RESEARCH ARTICLE OPEN ACCESS

Comparative Analysis of Lossless Image Compression Based On Row By Row Classifier and Various Encoding Schemes on Color Images

Ramandeep Kaur, Dr. Sukhjeet K. Ranade
M.Tech (CSE), Department of Computer Science, Punjabi University, Patiala.
Assistant Professor, Department of Computer Science, Punjabi University, Patiala.

ABSTRACT
Lossless image compression is needed in many fields like medical imaging, telemetry, geophysics, remote sensing and other applications, which require exact replica of original image and loss of information is not tolerable. In this paper, a near lossless image compression algorithm based on row by row classifier with encoding schemes like Lempel Ziv Welch (LZW), Huffman and Run Length Encoding (RLE) on color images is proposed. The algorithm divides the image into three parts R, G and B, apply row by row classification on each part and result of this classification is records in the mask image. After classification the image data is decomposed into two sequences each for R, G and B and mask image is hidden in them. These sequences are encoded using different encoding schemes like LZW, Huffman and RLE. An exhaustive comparative analysis is performed to evaluate these techniques, which reveals that the proposed algorithm have smaller bits per pixel (bpp) than simple LZW, Huffman and RLE encoding techniques.

I. INTRODUCTION

In today era of computer industry, the major challenges are to minimize the time of transmission data over network and reduce the storage space. Transmission time can be reduced significantly by reducing the size of data files. A data file can be text, graphic images, audio, video or algorithms. Reducing the size of data files also minimize the storage capacity or space requirement on the computer. In many fields like medical imaging, remote sensing, military affairs and scientific research, both low storage consumption and high image quality is required. Therefore lossless and near lossless compression methods are strongly needed. Bit-plane encoding, Run Length Encoding (RLE), Huffman, Lempel Ziv Welch (LZW), arithmetic encoding and lossless predictive encoding are all popular lossless compression methods [1].Data compression is the technique to reduce the size of data files by reducing redundancy, which helps to decrease data storage requirements and hence minimize time and communication cost [2]. It can also be defined as the method of encoding rules that allow substantial reduction in the total number of bits to store or transmit a file [3]. Image compression is the application of data compression on digital image. Image compression is the technique to reduce redundant and irrelevant image data in order to store or transmit data over network in very efficient form. Image compression can be of two types, lossless image compression and lossy image compression. If the process of redundancy removing is reversible i.e. the exact reconstruction of the original image can be achieved then it is known as lossless image compression [4]. Lossless image compression methods used in applications like medical imaging, scientific images, satellite imaging, artificial images remote sensing and forensic analysis. On the other side lossy compression methods are used in web browsing, photography, image editing and printing [4-6]. In lossy compression there is a problem of compression artifacts, because it works on low bit rate. Yang and Bourbakis presented a paper to discuss the comparison of various lossless compression schemes depending upon their performance. Lossless image compression can always be modeled as a two stage procedure first one is décor-relation and second is entropy coding [4]. A new compression scheme based on sub block interchange (SBI) was proposed by Ng and Cheng [7]. When this scheme is compared with Burrows and Wheeler Transformation (BWT) [8], SBI gains a little bit improvement in the compresses ratio but have a greater improvement in the compression speed. BWT takes much longer time for compression than GIF because input data requires a sorting procedure based on quick sort algorithm [8]. Triantafyllidis and Strintzis presented a technique for the implementation of context based adaptive arithmetic entropy coding. A drawback of arithmetic coding of images using the above adaptive model is that it does not take into account the high amount of correlation between adjacent pixels [9]. Yao-hua et al. presented a paper in which a near-lossless image compression algorithm using classification, information hiding and LZW is proposed. The algorithm takes advantage of the effects of the distribution of pixels on compression ratio. In the proposed algorithm, pixels of an image are classified row by row; pixels similar in value are gathered and the classification result is recorded in a mask image [1].In this paper lossless compression algorithm on color images having 24 bits per pixel (bpp) value is proposed using row by row classifier and different popular encoding schemes like LZW, Huffman and RLE. The proposed algorithm is a preprocessing stage followed by different encoding schemes which helps to reduce the size of image more than that of simple encoding schemes.

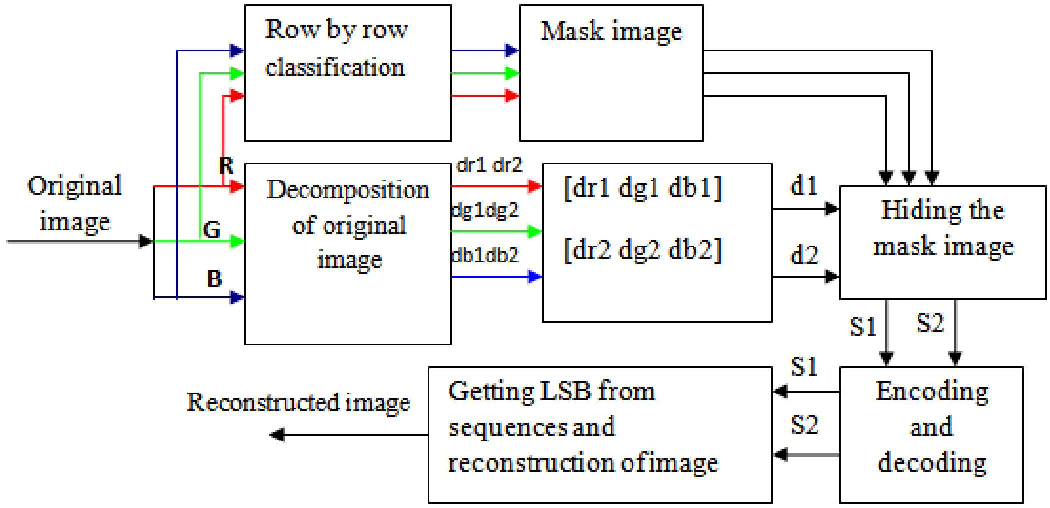

Fig. 1. The Flow of Proposed Algorithm on Color Images

II. THE PROPOSED ALGORITHM

The proposed algorithm is mainly data re-ordering process before applying the compression technique. To achieve better compression from a compression technique, input data may be processed in such a manner that correlation among the elements of the data can be increased. Here pixels of an image taken row by row are placed in two groups based on a threshold value. So that pixels having similar values fall in a same group. That is more correlation can achieve in the values of pixels. The proposed algorithm consists of 4 steps which are shown in Fig. 1. These steps are discussed in following subsections.

2.1 Classification row by row

In this step classification will be performed Row by row based on determining the threshold by using histogram of each row. Classification is done to put pixels in classes and pixels in same class must have similar value. Here classes will be divided according to the threshold calculated using histogram of each row. The number of classes will be six, two for each Red (Clr_1, Clr_2), Green (Clg_1, Clg_2) and Blue (Clb_1, Clb_2).After calculating threshold for each row, the pixels of each part (R, G and B) can be classified using below method:

For Red part $$\begin{cases} Ir(i,j) \in Clr_1, & Ir(i,j) \geq thr(i) \\ Ir(i,j) \in Clr_2, & Ir(i,j) < thr(i) \end{cases}$$

Similarly we can classify Green and Blue parts. Ir represents Red part of original image, similarly Ib and Ig can be used for Blue and Green part. thr (i) represents threshold of each row of Red part, similarly thg(i) and thb(i) can be used for Green and Blue part.
i and j represents rows and column respectively and i, j = 0, 1, 2……………..S-1. (Clr_1, Clg_1 and Clb_1) and (Clr_2, Clg_2 and Clb_2) represents classes belongs to great pixel value and small pixel value.
Using threshold calculated above, a mask image Mr, Mg and Mb for red, green and blue part respectively can be formed of size $S \times S$ by allotting 0 and 1 for each pixel value as given below.
Mask image for Red Part is

$$\begin{cases} Mr(i,j) = 1, & Ir(i,j) \geq thr(i) \\ Mr(i,j) = 0, & Ir(i,j) < thr(i) \end{cases}$$

Similarly we can generate mask images Mg and Mb for Green and Blue part respectively. Mr, Mg and Mb contain the information of classification results.

2.2 Decomposition of image data

After the classification of pixels, Original image is divided into six sequences, two for each Red, Green and Blue part and each sequence contains pixels of one class.

Here the need to arrange the pixels of R,G and B part of image along the scan line by study row by row as given below :

Dr={Ir (0, 0), Ir (0, 1), Ir (0, 2)…………..Ir (0, S-1)
………………………………………………………
Ir (S-1, 0), Ir (S-1, 1)…………..……..Ir (S-1, S-1)}
Similarly we can decompose Green (Dg) and Blue (Db) part of Image.

Red part will be decomposed into two sequences dr_1 and dr_2 based on threshold calculating in step first, method performed is given below.

$$\begin{cases} Ir(i,j) \in dr_1,\ Ir(i,j) \geq thr(i)\ or\ \ Mr(i,j)=1 \\ Ir(i,j) \in dr_2,\ Ir(i,j) < thr(i)\ or\ \ Mr(i,j)=0 \end{cases}$$

Similarly we can calculate above for Green and Blue part. Where (dr_1, dr_2), (dg_1, dg_2), and (db_1, db_2) represents subsequences for Dr, Dg and Db respectively and contains the pixels with great pixel value and small pixel value.

2.3 Hiding the mask image

Before hiding the mask image, the six decomposed sequences are combined into two sequences d1 and d2 depending upon their value calculated during decomposition.

d1 = [$dr_1\ dg_1\ db_1$] and d2 = [$dr_2\ dg_2\ db_2$]

For decoding process mask images Mr, Mg and Mb will be required and these mask images required extra storage space. To reduce this mask images will be hide in least significant bit (LSB) of d1 and d2. We need to arrange the mask images Mr, Mg and Mb according to scan line so that we can provide the exact value to LSB of exact pixel of the sequences d1 and d2.

Now we will arrange the pixels of Mr, Mg and Mb along the scan line, we get a sequence of binary value Br, Bg and Bb as given below:
Br= {Mr (0, 0), Mr (0, 1), Mr (0, 2)…….Mr (0, S-1)
………………………………………………………
Mr (S-1, 1), Mr (S-1, 2)……………..Mr (S-1, S-1)}
And this approach will be similar to Bg and Bb. Now we hide the first element for Br i.e. Mr (0, 0) into LSB of first element of d1, then the second element Mr (0, 1) into second element of d1 and the process will be followed accordingly. When all the elements of d1 are used then elements of d2 will be used to hide the rest elements of mask image.

To hide the mask image, we check the LSB of each element of d1 and d2. If the LSB of element or unassigned byte variable is 1, then we perform the "OR" operation with binary value of the element with 1 otherwise perform "AND" with 254.

$$El = \begin{cases} El \,\&\, 254, & if \quad LSB(El)=0 \\ El \,|\, 1, & if \quad LSB(El)=1 \end{cases}$$

And after performing this operation, result will be stored in two new sequences S1 and S2 as shown in Fig. 1.

2.4 Encoding and decoding using LZW, Huffman and RLE and getting LSB from sequences

Now these two sequences S1 and S2 will be encoded and decoded using LZW, Huffman and RLE. LZW is an error free compression approach and helps to remove spatial redundancy present in an image [4]. LZW assigns fixed length code word to variable length sequence of source symbols; it does not require prior knowledge of the probability of the occurrence of the symbols to be encoded. LZW is a dictionary based compression algorithm. This compression algorithm is used in many imaging file format GIF, TIFF and PDF [1].

Huffman coding is entropy coding algorithm and is used for lossless data compression. It is a form of statistical coding applicable to many form of data transmission like text file, program files

Run length encoding is a very simple form of data compression in which runs of data (that is, sequences in which the same data value occurs in many consecutive data elements) are stored as a single data value rather than as the original run. This is most useful on data that contains many such runs for example, simple graphic images such as icons, line drawings, and animations. Run-length encoding performs lossless data compression and is well suited to palette-based bitmapped images such as computer icons. It does not work well at all on continuous-tone images such as photographs, although JPEG uses it quite effectively on the coefficients that remain after transforming and quantizing image blocks. It is also used in fax machines.

After the decoding process we will get LSB from sequences S1 and S2 and mask images Mr, Mg and Mb will be extracted accordingly. After applying all the above step in reverse process the original image will be reconstructed.

III. EXPERIMENTAL RESULT AND COMPARATIVE ANALYSIS

The above algorithm is performed on color images of size $S \times S$ having 24bpp value using "Matlab" version 7.14.0.739. Some of the popular test images used are "Airplane", "Baboon", "Barbara", "House", "Leena", "Livingroom", "Pepper", and "Tree". From original image R, G and B part extracted first and then row by row classification and decomposition is performed which is followed by different encoding and decoding schemes. Fig. 2 shows the original image of "Leena" its mask images and reconstructed image.

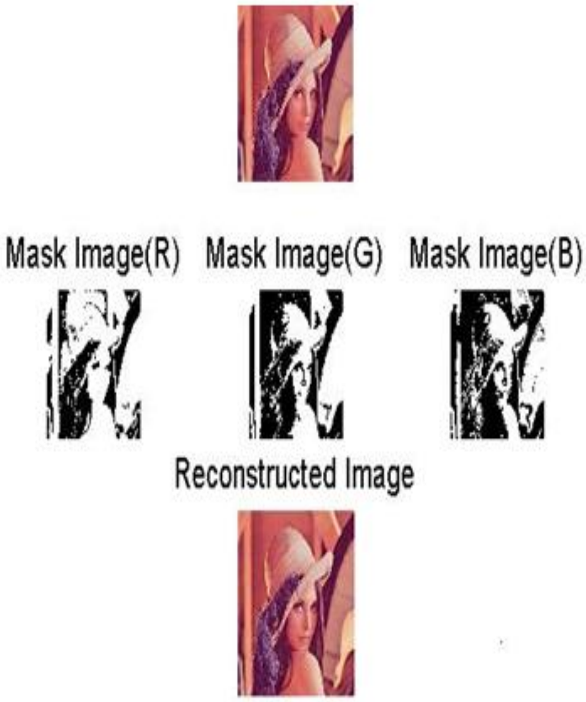

Fig. 2. Original Image, Mask Image (R, G and B) and Reconstructed Image


(a)


(b)


(c)


(d)


(e)


(f)


(g)


(h)

(a)Airplane; (b) Baboon; (c) Barbara; (d) House; (e) Leena; (f) Livingroom; (g) Peppers; (h) Tree;
Fig. 3. The Popular Test Images

The bpp value of the test images using the proposed algorithm with different encoding schemes can be analyzed through graph as shown in Fig. 4. Average bpp value calculated using simple RLE is 20.36625 and using the proposed algorithm with RLE is 16.88 and in LZW it is 23.4975 and 20.31375 and in case of Huffman the value is 22.14 and 19.42625.From this we conclude that the image has smaller bpp value when compressed through the proposed algorithm. During comparative analysis between different encoding schemes using algorithm, the results shows that RLE have smaller bpp value as compared to LZW and Huffman.

During the process of mask image hiding there is a possibility of change of LSBs, due to which there will be a noise in reconstructed image. However PSNR value of test images shows that there is very significant change in the quality of reconstructed images. It is very difficult to notice from naked eyes any difference between original image and reconstructed image. Table 1. shows PSNR values of test images

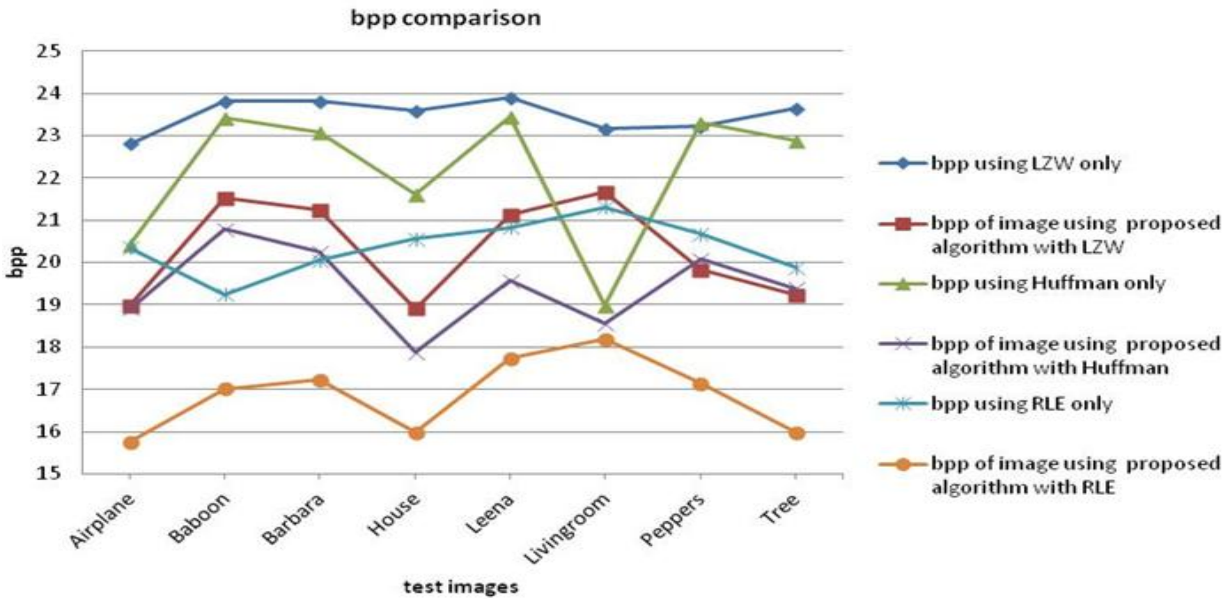

Fig. 4. The bpp Comparison of Image Using Different Methods

Table 1. PSNR of Test Images

Image name	Airplane	Baboon	Barbara	House	Leena	Livingroom	Peppers	Tree
PSNR	52.7.914	53.0024	52.9521	53.2185	52.9947	52.8819	53.0265	52.9242

IV. CONCLUSION AND FUTURE WORK

A near lossless compression algorithm using row by row classification, mask image hiding, decomposition and different encoding schemes like LZW,RLE and Huffman is proposed on color images. Using Above algorithm smaller bpp value is obtained as compared to simple LZW, RLE and Huffman encoding.

During comparative analysis between these encoding schemes using discussed algorithm we concluded that RLE have smaller bpp value as compared to other encoding schemes. In future task, other encoding schemes other than above discussed encoding techniques can be used on this correlation based algorithm to reduce the size of compressed image.

REFERENCES

[1] X. Yao-hua, T. Xiao-an and S. Mao-yin, "*Image compression based on classification row by row and LZW encoding*". *Congress on image and signal processing*, 2008, pp. 617-621.

[2] N.Ranganathan, S.G.Romaniuk and K.R. Namuduri, "*A lossless image compression algorithm using variable blocks size segmentation*". *IEEE Transactions on Image Processing*, Vol. 4, No. 10, pp. 1396-1406, 1995.

[3] M.-B. Lin, J.-F. Lee and G.E. Jan, "*A lossless data compression and decompression algorithm and its hardware architecture*". *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, Vol. 14, No. 9, pp. 925-936, 2006.

[4] M.Yang and N.Bourbakis, "*An overview of lossless digital image compression techniques*". *48th Midwest Symposium on Circuits and Systems*, 2005, pp. 1099-1102.

[5] K. Sayood and K. Anderson, "*A differential lossless image compression scheme*", *IEEE Transactions on Signal Processing*, Vol.*40*, No.1, pp. 236-241, 1992.

[6] S.A. Hassan and M. Hussain, "*Spatial domain lossless image data compression method". International conference on information and communication technologies (ICICT)*, 2011, pp. 1-4.

[7] K.S. Ng and L.M Cheng, "*SUB-BLOCK interchange for lossless image compression*". *IEEE Transactions on Consumer Electronics*, Vol.*45*, No.1, pp. 236-241, 1999.

[8] M. Burrows and D.J.Wheeler, "*A Block-sorting lossless data compression algorithm*", *SRC Research report 124, Digital Systems Research Center*, Palo, Alto, 1994.

[9] G.A. Triantafyllidis and M.G. Strintzis, "*A Context based adaptive Arithmetic coding technique for lossless image compression*". *Signal Processing Letters IEEE*, Vol.6, No.7, pp. 168-170, 1999.