International Journal of Computer & Mathematical Sciences
IJCMS
ISSN 2347 – 8527
Volume 6, Issue 8
August 2017

An Analysis Of Variation In Lossless Color Image Compression using n-MM Algorithm

Nitesh Agarwal, Suresh Kumar Jha, Amit Mishra
Jodhpur Institute of Engineering & Technology Jodhpur, India

ABSTRACT:-*Image compression is widely used in general life for many purposes. Image compression process reduces the required storage size of an image, to be stored in digital devices. Digital devices and computational resources have limited communication & storage capabilities. Hence there is a need of image compression for high quality digital images. For example, a single high quality image may require 10 to 100 million of bits for representation. If such type of images are in communication with low bandwidth network, then image compression becomes necessary before communication to cope up with this limitation. So image compression is an important aspect for a researcher that improves image compression process by introducing new techniques or methods in this field. This paper discusses an approach to embed n-MM method before entropy encoding. It uses repeated sequence of values before entropy encoding (RLE), to get high compression ratio.*

Keywords: -image, compression, n-MM, RLE, entropy, encoding.

INTRODUCTION

There are numerous applications of image processing, such as satellite imaging, medical imaging and video transmission, where the image size or image stream size is too large and requires a large amount of storage space or high bandwidth for communication in its original form. Every storage device & communication bandwidth cannot satisfy this requirement hence image compression techniques are used in such type of applications where image size is too large to store and for communication purposes. Image compression plays a very important role in application like tele-videoconferencing, remote sensing, document & medical imaging and facsimile transmission, which depends on the efficient manipulation, storage & transmission of binary, gray scale or color images.

Image compression techniques can be classified into two categories lossless image compression and lossy image compression. Images that provide numerical, secure & financial information are compressed using lossless image compression because original data is required after decompression process. But other images like multimedia images can be compressed using lossy image compression. The main reason to opt lossy image compression is thatthe human eye is very tolerant of approximation error in an image. Lossless image compression use some entropy encoding techniques like Run Length Encoding (RLE), Huffman Encoding, LZW (Lempel Ziv Welch) Encoding, and Area Encoding. This paper deals with RLE as an entropy encoding in lossless image compression. RLE entropy encoding gives a good compression ratio when image have repeated pixel values sequentially, but all the images do not have such type of repeated pattern. This paper suggests application of n-MM method to make repeated sequence before entropy encoding.

LOSSLESS COLOR IMAGE COMPRESSION

In lossless compression, every single bit of data will be available after decompression process. All the information will be completely restored. This techniques is applicable for text or spreadsheet files, where losing words or financial data could pose a problem. Process of lossless image compression is shown in fig 1 [5].

International Journal of Computer & Mathematical Sciences
IJCMS
ISSN 2347 – 8527
Volume 6, Issue 8
August 2017

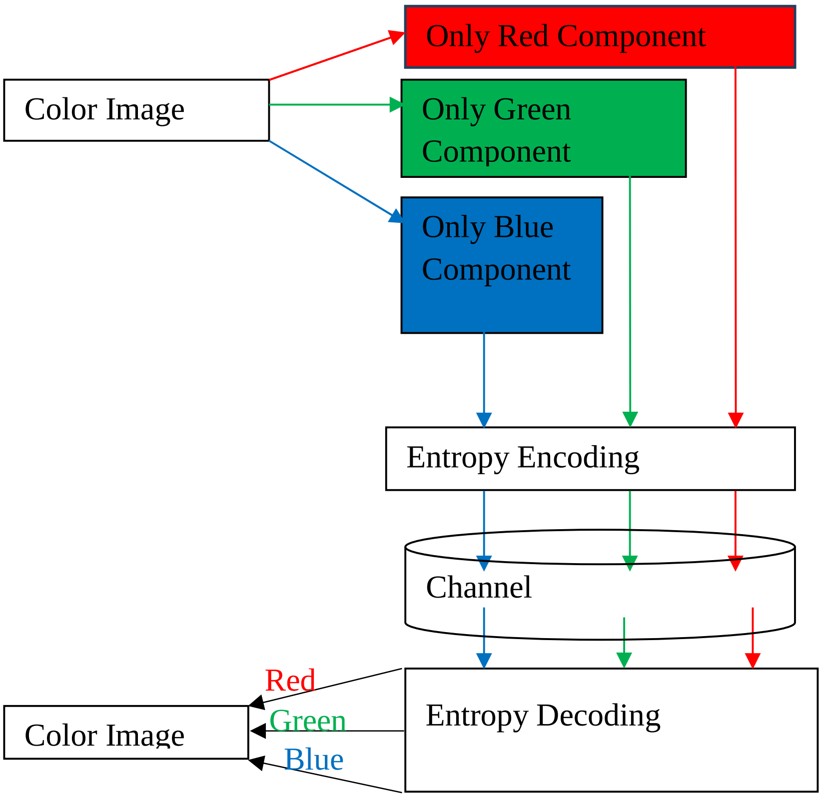

Fig 1: Lossless Color Image Compression

N-MM METHOD: -Most of the images a have co-relation among adjacent pixels. Therefore, finding a less correlated representation of image is one of the most important tasks. One of the basic concepts in compression is the reduction of redundancy and Irrelevancy. This can be done by removing duplication from the image. Sometime, Human Visual System (HVS) cannot notice some parts of the signal, i.e. omitting these parts will not be noticed by the receiver. It is called irrelevancy. n-MM method reads each pixel value row by row and divide each pixel value by *n*, if the pixel value is completely divisible by n, then there is no change in original pixel value and it remains same as previous value. But if pixel value is not completely divisible by n, and remainder value is being generated, then if this remainder less than n/2, then it will be subtracted from original pixel value and if not then remainder will be subtracted from n and the result of this subtraction will be added to the original pixel value. The basic idea in n-MM method is to check the whole pixels metrics and transform each pixel into a number divisible by n. For example suppose n=5 and the matrix of original pixel values is as given in Table 1. This table shows that RLE cannot be applied because pixel values are not being repeated in this table.

Table 1: Input Pixel matrix

121	122	122	123	124	125	105	110
130	132	132	131	134	135	133	220
221	222	222	223	224	225	205	300
425	426	427	500	501	502	501	905
521	522	522	523	524	525	555	660
630	632	632	631	634	635	633	633
851	852	852	963	964	965	205	300
425	426	427	500	501	502	501	905

So n-MM method can be applied to this table (as mentioned above) to convert this matrix in a form where RLE can be applied for compression, because RLE can be applied only for repeated values.

Table 2: Pixel matrix of After n-MM method with variable n =5

120	120	120	125	125	125	105	110
130	130	130	130	135	135	135	220
220	220	220	225	225	225	205	300
425	425	425	500	500	500	500	905
520	520	520	525	525	525	555	660
630	630	630	630	635	635	635	635
850	850	850	965	965	965	205	300
425	425	425	500	500	500	500	905

After the n-MM method pixel matrix contain a good no of repeated pixel as shown in table 2. This repeated pixel helps the RLE to compress pixel matrix.

n-MM algorithm
Input: Pixel matrix of input image.
Output: Modified Pixel Matrix.

```
{
  w = width of pixel matrix;
  h = height of pixel matrix;
pixel[h][w] = pixel matrix of original image;
for(i=0; i<h; i++)
   {
for(j=0; j<w; j++)
     {
If (pixel[i][j]%n>=n/2)
pixel[i][j] =  pixel[i][j]+(n-pixel[i][j]%n);
else
pixel[i][j] =  pixel[i][j]-pixel[i][j]%n;
    }
    }
}[6]
```

This method can be used, where some reduction in image quality can be tolerated.

RLE (RUN LENGTH ENCODING):- This is a very simple compression technique method used for compressing sequential data. Many digital images consist of repeated pixel values. In this case, RLE can be used for compression. In the proposed method, the output of n-MM method can be used by RLE technique, as this method converts pixel value to specific values to maintain repeated sequence. In our example, table 2 can be processed by RLE technique and it will generated table 3 for the given values.

Table 3: Pixel matrix of RLE method

Pixel Value	Repetition	Pixel Value	Repetition	Pixel Value	Repetition
120	3	205	1	630	4
125	3	300	1	635	4
105	1	425	3	850	3
110	1	500	4	965	3
130	4	905	1	205	1
135	3	520	3	300	1
220	1	525	3	425	3
220	3	555	1	500	4
225	3	660	1	905	1

Table 3 requires less storage space as compare to table 1 because Table 1 contains 64 values to store while table 3 requires only 30 values for storage [5]. Compression ratio can be calculated as:-

Compression Ratio (CR) = 64/30 = 2.13

METHEDOLOGY

Lossless Image Compression wirh n-MM algorithm

The lossless image compression steps can be explained using the following algorithm:-

1. Create pixel matrix of the image & divide it into separate blocks for each component (Red, Green, Blue)
2. Apply n-MM algorithm on each block.
3. Apply RLE on each block & store this encoded block on secondary storage.
4. To get required image, read encoded blocks from secondary storage & apply entropy decoding (Run Length Decoding) on each encoded block.
5. Combine all blocks to get pixel matrix.
6. Using pixel matrix, required image can be generated.

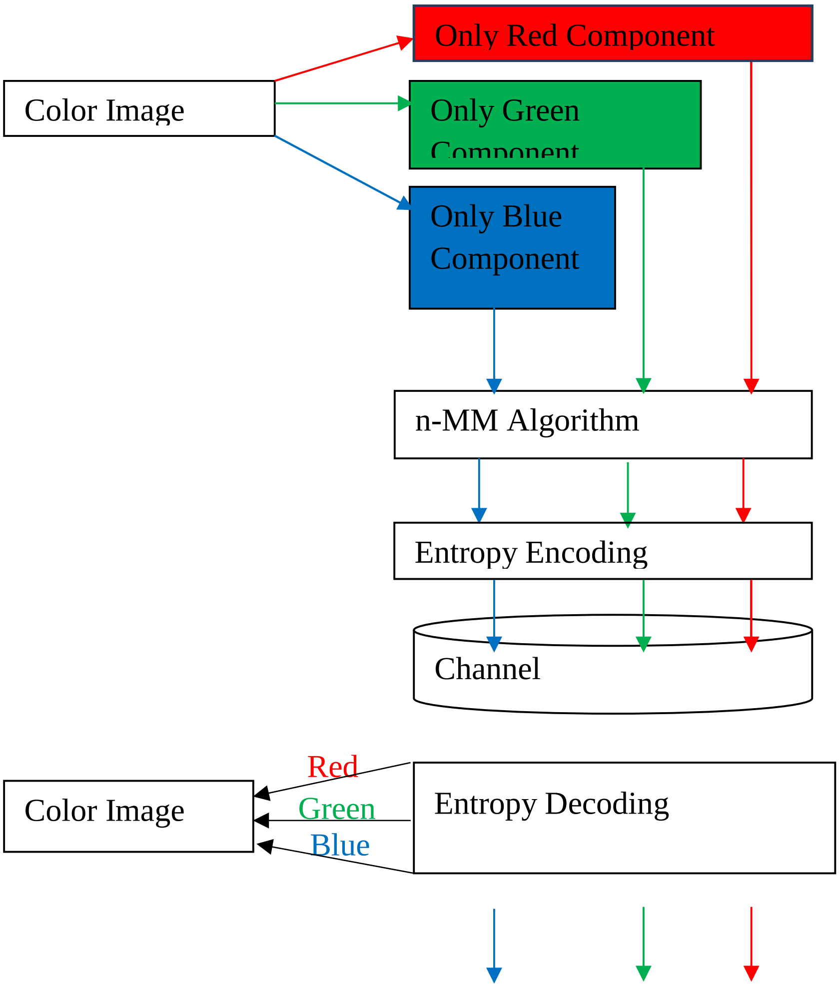

Fig 2: Lossless Color Image Compression Process with n-MM algorithm

International Journal of Computer & Mathematical Sciences
IJCMS
ISSN 2347 – 8527
Volume 6, Issue 8
August 2017

Now we can find MSE (Mean Squared Error), PSNR (Peak Signal to Noise Ratio) & CR (Compression Ration) to determine quality of image obtain by proposed method for each value variable n used in n-MM algorithm.

$$MSE_n = \frac{1}{H*W}\sum_{x=0}^{H-1}\sum_{y=0}^{W-1}[O(x,y) - M_n(x,y)]^2 \quad (1)$$

$$PSNR_n = 20*\log_{10}(MAX) - 10*\log_{10}(MSE_n) \quad (2)$$

$$CR_n = \frac{Original\ \ Image\ \ size}{Output\ \ Image\ \ size} \quad (3)$$

Where

H=Height of Image

W= Width of Image

MAX = maximum value of a pixel

MSE_n, $PSNR_n$&CR_n is MSE, PSNR& CR at variable n used in n-MM method.

Quality of image obtain by proposed method is depend on MSE_n&$PSNR_n$ value. If as the MSE value increases PSNR value decreases then we get a bad quality of image by proposed method & if as the MSE value decreases PSNR value increases we get a batter quality image hence on basis of this MSE_n&$PSNR_n$ value proposed method gives a best value of variable n on which we get an optimum compressed image with best quality.

OUTPUT

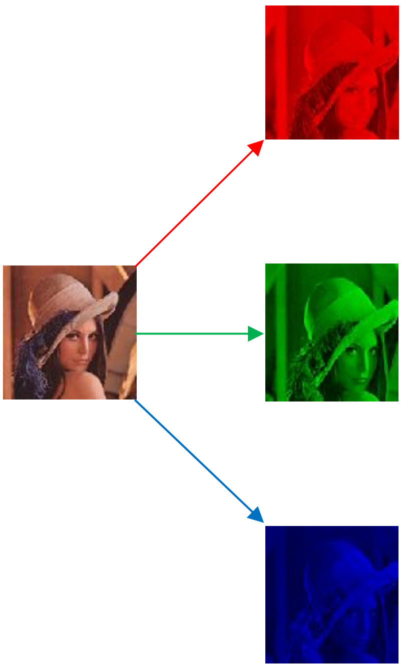

Fig 3: Color Components of input image

Table 4: Results

Value of n	Compressed Size in KB (for each component)			MSE	PSNR	CR
	Red	Green	Blue			
1	607	495	487	0.00	infinity	1.44
2	432	424	432	0.50	51.12	1.78
3	400	392	400	1.66	45.90	1.93
4	376	360	368	3.49	42.69	2.08
5	352	336	336	6.00	40.34	2.25
6	328	312	312	9.10	38.50	2.42
7	304	288	288	12.95	37.00	2.61
8	288	272	280	17.35	35.74	2.74
9	272	248	256	22.74	34.56	2.96
10	256	240	248	28.48	33.58	3.09

CONCLUSION

The result presented in this document shows that

1. The results shows that as the value of variable n increases storage size of image decreases as shown in table 4.
2. As the value of n increases CR_n also increases.
3. As the value of n increases proposed process add more noises in the image i.e. value of MSE_n increases as shown in table 4.

FUTURE SCOPE

Image compression process is very important field for researcher. There are wide range of scope in this field. Proposed method deals with variable n used in n-MM algorithm in future we try to improve this method so that a best value of n is taken dynamically on the basis of input image& network bandwidth.

REFERENCES

[1] A. H. Husseen , S. Sh. Mahmud & R. J. Mohammed " Image Compression Using Proposed Enhanced Run Length Encoding Algorithm ", Ibn Al-Haitham Journal For Pure And Applied Science, VOL 24(1), pp. 315-328, 2011.

[2] A.M. Raid, W.M. khedr, M.A. El-dosuky and wesan Ahmed " JPEG image compression using Discrete Cosine Transform – A survey", International Journal of Computer Science & Engineering Survey (IJCSES), vol 5, no 2, pp. 29-47, April 2014.

[3] Andrew B. Watson, "Image Compression Using Discrete Cosine Transform", NASA Ames Research Centre, 4(1), pp. 81-88,1994

[4] Anjali Kapoor and Dr. RenuDhir, "Image Compression Using Fast 2-D DCT Technique", International Journal on Computer Science and Engineering (IJCSE), vol. 3 pp. 2415-2419, 6 June 2011.

[5] Asha Lata&Permendersingh "Review of Image Compression Techniques", International Journal of Emerging Technology & Advanced Engineering (IJETAE), vol 3, issue 7, pp. 461-464, July 2013.

[6] Firas A. Jassim and Hind E. Qassim, " FIVE MODULUS METHOD FOR IMAGE COMPRESSION", signal & image processing: An International Journal (SIPIJ), vol 3, no.5, pp. 19-28, Octobar 2012

[7] Harley R. Myler and Arthur R. Weeks "The Pocket Handbook of Image Processing Algorithms in C", ISBN 0-13-642240-3 Prentice Hall P T R Englewood Cliffs, New Jercy 07632.

[8] Gregory K. Wallance, “The JPEG Still Picture Compression Standard”, IEEE Transaction on Consumer Electronics, Volume 38, No. 1, February 1992.

[9] Iain E.G. Richardson “H.264 and MPEG-4 Video Compression: Video Coding for Next-generation Multimedia”, ISBN 0470848375, 9780470848371, Wiley,2003.

[10] Jesse D. Kornblum “Using JPEG quantization tables to identify imagery processed by software”, ELSEVIER, DIGITAL INVESTIGATION 5, pp. S21-S25,2008.

[11] Maneesha Gupta and Dr.Amit Kumar Garg, “Analysis Of Image Compression Algorithm Using DCT” International Journal of Engineering Research and Applications (IJERA), vol.2, pp. 515-521, Jan-Feb 2012

[12] N.Ahmed, T.Natarajan, and K.R. Rao, “Discrete Cosine Transform”, IEEE Transactions on Computers, vol. C-32, pp. 90-93, Jan. 1974.

[13] Swati Dhamija and Priyanka Jain “Comparative Analysis for Discrete Sine Transform as a suitable method for noise estimation”IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 5, No 3, September 2011 pp. 162-164.

[14] V.Singh “Recent Patents on Image Compression – A Survey”, Recent Patents on signal Processing (Bentham Open), vol 2, pp. 47-62, 2010 .