
00011111	01100001	11000110
01011110	10110000	11000001

This 8-bit image is composed of eight one-bit planes. Plane 1 contains the lowest order bit of all pixels in the image, and plane 8 contains the highest order bit of all pixels in the image. The generated bitplane can be shown in the following figure:

0	0	1
0	0	1
0	1	1

Layer8

1	0	0
0	1	1
1	0	1

Layer7

0	1	1
0	1	0
0	1	0

Layer6

0	0	0
1	0	0
1	1	0

Layer5

0	0	1
1	0	0
1	0	0

Layer4

1	1	1
1	0	1
1	0	0

Layer3

0	1	1
1	0	1
1	0	0

Layer2

0	1	0
1	1	0
0	0	1

Layer1

Now we can see that there are many repeated values in the new form of the image.

IV. THE MODIFIED RLE ALGORITHMS

The Run Length Encoding (RLE) is a simple method counts the values or runs and their repeated time as pairs of run and repeat (I,N), where I representing the run and N representing the repeat [3].

Fig. 4.Illustration of RLE for a binary input sequence

A. **The Algorithm I3BN**

The modified RLE algorithm I3BN using the same idea of classical RLE which countsrepeated runs but instead of sending the values and runs as pairs,I3BN will send the run (I) then 3 bits (3B)-- then how many times repeated the value minus three (N) to generate the form I3BN.
After trying some numbers, the 3 bits (3B) has been chosento minimize number of bits that used to represent the run repeat by sending one bit or two or three followed by the number of repeated runs [10]. Algorithm I3BN using three symbol b1, b2 and b3, which take 1 if the value I repeated and 0 if absent.
The structure of coded data for algorithm I3BN can be represented by the following block diagram:

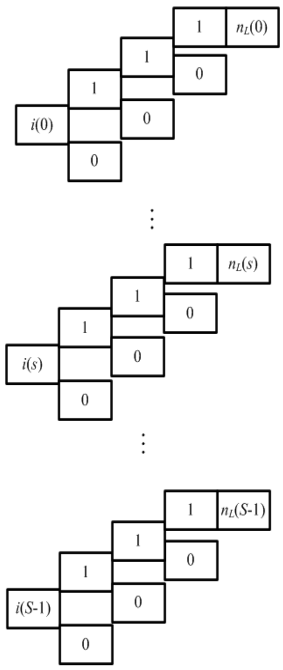

Fig. 5. Data structure of I3BN

So if the run repeated one time we will send the binary series 10, if the run repeated two times we will send 110, if the run repeated three times we will send 1110 and if the run repeated more than three times we will send the sequence 111 followed by reminder from subtraction three from the value of repeat and send it in binary form.

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319 5940
International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 2, February 2017

For example if the run repeated 8 times, we will send 111 followed by 5 and the code will be 111(101).

The code size R_{I3BN} (in bits) of algorithm I3BN defined by the following expression:

$$R_{I3BN} = S(BD_I + 1) + \sum_{s=0}^{S-1} b1(s) + \sum_{s=0}^{S-1} b2(s) + BD_N \sum_{s=0}^{S-1} b3(s) \qquad (2)$$

Where BD_I is the bit depth of the image and BD_N is the bit depth of the maximum repeat.

The modified algorithm I3BN will work perfectly with bitplane by increasing the number of runs and decreasing the number of bits to represents runs. For example if we have the following 24 bits as input data
000000001111001111111111
The repeat for this code is 8,4,2,10 and its I3BN code will be:
111101 111001110 111111
The drawback of this algorithm is the fixed number of bits reserved to represent the repeat; it will be fixed according to the maximum repeat even if the repeat needs one bit.

B. **The Algorithm I3BSN**

The algorithm I3BSN is similar to algorithm I3BN but with variant length of bits to represent the repeat exactly as needed and will not wasting memory space. So we will need to find how many bits needed to represent each repeat.

The structure of coded data for algorithm I3BSN can be represented by the following block diagram:

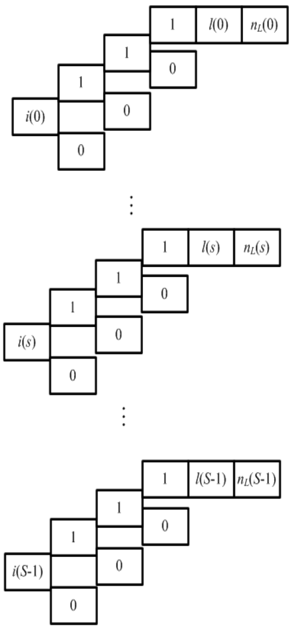

Fig. 6. Data structure of I3BSN

For example if we have the following 24bits as input data:
111111111111111111110000
The first repeat is 20 so we need 5 bits to represent it, then the second is 4 which needs only one bit to represent it and so on.

The repeat run for code above is 20,4.
The I3BSN code for this repeat will be:
111(101)10001 111(001)1
The coded data will be less than from the previous algorithm I3BN and saving more memory space.

V. BITPLANE EXPERIMENT IMPLEMENTATION

Four grayscale images with size 512×512 pixel include different images (forest, city, field, sea)have been used to test modified algorithms. The test images shown in Fig.6 below:

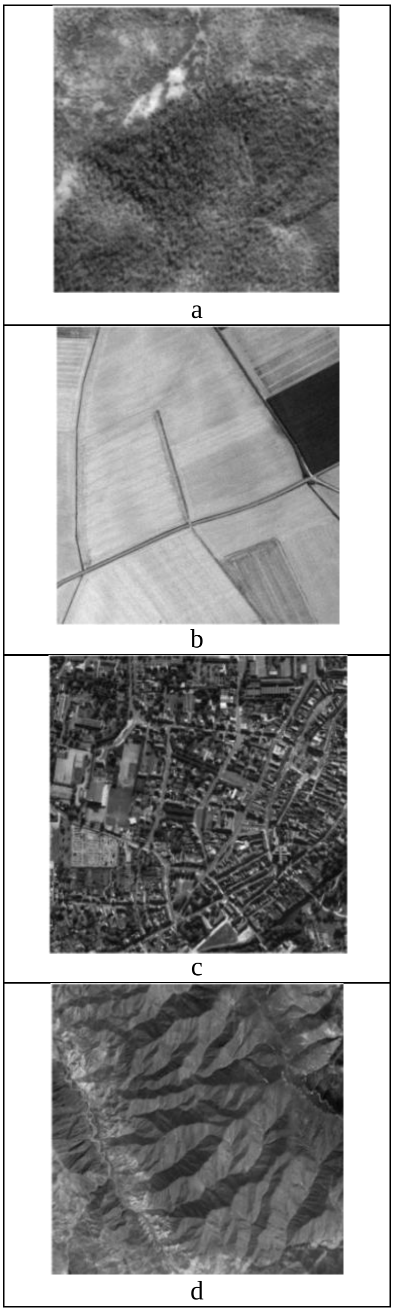

Fig. 6. Test images used in the experiment a-forest, b-city, c-field, d-sea

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319 5940
International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 2, February 2017

First step is splitting the test images into 8 layers or bitplanes (each layer will be binary image contains 0,1 only) as in the following figure:

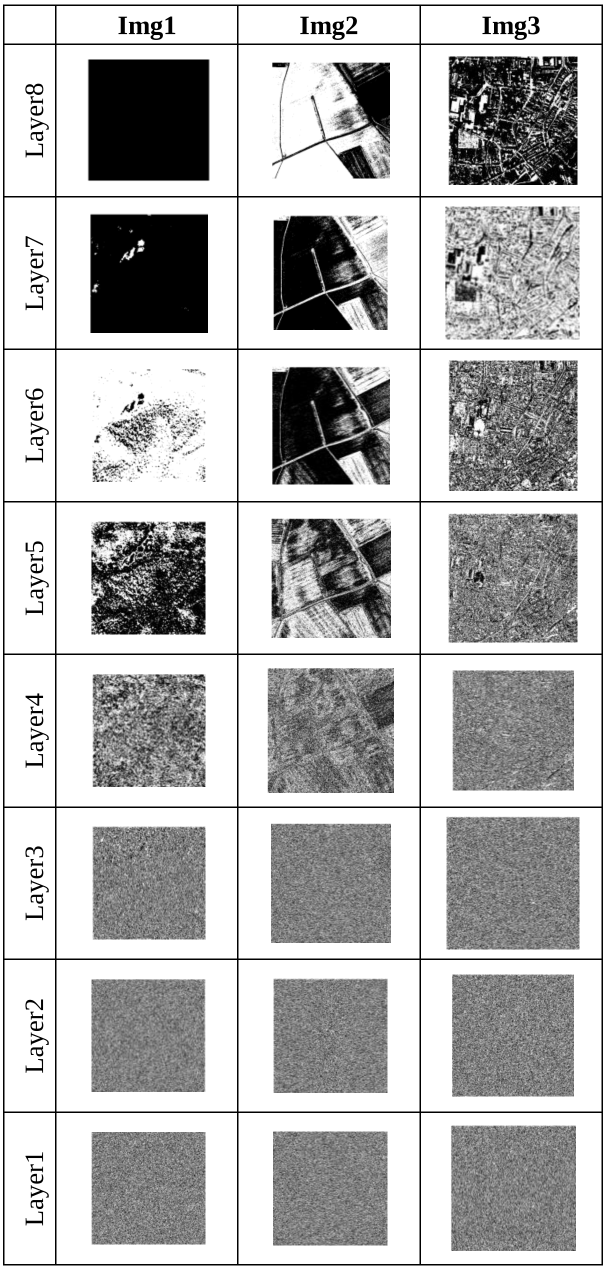

Fig. 7.Bitplane slicing for test images (Img1, Img2, Img3)

It is possible to remove some information from an image without any apparent change in its visual appearance because the first three bits (Layer1, Layer2, and Layer3) does not contribute so much information in image formation (their value together will be 7 if all of them are set to 1). The image can be stored with the information provided by bit4 to bit8 only. Thus number ofbits per pixel can be reduced to 5 which save more storage space [2][4]. This idea will be another part of our experiment by reducing the three LSB from the images to generate new reduced image as shown in following figure:

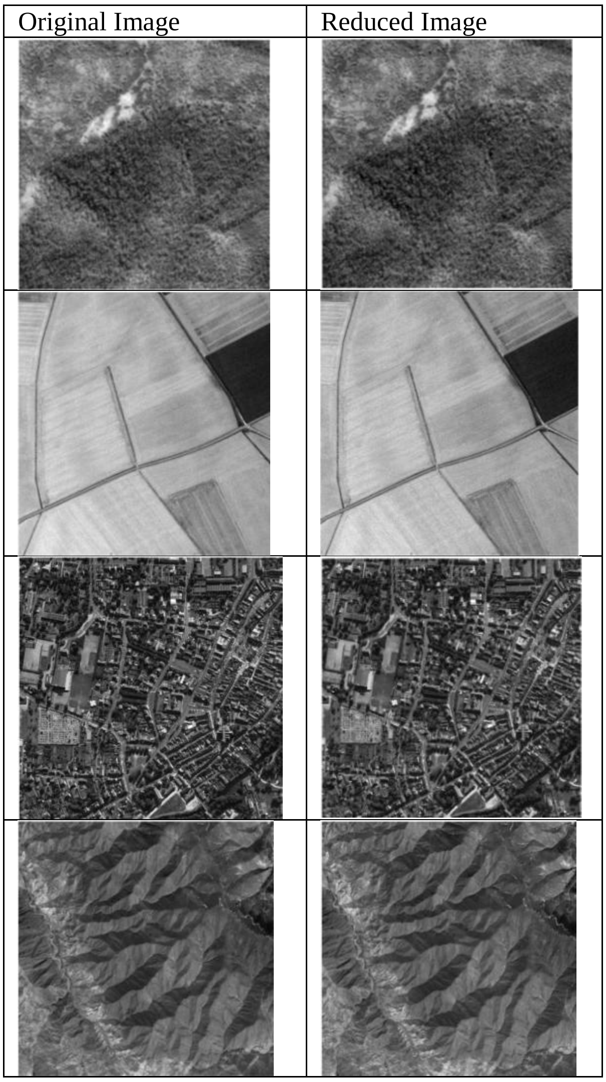

Fig. 7. The test images with their 5bit reduced images

We can see that there is no big change in the visual appearance of the images because the 3 LSB does not contribute big value especially with dark images and only the light images will contains minor effect.

VI.RESULTS AND DISSCUTION

The first part of our experiment is splitting the images into 8 bitplanes and implementing the modified RLE algorithms I3BNand I3BSN on each plane.
The results were very good as we can see in the table 2,3 below:

Table 2:compression ratio of bitplane for I3BN

Image	Img1	Img2	Img3	Img4
Layer8	11915.636	1.801	1.095	1.346
Layer7	22.524	1.548	0.809	1.269
Layer6	1.600	1.328	0.64	0.875
Layer5	1.051	0.738	0.622	0.658
Layer4	0.729	0.633	0.597	0.642

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 6, Issue 2, February 2017

Layer3	0.635	0.615	0.64	0.619
Layer2	0.639	0.64	0.641	0.640
Layer1	0.615	0.64	0.615	0.640
Total bits	2032826	2525868	3076507	2748183

TABLE 3:COMPRESSION RATIO OF BITPLANE FOR I3BSN

Image	Img1	Img2	Img3	Img4
Layer8	9709.037	1.959	1.163	1.413
Layer7	30.389	1.699	0.892	1.338
Layer6	2.307	1.387	0.716	0.960
Layer5	1.130	0.806	0.657	0.734
Layer4	0.771	0.639	0.630	0.648
Layer3	0.642	0.624	0.624	0.627
Layer2	0.623	0.624	0.625	0.624
Layer1	0.624	0.624	0.623	0.624
Total bits	1941583	2471481	2958930	2672057

From results above we can see that with bitplane slicing, the proposed algorithms I3BN, I3BSN provides very good compression ratio for layer 8 (up to 11915.636 , 979.037 time) of the dark images because the MSB always will be zero and (up to 1.346, 1.413 time) for light images. While the compression ratio for LSB was not big ratio and it is almost similar for all images and could be removed to achievea good ratio.
The second part is implementing the modified algorithms on the images after reducing the three LSB and we got the results below:

TABLE 5: Compression Ratioafter reducing LSB

Image	Img1	Img2	Img3	Img4
I3BN	2.328	1.354	1.047	1.305
I3BSN	2.457	1.357	1.058	1.330

VII. CONCLUSION AND FUTURE WORK

The Results showing that the modified algorithm provides compression ratio up to 11915.636-1.346,9709.037-1.413 times for MSB according to the nature of the image and its bitplanes.
The modified algorithms provide compression ratio up to2.457-1.058 times when removing the first three LSB bits. For Future work it will be good to use the proposed algorithms to compress the colored images with different color space like RGB and HSV.

REFERENCES

[1] S. Smith, The Scientist and Engineer's Guide to Digital Signal Processing, 2nd Edition, California Technical Publishing 1999.
[2] C. Solomon, T. Breckon, Fundamentals of Digital Image Processing, A Practical Approach with Examples in Matlab, John Wiley & Sons, 2011.
[3] A. Bovik, The Essential Guide to image Processing, Elsevier Inc. USA 2009.
[4] S. Halder, D. Bhattacharjee, M. Nasipuri, D. Kumar Basu,ALow Space Bit-Plane Slicing Based Image Storage Method using Extended JPEG Format, International Journal of Emerging Technology and Advanced Engineering,Vol 2, Issue 4, 2012.
[5] G.Gao , K. Xiong ,L.Shang, Bit-plane Image Coding Scheme Based On Compressed Sensing,Applied Mathematics and Information Sciences, An International JournalAppl. Math. Inf. Sci. 6, No. 3, 2012.
[6] V. Biala, D. Kaur, Lossless Compression Technique Using ILZW with DFRLC, International Journal of Computer Science Issues, Vol. 9, Issue 3, No 3, 2012.
[7] P. Pandian,N. Sivanandam, Hybrid Algorithm for Lossless Image Compressionusing Simple Selective Scan order with Bit Plane Slicing, Journal of Computer Science 8, Science Publications 2012.
[8] M. Gupta,R. Anand, Color Image Compression Using Set of Selected Bit Planes, IJECT Vol. 2, Issue 3, 2011.
[9] S. Santraa ,P. Acharjya, A Novel Approach of Digital Image Compression Technique using Bit Plane Size Reduction , International Journal of Current Engineering and Technology 2013.
[10] H. Albahadily, V.Yu. Tsviatkou, V.K. Kanapelka, New Modified RLE Algorithms to Compress Grayscale Images with Lossy and Lossless Compression, International Journal of Advanced Computer Science and Applications, Vol. 7, No. 7, 2016.
[11] A. Kumar ,P. Singh , Aggrandize Bit Plane Coding using Gray Code Method, International Journal of Computer Applications Volume 20, No.6, 2011.
[12] L. Matos, A. Neves , A. Pinho , Lossy-to-Lossless Compression of Biomedical Images Based on Image Decomposition, Applications of Digital Signal Processing through Practical Approach, InTech. 2015.
[13] R. Gonzalez, R. Woods, Digital Image Processing 2nd ed., Prentice Hall, Inc., New Jercy.
[14] K. Ting, D. Bong, Y. Wang, Performance Analysis of Single and Combined Bit-Planes Feature Extraction for Recognition in Face Expression Database, Proceeding of the International Conference on Computer and Communication Engineering, Kuala Lumpur, Malaysia,May 2008.
[15] R. Rangayyan ,BiomedicalImage Analysis, University of CalgaryCRC Press LLC, Canada , 2005.
[16] A. McAndrew, An Introduction to Digital ImageProcessing with MatlabNotes for SCM2511 ImageProcessing,Victoria University of Technology, 2004.
[17] A. Bovik, Hand Book of Image AndVideo Processing, Academic Press Austin USA, 2000.