A New Lossy Image Compression Technique Using DCT, Round Variable Method & Run Length Encoding

Nitesh Agarwal[1]
Department of Computer Science
Jodhpur Institute of Engineering & Technology
Jodhpur, India

Dr. Sandeep Mathur[2]
Department of Mathematics
Jodhpur Institute of Engineering & Technology
Jodhpur, India

Dr. Anjali Mathur[3]
Department of Mathematics
Jodhpur Institute of Engineering & Technology
Jodhpur, India

***Abstract*—An Image have important roles in human being life, it is used to store our memorable movement, used to represent a thing , used for communication purpose, even in video processing system it has important role because each frame of a video is made of an image. A high-quality image may require 10 to 100 million bits for representation. For example, a nearly photographic image requires approximately 1,280 rows of 800 pixels each, with 24 bits of color information per pixel, that is, a total of 24,576,000 bits, or 3,072,000 bytes. The large data files associated with images thus drive the need for extremely high compression ratios to make storage (particularly of movies) practical. Image compression process use two technique to compress image lossless image compression & lossy image compression. Images that provide numerical, Secure & financial information compressed using lossless image compression because we required original data back after decompression process. Lossless image compression use some entropy encoding technique but its compression ratio is low w.r.t lossy image compression. This paper describe a new lossy image compression technique by introducing an extra module named as Round Variable Method (RVM) in lossy image compression process This paper deals with comparative study of a compressed image on the basis of different value of round variable used in RVM.**

Key Words*: *RVM, DCT, RLE, MSE, PSNR.

1. INTRODUCTION

Digital devices & computational resources have limited communication & storage capabilities for example without compression, a CD with a storage capacity of approximately 600 million bytes only. Hence we need to compress image data for storing purpose as well as for communication over a network for example if there is a video conference organize by an organization which has a low bandwidth of a network, if image quality high for each frame of a video then it is difficult to deliver idea by video using low bandwidth over network provided by organization, hence organization need some image compression technique by which it compress the video before communication to provide a good communication in synchronize manner. Image compression process use two technique to compress image lossless image compression & lossy image compression. Images that provide numerical, Secure & financial information compressed using lossless image compression because we required original data back after decompression process. Lossless image compression use some entropy encoding technique but its compression ratio is low w.r.t lossy image compression. But other images like multimedia images can be compressed using lossy image compression. Lossy image compression require some transformation, quantization & entropy encodings to compress an image. Normally lossy image compression is process as follow

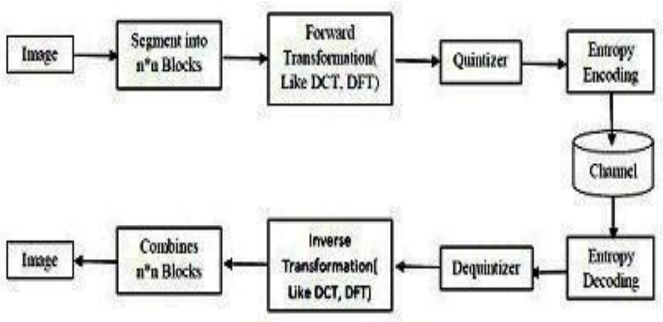

Fig 1: Lossy Image Comression

Because the human eye is very tolerant of approximation error in an image. Hence we may decide to exploit this tolerance to produce increased compression, at the expense of image quality by reducing some pixel data or information using this idea this paper introduce a new module named as RVM before entropy encoding techniques. In this paper RLE is used for encoding & RLE can compressed an image only if input values repeated at least for three times sequentially. Fig 2 describe the complete lossy image compression using RVM method.

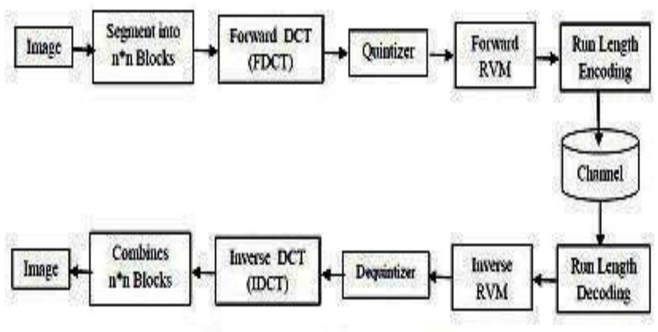

Fig 2: Proposed Lossy Image Compression

This lossy image compression mostly based on a transformation DCT (Discrete Cosine Transform), quantization of transform value using standard quantization table, RVM method & Run Length Encoding (RLE).

1.1 DCT (Discrete Cosine Transform) [1]

DCT convert an image into its equivalent frequency domain by partitioning image pixel matrix into blocks of size N*N. An image is a 2D pixel matrix hence 2D DCT is used to transform an image.

2-D DCT can be defined as

$$C(u,v)=\alpha(u)\alpha(v)\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} f(x,y)\cos\left[\frac{\pi(2x+1)u}{2N}\right]\cos\left[\frac{\pi(2y+1)v}{2N}\right] \quad (1)$$

for $u, v = 0,1,2,\ldots,N-1$.

& inverse transformation is defined as

$$f(x,y)=\sum_{u=0}^{N-1}\sum_{v=0}^{N-1} \alpha(u)\alpha(v)c(u,v)\cos\left[\frac{\pi(2x+1)u}{2N}\right]\cos\left[\frac{\pi(2y+1)v}{2N}\right] \quad (2)$$

Where $C(u,v)$ represents frequency value for u, v & $f(x,y)$ represents pixel color value at position (x, y).

$$\alpha(u)=\begin{cases}\sqrt{\frac{1}{N}} & for\ u=0\\ \sqrt{\frac{2}{N}} & for\ u\neq 0\end{cases} \quad (3)$$

$$\alpha(v)=\begin{cases}\sqrt{\frac{1}{N}} & for\ v=0\\ \sqrt{\frac{2}{N}} & for\ v\neq 0\end{cases} \quad (4)$$

1.2 Quantization

A Quantizer simply reduces the number of bits needed to store the transformed coefficients by reducing the precision of those values. Since this is a many-to-one mapping, it‟s a lossy process and is the main source of compression in an encoder.

The quantization matrix is designed to provide more resolution to more perceivable frequency components over less perceivable components (usually lower frequencies over high frequencies) in addition to transforming as many components to 0, which can be encoded with greatest efficiency. A DCT block is quantize using following formula

$$QDCT(i,j)= ROUND\left(\frac{DCT\ (i,j)}{QT\ (i,j)}\right) \quad (5)$$

& this QDCT block dequantize by following formula

$$DCT(i,j) = ROUND\ (QDCT(i,j) * QT(i,j)) \quad (6)$$

For i, j= 0, 1, 2, 3…….,N-1

Where (i,t) define position of input & output value, QDCT is DCT block after quantization, QT is standard quantization matrices & defined as

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

Table 1: Quantization Matrices [9]

In this lossy image compression QDCT provides input for RVM method.

1.3 RVM (Random Variable Method)

RVM take inputs from quantize block & round the input value according to its variable for ex RVM has input as 101,102,103,104,105,106,107,108,109,110 & variable used in RVM is 5 then it round the each input value as 100,100,100,105,105,105,105,110,110,110 the reason behind to use this method before encoding is that an entropy encoding technique gives a high compression as the input values have repeated data sequentially. & it work as

$$RVM(i,j) = ROUND\left(\frac{QDCT(i,j)}{X}\right) \quad (7)$$

& its inverse work as

$$QDCT(i,j) = ROUND\left(RVM(i,j) * X\right) \quad (8)$$

Where X is variable used in RVM method [10].

1.4 RLE (Run Length Encoding) [12]

This is a very simple compression technique method used for compressing sequential data. Many digital image consist pixel values that are repeats sequentially for such type of image RLE is useful. In proposed method RLE receive sequential data from RVM block & store input value that repeats & no of time that input value repeat sequentially. For ex. RVM block has data as

0	0	0	1
6	6	2	2
1	1	1	1
5	8	8	8

Fig 3: Pixel matrix of size 4*4

To store above 4*4 RVM block total 16 values are required to store but after applying RLE only 12 values are required to store such as

0	3
6	2
2	2
1	4
5	1
8	3

Fig 4: matrix After RLE

2. Main Results

2.1 Implementation of Proposed Lossy Image Compression

This paper describe how a b/w (8 bit) image compressed using proposed method

Steps involved in this implementation

1. Create pixel matrix of the image & divided it into blocks of size 8*8
2. Apply FDCT (Forward Discrete Sine Transform) on each 8*8 block of pixel matrix to get equivalent 8*8 DCT blocks using eq (1).
3. Apply eq (5) on each block of DCT to get QDCT block.
4. Apply eq (7) on each block of QDCT to get RVM block.
5. Combine each RVM block & apply RLE on combine block & store this encoded block on secondary storage.
6. To get Original image read RLE block from secondary storage & decode it to get combine RVM block & divide it into 8*8 small RVM block
7. Apply eq (8) on each RVM block to get QDCT blocks.
8. To get DCT blocks apply eq (6) on each QDCT block.
9. Apply eq (2) on each DCT block to get IDCT blocks.
10. Combine all IDCT blocks to get pixel matrix.
11. Using pixel matrix we get required image.
12. Now we 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 X variable used in RVM method. MSE_X, $PSNR_X$ & CR_X calculated by following formulas [11] -

$$MSE_x = \frac{1}{H * W} \sum_{x=0}^{H-1} \sum_{y=0}^{W-1} [O(x,y) - M_X(x,y)]^2 \quad (9)$$

$$PSNR_x = 20*\log_{10}(MAX) - 10*\log_{10}(MSE_x) \quad (10)$$

$$CR_x = \frac{Original\ Image\ size}{Output\ Image\ size} \quad (11)$$

Where H=Height of Image, W= Width of Image, variable MAX shows max value of a pixel for example here image is 8 bit hence MAX=255, MSE_x, $PSNR_x$ & CR_x is MSE, PSNR & CR at variable X used in RVM method.

Quality of image obtain by proposed method is depend on MSE & PSNR$_x$ 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$_x$ & PSNR$_x$ value proposed method gives a best value of X on which we get a high compressed image with best quality.

2.2 Outputs

2.2.1 Image Compression without RVM method

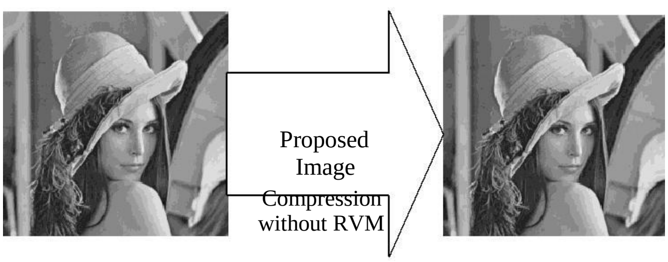

Fig 5: Uncompressed Image
Size= 768 KB

Fig 6: Compressed Image
Size= 103 KB

	MSE	PSNR	CR
Image Compression without RVM	17	36	7

Table 2: MSE, PSNR, CR without RVM

2.2.2 Image Compression with proposed method

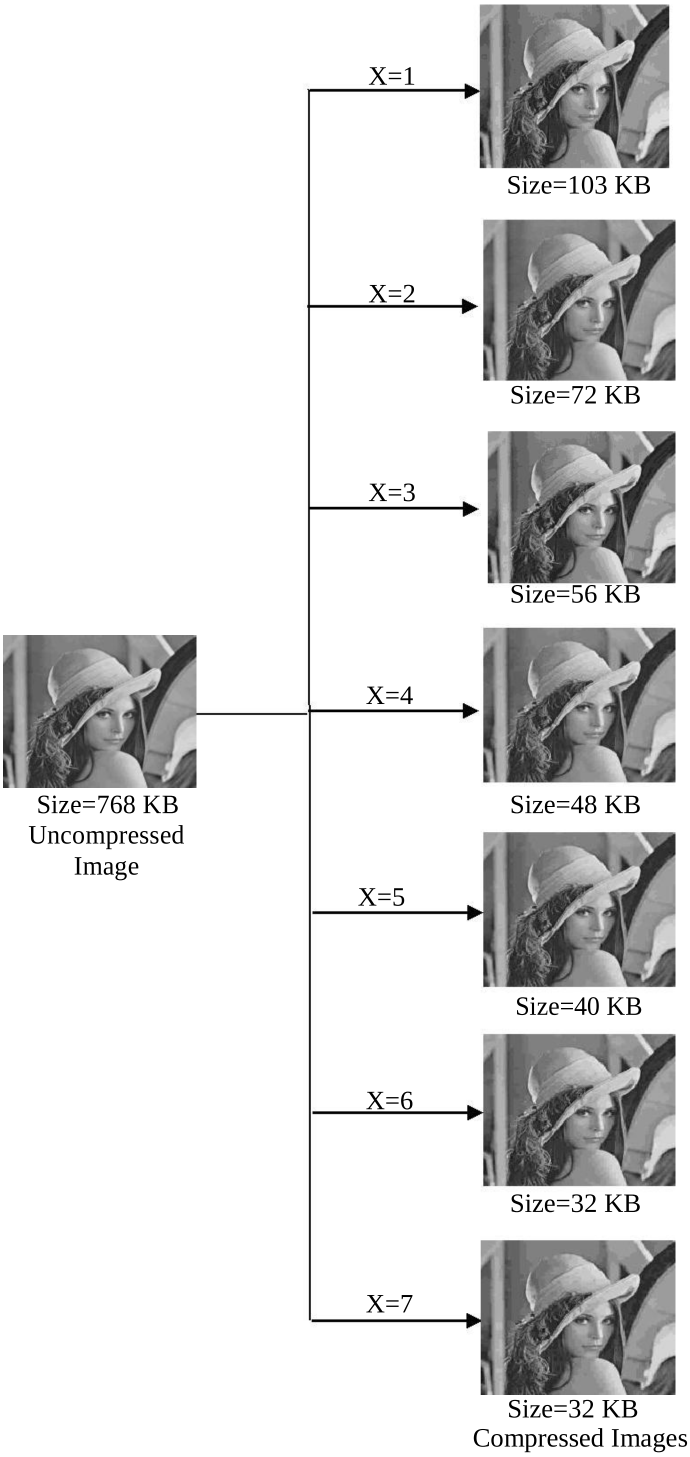

Fig 7: Proposed Lossy Image Compression with variation in RVM Variable X

X	MSE_X	$PSNR_X$	CR_X
1	17	36	7
2	41	32	11
3	38	32	14
4	54	31	16
5	61	30	19
6	80	29	24
7	87	29	24
8	107	28	24
9	121	27	32
10	138	27	32
11	153	26	32

Table 3: MSEx, PSNRx, CRx on different value of X

2.2.2.1 Graphs

1) RVM variable X vs. CR_x

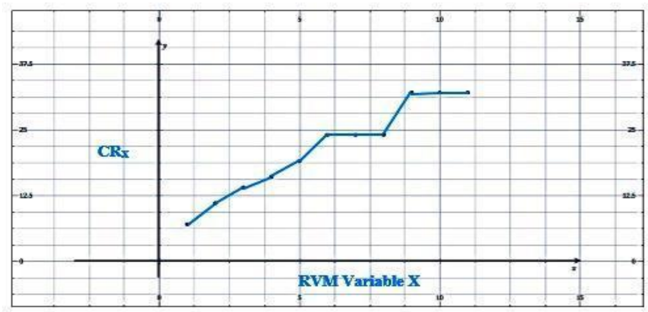

Fig 8: Variation in CRx with different value of RVM variable X

2) RVM variable X vs. MSE_x

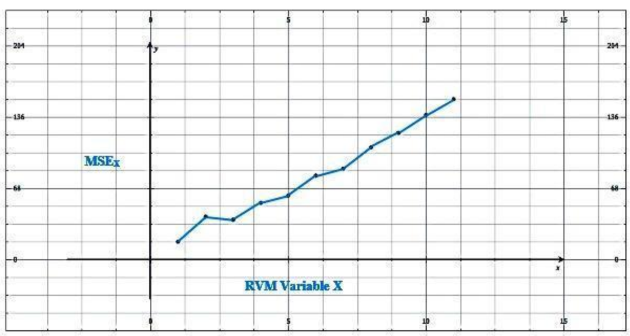

Fig 9: Variation in MSEx with different value of RVM variable X

3) RVM variable X vs. $PSNR_x$

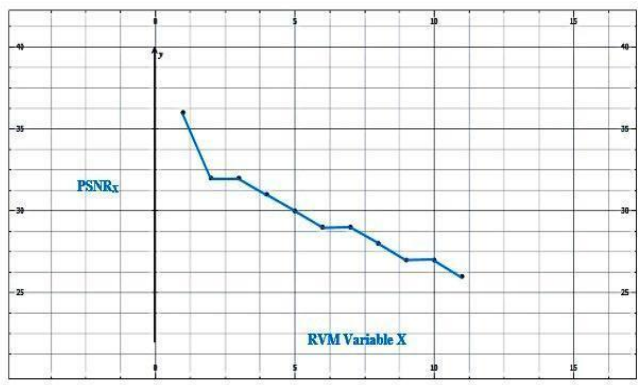

Fig 10: Variation in PSNRx with different value of RVM variable X

4) RVM variable X vs. $PSNR_x$ & MSE_x

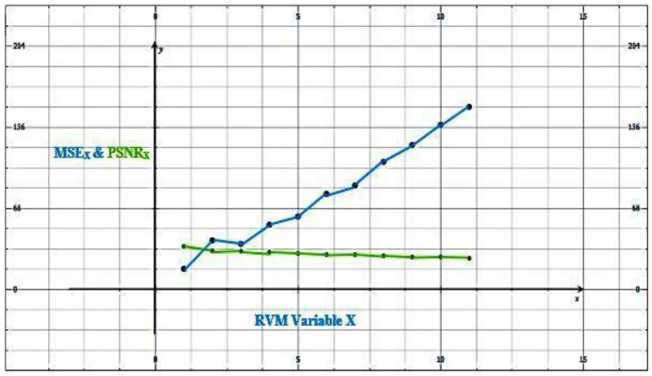

Fig 11: Variation in PSNRx & MSEx with different value of RVM variable X

5) MSE_x vs. $PSNR_x$

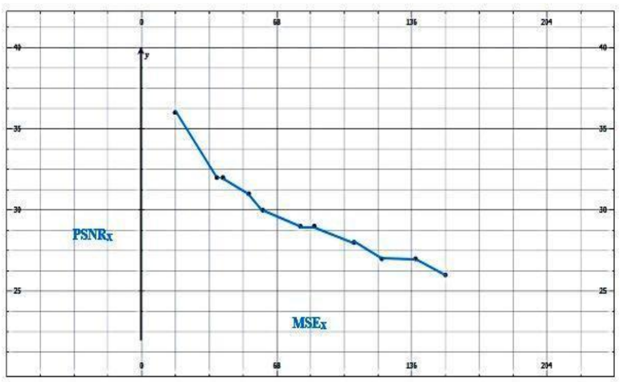

Fig 12: Graphical comparison between MSEx & PSNRx

3. CONCLUSION

The result presented in this document shows that

1. The results shows that as the value of variable X increases storage size of image decreases as shown in Fig 7.
2. As the value of X increases CR_x also increases as shown in Fig 8.
3. As the value of X increases proposed process add more noises in the image i.e. value of MSE_x increases as shown in Fig 9.
4. As the value of X increases $PSNR_x$ value decreases as shown in Fig 10.
5. As the MSE_x value decreases & $PSNR_x$ increases quality of image improves but CR_x decreases.
6. Fig 11 show that for value 2 & 3 of variable X we get good quality of compressed image.
7. Fig 11 shows that after the value 3 of variable X difference between MSE_x & $PSNR_x$ increases as the value of variable X increases i.e more noises is added to the image.
8. Fig 12 shows that as the value of MSE_x increases value of $PSNR_x$ variable decrease i.e. quality of image decreases.

REFERENCES

[1] N.Ahmed, T.Natatarajan, and K.R. Rao, "Discrete Cosine Transform", IEEE Transactions on Computers, vol. C-32, pp. 90-93, Jan. 1974.

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

[3] 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.

[4] Malini. S & Moni. R.S "Use of Discrete Sine Transform for A Noval Image Denoising Technique".

[5] R.C.Gonzalez, R.E.Woods, "*Digital Image Processing"* Pearson, 3rd Edition, India, 2009.

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

[7] Nitesh Agarwal and Dr. A.M. Khan "Application of DCT in image processing", International Journal of Engineering Research & Technology (IJERT), ISSN: 2278-0181, pp. 185-189, 2014.

[8] 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

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

[10] 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

[11] 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.

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

[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] 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

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