ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 5, Issue 6, December 2015

Performance analysis of color images with lossy compression by using inverse Haar matrix

Reem Talib Miri, Hussain Ali Hussain
Department of Mathematics, College of Science, Al- Mustansiriya University, Baghdad, Iraq

Abstract*— *Image compression involves reducing the size of image data files, while retaining necessary information. Compression is a necessary and essential method for creating image files with manageable and transmittable sizes. There have been many types of compression. This work exploits image compression based on Haar matrix. In the proposed method, compressed the RGB image lossy, the goal of this compression is to reduce size of images, and then the compression method is started by applying Haar matrix on image. Haar matrix works on numbers of adjacent pixels in a matrix which contains proximate colors into one color by calculating the average and the difference divided by two between each pixel and its adjacent pixel and repeat the process for all the pixels in rows and columns. We will notice that the pixels which contain proximate colors the difference will be very small and by using the threshold the smallest differences equal to zero and then when we reconstruct the image the colors of adjacent pixels will be the same and the size of the image will be smaller the original image. We measured the performance of this method by using Peak Signal to Noise Ratio (PSNR), Mean Square Error (MSE) and Compression Ratio (CR).

Index Terms— **Compression Ratio, Lossy compression, Mean Square Error, Peak Signal to Noise Ratio, RGB image.**

I. INTRODUCTION

In this paper, we compress colored images. The RGB image is divided into three different matrices and we use the same procedure for each matrix as we did for gray images. Then we combine these three matrices to get the RGB image back. In 24-bit color images each primary color (Red, Green, and Blue) is represented by one byte (8 bit) each byte represents the intensity of the color and ranges from 0 to 255.The darkest intensity or color value is 0 and the brightest value is 255[1].To get rid of redundant data we compress the image matrix and change it into matrix has high number of zero entries this matrix is called sparse[2],[3],[4],[5]. Which take up very less memory in the system? We can made sparse matrix by choosing a non-negative small value Ɛ where Ɛ is called threshold value. In the next step the elements of the matrix will be replaced by zero if the absolute value of elements in matrix are less than or equal to the Ɛ[6].

Mathematically we can write, if $CS_j = \{a_{ij}\}$ then

$$SCS_j = \begin{cases} a_{ij} & \text{if } |a_{ij}| > \varepsilon , \\ 0 & \text{if } |a_{ij}| \le \varepsilon . \end{cases}$$

We have two types of compression depend upon choosing the epsilon value. They are

A. Lossless compression

If we choose Ɛ= 0 this means that we cannot modify any of the elements and we will not lose original information. This compression is called lossless compression.

B. Lossy compression

If we choose Ɛ > 0 this means that we modify some of the elements and in this way we lose some of the information. This compression is called lossy compression.

By making the comparison between lossless and lossy compression, we observe the following things. In lossless compression we do not loose original information and we get original image back. In lossy compression we lose some of our original information and it reduces the image quality. We cannot get the original image back, we can only make the approximation of the original image. This method removes the details of image which are not noticeable by human eyes. The difference between the original image matrix and reconstructed matrix will depend upon choosing the epsilon value. If the epsilon is large then the difference will be large. If the epsilon is zero we do not loss any information about the image and we will get back the original image matrix. Compression ratio is used to measure the level of compression and it is defined as the ratio of the number of non-zero entries in the matrix before thresholding to the number of non zero entries in the matrix after thresholding (sparse matrix)[6] .

II. HAAR MATRIX AND INVERSE HAAR MATRIX

$$Ħ_{8\times8} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & 1 & -1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & -1 & -1 \\ 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \end{bmatrix}$$

The Haar matrix is the multiplication of the following three matrices. $Ħ_{8\times8} = \hbar_3 . \hbar_2 . \hbar_1$ [7]

Where $\hbar_1, \hbar_2, \hbar_3$ are:

$$\hbar_1 = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \end{bmatrix}$$

ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 5, Issue 6, December 2015

$$_2=\begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\hbar_3=\begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

The inverse of Haar matrix $Ħ_{8\times8}$ is:

$$(Ħ_{8\times8})^{-1} = \begin{bmatrix} 1/8 & 1/8 & 1/4 & 0 & 1/2 & 0 & 0 & 0 \\ 1/8 & 1/8 & 1/4 & 0 & -1/2 & 0 & 0 & 0 \\ 1/8 & 1/8 & -1/4 & 0 & 0 & 1/2 & 0 & 0 \\ 1/8 & 1/8 & -1/4 & 0 & 0 & -1/2 & 0 & 0 \\ 1/8 & -1/8 & 0 & 1/4 & 0 & 0 & 1/2 & 0 \\ 1/8 & -1/8 & 0 & 1/4 & 0 & 0 & -1/2 & 0 \\ 1/8 & -1/8 & 0 & -1/4 & 0 & 0 & 0 & 1/2 \\ 1/8 & -1/8 & 0 & -1/4 & 0 & 0 & 0 & -1/2 \end{bmatrix}$$

The inverse Haar matrix is the multiplication of the following three matrices. $(Ħ_{8\times8})^{-1}=(\hbar_3)^{-1}.(\hbar_2)^{-1}.(\hbar_1)^{-1}$

The inverse of $\hbar_1$, $\hbar_2$, $\hbar_3$ are:

$$(\hbar_1)^{-1}=\begin{bmatrix} 1/2 & 0 & 0 & 0 & 1/2 & 0 & 0 & 0 \\ 1/2 & 0 & 0 & 0 & -1/2 & 0 & 0 & 0 \\ 0 & 1/2 & 0 & 0 & 0 & 1/2 & 0 & 0 \\ 0 & 1/2 & 0 & 0 & 0 & -1/2 & 0 & 0 \\ 0 & 0 & 1/2 & 0 & 0 & 0 & 1/2 & 0 \\ 0 & 0 & 1/2 & 0 & 0 & 0 & -1/2 & 0 \\ 0 & 0 & 0 & 1/2 & 0 & 0 & 0 & 1/2 \\ 0 & 0 & 0 & 1/2 & 0 & 0 & 0 & -1/2 \end{bmatrix}$$

$$(\hbar_2)^{-1}=\begin{bmatrix} 1/2 & 0 & 1/2 & 0 & 0 & 0 & 0 & 0 \\ 1/2 & 0 & -1/2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1/2 & 0 & 1/2 & 0 & 0 & 0 & 0 \\ 0 & 1/2 & 0 & -1/2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$(\hbar_3)^{-1}=\begin{bmatrix} 1/2 & 1/2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1/2 & -1/2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

III. LOSSY COMPRESSION OF COLOR IMAGES

We will explain process on 8×8 image matrix. The process can be generalized to $2^j\times2^j$ matrix, where is j non-negative integers number. If there are 2^j elements in row and column in matrix, then the process of row and column will consist of j steps. In our case j =3.

Let R is a red matrix, G is a green matrix, and B is a blue matrix given below by using the command (imread image) in MATLAB.

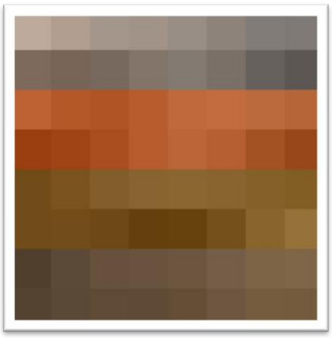

FIG (1) REPRESENTS COLOR IMAGE IS DIVIDED INTO THREE MATRICES

$$R=\begin{bmatrix} 187 & 174 & 165 & 158 & 151 & 136 & 125 & 122 \\ 136 & 131 & 133 & 143 & 144 & 132 & 114 & 102 \\ 173 & 163 & 158 & 165 & 175 & 177 & 170 & 164 \\ 141 & 146 & 158 & 171 & 177 & 169 & 153 & 140 \\ 120 & 128 & 139 & 144 & 145 & 143 & 138 & 133 \\ 106 & 107 & 103 & 94 & 94 & 109 & 130 & 143 \\ 82 & 93 & 102 & 104 & 108 & 118 & 124 & 127 \\ 81 & 87 & 91 & 90 & 97 & 107 & 114 & 113 \end{bmatrix}$$

$$G=\begin{bmatrix} 168 & 156 & 147 & 144 & 138 & 127 & 120 & 119 \\ 98 & 93 & 96 & 108 & 113 & 103 & 88 & 79 \\ 98 & 88 & 84 & 93 & 105 & 109 & 106 & 102 \\ 61 & 66 & 78 & 93 & 101 & 95 & 81 & 70 \\ 67 & 75 & 86 & 93 & 94 & 92 & 88 & 86 \\ 72 & 73 & 69 & 60 & 60 & 75 & 97 & 110 \\ 58 & 69 & 76 & 79 & 81 & 89 & 95 & 97 \\ 62 & 69 & 70 & 70 & 72 & 83 & 87 & 86 \end{bmatrix}$$

$$B=\begin{bmatrix} 154 & 142 & 137 & 133 & 130 & 118 & 114 & 112 \\ 77 & 72 & 77 & 88 & 95 & 85 & 71 & 63 \\ 59 & 49 & 45 & 53 & 69 & 72 & 70 & 65 \\ 12 & 17 & 29 & 44 & 52 & 46 & 31 & 19 \\ 15 & 23 & 34 & 40 & 41 & 39 & 35 & 30 \\ 27 & 28 & 24 & 14 & 14 & 27 & 46 & 59 \\ 32 & 41 & 49 & 49 & 51 & 57 & 61 & 61 \\ 45 & 49 & 51 & 46 & 50 & 57 & 60 & 57 \end{bmatrix}$$

Lossy compression by Haar matrix

ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 5, Issue 6, December 2015

It is compress columns and rows together in same step.

First step for lossy compression

The product of

$((\hbar_1)^{-1})^T \times R \times (\hbar_1)^{-1} = CSR_1$,

$((\hbar_1)^{-1})^T \times G \times (\hbar_1)^{-1} = CSG_1$,

$((\hbar_1)^{-1})^T \times B \times (\hbar_1)^{-1} = CSB_1$ are equivalent to the following procedures.

Divide elements of each column in R,G,B in to four pairs. Calculate the average of each of these pairs. These numbers will be the first four elements in corresponding column in B_{R1}, B_{G1}, B_{B1} new matrices. Subtract corresponding average from the first elements of the pairs. Calculated numbers are the last four elements in corresponding column in B_{R1}, B_{G1}, B_{B1}. Then divide elements of each row in B_{R1}, B_{G1}, B_{B1} in to four pairs. Calculate the average of each of these pairs. These numbers will be the first four elements in corresponding row in CSR_1 ,CSG_1 ,CSB_1. Subtract corresponding average from the first elements of the pairs. Calculated numbers are the last four elements in corresponding row in CSR_1 ,CSG_1 ,CSB_1 .
We get the matrix CSR_1 ,CSG_1 ,CSB_1.

$$CSR_1=\begin{bmatrix} 157 & 149.75 & 140.75 & 115.75 & 4.5 & -0.75 & 6.75 & 3.75 \\ 155.75 & 163 & 174.5 & 156.75 & 1.25 & -5 & 1.5 & 4.75 \\ 115.25 & 120 & 122.75 & 136 & -2.25 & 1 & -3.25 & -2 \\ 85.75 & 96.75 & 107.5 & 119.5 & -4.25 & -0.25 & -5 & -0.5 \\ 23.5 & 11.75 & 2.75 & 7.75 & 2 & 4.25 & 0.75 & -2.25 \\ 12.25 & -1.5 & 1.5 & 10.25 & 3.75 & 1.5 & -2.5 & -1.75 \\ 8.75 & 21.5 & 21.25 & -0.5 & -1.75 & -3.5 & 4.25 & 4.5 \\ 1.75 & 6.25 & 5.5 & 6 & -1.25 & -0.75 & 0 & -1 \end{bmatrix}$$

$$CSG_1=\begin{bmatrix} 128.75 & 123.75 & 120.25 & 101.5 & 4.25 & -2.25 & 5.25 & 2.5 \\ 78.25 & 87 & 102.5 & 89.75 & 1.25 & -6 & 0.5 & 3.75 \\ 71.75 & 77 & 80.25 & 95.25 & -2.25 & 0.5 & -3.25 & -2.75 \\ 64.5 & 73.75 & 81.25 & 91.25 & -4.5 & -0.75 & -4.75 & -0.25 \\ 33.25 & 21.75 & 12.25 & 18 & 1.75 & 3.75 & 0.25 & -2 \\ 14.75 & 1.5 & 4.5 & 14.25 & 3.75 & 1.5 & -2.5 & -1.75 \\ -0.75 & 12.5 & 12.75 & -8.25 & -1.75 & -4 & 4.25 & 3.75 \\ -1 & 3.75 & 3.75 & 4.75 & -1 & -0.75 & 0.75 & -0.75 \end{bmatrix}$$

$$CSB_1=\begin{bmatrix} 111.25 & 108.75 & 107 & 90 & 4.25 & -1.75 & 5.5 & 2.5 \\ 34.25 & 42.75 & 59.75 & 46.25 & 1.25 & -5.75 & 0.75 & 4.25 \\ 23.25 & 28 & 30.25 & 42.5 & -2.25 & 1 & -2.75 & -2 \\ 41.75 & 48.75 & 53.75 & 59.75 & -3.25 & 1.25 & -3.25 & 0.75 \\ 36.75 & 26.25 & 17 & 23 & 1.75 & 3.75 & 0.5 & -1.5 \\ 19.75 & 6.25 & 10.75 & 21.25 & 3.75 & 1.75. & -2.25 & -1.75 \\ -4.25 & 9 & 9.75 & -10 & -1.75 & -4 & 3.75 & 4.5 \\ -5.25 & 0.25 & 0.25 & 1.25 & -1.25 & -1.25 & 0.25 & -0.75 \end{bmatrix}$$

Second step for lossless compression

The product of

$((\hbar_2)^{-1})^T \times CSR_1 \times (\hbar_2)^{-1} = CSR_2$,

$((\hbar_2)^{-1})^T \times CSG_1 \times (\hbar_2)^{-1} = CSG_2$,

$((\hbar_2)^{-1})^T \times CSB_1 \times (\hbar_2)^{-1} = CSB_2$ are equivalent to the following procedures. We consider the first four elements in each column in CSR_1 ,CSG_1 ,CSB_1as two pairs. And calculate averages of these pairs. These numbers are the first two elements in corresponding columns in B_{R2}, B_{G2}, B_{B2} new matrices. The third and fourth elements of corresponding columns in B_{R2}, B_{G2}, B_{B2} are subtracting these averages from the first element of each pair. The last four elements in new columns are the same as the last four elements in corresponding columns in CSR_1 ,CSG_1 ,CSB_1. Then we consider the first four elements in each row in B_{R2}, B_{G2}, B_{B2} as two pairs. And calculate averages of these pairs. These averages are the first two elements in corresponding rows in CSR_2 ,CSG_2 ,CSB_2. The third and fourth elements of corresponding rows in CSR_2 ,CSG_2 ,CSB_2 are subtracting these averages from the first element of each pair. The last four elements in new rows are the same as the last four elements in corresponding rows in B_{R2}, B_{G2}, B_{B2}.

We get the matrices CSR_2 ,CSG_2 ,CSB_2 as following.

$$CSR_2=\begin{bmatrix} 156.375 & 146.9375 & 0 & 10.6875 & 2.875 & -2.875 & 4.125 & 4.25 \\ 104.4375 & 121.4375 & -3.9375 & -6.3125 & -3.25 & 0.375 & -4.125 & -1.25 \\ -3 & -18.6875 & 3.625 & 1.8125 & 1.625 & 2.125 & 2.625 & -0.5 \\ 13.1875 & 7.9375 & 1.5625 & -0.3125 & 1 & 0.625 & 0.875 & -0.75 \\ 17.625 & 5.25 & 5.875 & -2.5 & 2 & 4.25 & 0.75 & -2.25 \\ 5.375 & 5.875 & 6.875 & -4.375 & 3.75 & 1.5 & -2.5 & -1.75 \\ 15.125 & 10.375 & -6.375 & 10.875 & -1.75 & -3.5 & 4.25 & 4.5 \\ 4 & 5.75 & -2.25 & -0.25 & -1.25 & -0.75 & 0 & -1 \end{bmatrix}$$

$$CSG_2=\begin{bmatrix} 104.4375 & 103.5 & -0.9375 & 7.875 & 2.75 & -4.125 & 2.875 & 3.125 \\ 71.75 & 87 & -3.625 & -6.25 & -3.375 & -0.125 & -4 & -1.5 \\ 21.8125 & 7.375 & 3.4375 & 1.5 & 1.5 & 1.875 & 2.375 & -0.625 \\ 2.625 & 0.75 & 1 & -1.25 & 1.125 & 0.625 & 0.75 & -1.25 \\ 27.5 & 15.125 & 5.75 & -2.875 & 1.75 & 3.75 & 0.25 & -2 \\ 8.125 & 9.375 & 6.625 & -4.875 & 3.75 & 1.5 & -2.5 & -1.75 \\ 5.875 & 2.25 & -6.625 & 10.5 & -1.75 & -4 & 4.25 & 3.75 \\ 1.375 & 4.25 & -2.375 & -0.5 & -1 & -0.75 & 0.75 & -0.75 \end{bmatrix}$$

$$CSB_2=\begin{bmatrix} 74.25 & 75.75 & -1.5 & 7.625 & 2.75 & -3.75 & 3.125 & 3.375 \\ 35.4375 & 46.5625 & -2.9375 & -4.5625 & -2.75 & 1.125 & -3 & -0.625 \\ 35.75 & 22.75 & 2.75 & 0.875 & 1.5 & 2 & 2.375 & -0.875 \\ -9.8125 & -10.1875 & 0.5625 & -1.5625 & 0.5 & -0.125 & 0.25 & -1.375 \\ 31.5 & 20 & 5.25 & -3 & 1.75 & 3.75 & 0.5 & -1.5 \\ 13 & 16 & 6.75 & -5.25 & 3.75 & 1.75 & -2.25 & -1.75 \\ 2.375 & -0.125 & -6.625 & 9.875 & -1.75 & -4 & 3.75 & 4.5 \\ -2.5 & 0.75 & -2.75 & -0.5 & -1.25 & -1.25 & 0.25 & -0.75 \end{bmatrix}$$

Third step for lossless compression

The product of

$((\hbar_3)^{-1})^T \times CSR_2 \times (\hbar_3)^{-1} = CSR_3$,

$((\hbar_3)^{-1})^T \times CSG_2 \times (\hbar_3)^{-1} = CSG_3$,

$((\hbar_3)^{-1})^T \times CSB_2 \times (\hbar_3)^{-1} = CSB_3$ are equivalent to the following procedures. For every columns of CSR_2 ,CSG_2 ,CSB_2 we calculate the averages of the first and second elements. The averages will be the first elements in columns of B_{R3}, B_{G3}, B_{B3} (where B_{R3}, B_{G3}, B_{B3} are new matrices) and the second elements in columns of B_{R3}, B_{G3}, B_{B3} will be the subtracting the averages from the first elements in columns of CSR_2 ,CSG_2 ,CSB_2. Others elements in columns are the same

ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 5, Issue 6, December 2015

as in CSR_2 ,CSG_2 ,CSB_2.Then for every rows of B_{R3}, B_{G3}, B_{B3} we calculate the averages of the first and second elements. The averages will be the first elements in rows of CSR_3 ,CSG_3 ,CSB_3 and the second elements in rows of CSR_3 ,CSG_3 ,CSB_3 will be the subtracting the averages from the first elements in rows of B_{R3}, B_{G3}, B_{B3}. Others elements in rows are the same as in B_{R3}, B_{G3}, B_{B3}.

As following we will get CSR_3 ,CSG_3 ,CSB_3.

CSR_3=

132.2969	-1.8906	-1.9688	2.1875	-0.1875	-1.25	0	1.5
19.3594	6.6094	1.9688	8.5	3.0625	-1.625	4.125	2.75
-10.8438	7.8438	3.625	1.8125	1.625	2.125	2.625	-0.5
10.5625	2.625	1.5625	-0.3125	1	0.625	0.875	-0.75
11.4375	6.1875	5.875	-2.5	2	4.25	0.75	-2.25
5.625	-0.25	6.875	-4.375	3.75	1.5	-2.5	-1.75
12.75	2.375	-6.375	10.875	-1.75	-3.5	4.25	4.5
4.875	-0.875	-2.25	-0.25	-1.25	-0.75	0	-1

CSG_3=

91.6719	-3.5781	-2.2813	0.8125	-0.3125	-2.125	-0.5625	0.8125
12.2969	4.0469	1.3438	7.0625	3.0625	-2	3.4375	2.3125
14.5938	7.2188	3.4375	1.5	1.5	1.875	2.375	-0.6250
1.6875	0.9375	1	-1.25	1.125	0.625	0.75	-1.25
21.3125	6.1875	5.75	-2.875	1.75	3.75	0.25	-2
8.75	-0.625	6.625	-4.875	3.75	1.5	-2.5	-1.75
4.0625	1.8125	-6.625	10.5	-1.75	-4	4.25	3.75
2.8125	-1.4375	-2.375	-0.5	-1	-0.75	0.75	-0.75

CSB_3=

58	-3.1563	-2.2188	1.5313	0	-1.3125	0.0625	1.375
17	2.4063	0.7188	6.0938	2.75	-2.4375	3.0625	2
29.25	6.5	2.75	0.875	1.5	2	2.375	-0.875
-10	0.1875	0.5625	-1.5625	0.5	-0.125	0.25	-1.375
25.75	5.75	5.25	-3	1.75	3.75	0.5	-1.5
14.5	-1.5	6.75	-5.25	3.75	1.75	-2.25	-1.75
1.125	1.25	-6.625	9.875	-1.75	-4	3.75	4.5
-0.875	-1.625	-2.75	-0.5	-1.25	-1.25	0.25	-0.75

The number of non-zero entries in CSR_3 ,CSG_3 ,CSB_3 are 62 ,64 ,63 respectively. The value of threshold are T_R=4.8750 ,T_G=1.6875,T_B= 0.875.

The sparse matrices of CSR_3 ,CSG_3 ,CSB_3 are

$SCSR_3$=

132.2969	0	0	0	0	0	0	0
19.3594	6.6094	0	8.5	0	0	0	0
-10.8438	7.8438	0	0	0	0	0	0
10.5625	0	0	0	0	0	0	0
11.4375	6.1875	5.875	0	0	0	0	0
5.625	0	6.875	0	0	0	0	0
12.75	0	-6.375	10.875	0	0	0	0
0	0	0	0	0	0	0	0

$SCSG_3$=

91.6719	-3.5781	-2.2813	0	0	-2.125	0	0
12.2969	4.0469	0	7.0625	3.0625	-2	3.4375	2.3125
14.5938	7.2188	3.4375	0	0	1.875	2.375	0
0	0	0	0	0	0	0	0
21.3125	6.1875	5.75	-2.875	1.75	3.75	0	-2
8.75	0	6.625	-4.875	3.75	0	-2.5	-1.75
4.0625	1.8125	-6.625	10.5	-1.75	-4	4.25	3.75
2.8125	0	-2.375	0	0	0	0	0

$SCSB_3$=

58	-3.1563	-2.2188	1.5313	0	-1.3125	0	1.375
17	2.4063	0	6.0938	2.75	-2.4375	3.0625	2
29.25	6.5	2.75	0	1.5	2	2.375	0
-10	0	0	-1.5625	0	0	0	-1.375
25.75	5.75	5.25	-3	1.75	3.75	0	-1.5
14.5	-1.5	6.75	-5.25	3.75	1.75	-2.25	-1.75
1.125	1.25	-6.625	9.875	-1.75	-4	3.75	4.5
0	-1.625	-2.75	0	-1.25	-1.25	0	0

The number of non-zero entries in $SCSR_3$,$SCSG_3$,$SCSB_3$ are 15, 39, 49.

Compression ratio is the ratio of the number of non-zero entries in the matrices before thresholding to the number of non zero entries in the matrices after thresholding (sparse matrices) .

CR=(62/15)+(64/39)+(63/49)=4.1333+1.6410+1.2857=7.06

Then we can sent the image by mail in internet or load it in website to decompressed the image after downloaded from the internet. We will used Haar matrix to reconstruction image.

<u>First step for reconstruction</u>

The product of

$(\hbar_3)^T \times SCSR_3 \times \hbar_3 = RSR_1$,

$(\hbar_3)^T \times SCSG_3 \times \hbar_3 = RSG_1$,

$(\hbar_3)^T \times SCSB_3 \times \hbar_3 = RSB_1$ are equivalent to the following procedures.

For every columns of $SCSR_3$, $SCSG_3$, $SCSB_3$ we calculate the addition of, and subtraction between, the first two elements. And replace the original numbers with the above results. We apply the same procedure to all other columns to get matrices B_{R4}, B_{G4}, B_{B4}.Then for every rows of B_{R4}, B_{G4}, B_{B4} we calculate the addition of, and subtraction between, the first two elements. And replace the original numbers with the above results. We apply the same procedure to all other rows to get matrices RSR_1 , RSG_1 , RSB_1.

RSR_1=

158.27	145.05	0	8.5	0	0	0	0
106.33	119.55	0	-8.5	0	0	0	0
-3	-18.69	0	0	0	0	0	0
10.56	10.56	0	0	0	0	0	0
17.63	5.25	5.88	0	0	0	0	0
5.63	5.63	6.88	0	0	0	0	0
12.75	12.75	-6.38	10.88	0	0	0	0
0	0	0	0	0	0	0	0

ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 5, Issue 6, December 2015

$$RSG_1=\begin{bmatrix} 104.44 & 103.5 & -2.28 & 7.06 & 3.06 & -4.13 & 3.44 & 2.31 \\ 71.75 & 87 & -2.28 & -7.06 & -3.06 & -0.13 & -3.44 & -2.31 \\ 21.81 & 7.38 & 3.44 & 0 & 0 & 1.88 & 2.38 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 27.5 & 15.13 & 5.75 & -2.88 & 1.75 & 3.75 & 0 & -2 \\ 8.75 & 8.75 & 6.63 & -4.88 & 3.75 & 0 & -2.5 & -1.75 \\ 5.88 & 2.25 & -6.63 & 10.5 & -1.75 & -4 & 4.25 & 3.75 \\ 2.81 & 2.81 & -2.38 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$RSB_1=\begin{bmatrix} 74.25 & 75.75 & -2.22 & 7.63 & 2.75 & -3.75 & 3.06 & 3.38 \\ 35.44 & 46.56 & -2.22 & -4.56 & -2.75 & 1.13 & -3.06 & -0.63 \\ 35.75 & 22.75 & 2.75 & 0 & 1.5 & 2 & 2.375 & 0 \\ -10 & -10 & 0 & -1.56 & 0 & 0 & 0 & -1.38 \\ 31.5 & 20 & 5.25 & -3 & 1.75 & 3.75 & 0 & -1.5 \\ 13 & 16 & 6.75 & -5.25 & 3.75 & 1.75 & -2.25 & -1.75 \\ 2.38 & -0.13 & -6.63 & 9.88 & -1.75 & -4 & 3.75 & 4.5 \\ -1.63 & 1.63 & -2.75 & 0 & -1.25 & -1.25 & 0 & 0 \end{bmatrix}$$

Second step for reconstruction

The product of

$(\hbar_2)^T \times RSR_1 \times \hbar_2 = RSR_2$,

$(\hbar_2)^T \times RSG_1 \times \hbar_2 = RSG_2$,

$(\hbar_2)^T \times RSB_1 \times \hbar_2 = RSB_2$

are equivalent to the following procedures.

For every columns of RSR_1 , RSG_1 , RSB_1 we calculate the addition of the first and third elements and their subtraction. We also calculate the addition of the second and fourth elements and their subtraction. And replace the original numbers with the above results. We apply the same procedure to all other columns to get matrices B_{R5}, B_{G5}, B_{B5}.Then for every rows of B_{R5}, B_{G5}, B_{B5} we calculate the addition of the first and third elements and their subtraction. We also calculate the addition of the second and fourth elements and their subtraction. We repeat the same procedure for all rows to obtain matrices RSR_2 , RSG_2 , RSB_2.

$$RSR_2=\begin{bmatrix} 155.27 & 155.27 & 134.86 & 117.86 & 0 & 0 & 0 & 0 \\ 161.27 & 161.27 & 172.23 & 155.23 & 0 & 0 & 0 & 0 \\ 116.89 & 116.89 & 121.61 & 138.61 & 0 & 0 & 0 & 0 \\ 95.77 & 95.77 & 100.48 & 117.48 & 0 & 0 & 0 & 0 \\ 23.5 & 11.75 & 5.25 & 5.25 & 0 & 0 & 0 & 0 \\ 12.5 & -1.25 & 5.63 & 5.63 & 0 & 0 & 0 & 0 \\ 6.38 & 19.13 & 23.63 & 1.88 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$RSG_2=\begin{bmatrix} 127.41 & 125.09 & 117.94 & 103.81 & 3.06 & -2.25 & 5.81 & 2.31 \\ 76.91 & 88.34 & 103.19 & 89.06 & 3.06 & -6 & 1.06 & 2.31 \\ 69.47 & 74.03 & 79.94 & 94.06 & -3.06 & -0.13 & -3.44 & -2.31 \\ 69.47 & 74.03 & 79.94 & 94.06 & -3.06 & -0.13 & -3.44 & -2.31 \\ 33.25 & 21.75 & 12.25 & 18 & 1.75 & 3.75 & 0 & -2 \\ 15.38 & 2.13 & 3.875 & 13.625 & 3.75 & 0 & -2.5 & -1.75 \\ -0.75 & 12.5 & 12.75 & -8.25 & -1.75 & -4 & 4.25 & 3.75 \\ 0.44 & 5.19 & 2.81 & 2.81 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$RSB_2=\begin{bmatrix} 110.53 & 109.47 & 106.13 & 90.88 & 4.25 & -1.75 & 5.44 & 3.38 \\ 33.53 & 43.47 & 60.63 & 45.38 & 1.25 & -5.75 & 0.69 & 3.38 \\ 23.22 & 27.66 & 30.44 & 42.69 & -2.75 & 1.13 & -3.06 & -2 \\ 43.22 & 47.66 & 53.56 & 59.56 & -2.75 & 1.13 & -3.06 & 0.75 \\ 36.75 & 26.25 & 17 & 23 & 1.75 & 3.75 & 0 & -1.5 \\ 19.75 & 6.25 & 10.75 & 21.25 & 3.75 & 1.75 & -2.25 & -1.75 \\ -4.25 & 9 & 9.75 & -10 & -1.75 & -4 & 3.75 & 4.5 \\ -4.38 & 1.13 & 1.63 & 1.63 & -1.25 & -1.25 & 0 & 0 \end{bmatrix}$$

Third step for reconstruction

The product of

$(\hbar_1)^T \times RSR_2 \times \hbar_1 = RSR_3$,

$(\hbar_1)^T \times RSG_2 \times \hbar_1 = RSG_3$,

$(\hbar_1)^T \times RSB_2 \times \hbar_1 = RSB_3$

are equivalent to the following procedures.

For every columns of RSR2 ,RSG2 ,RSB2 we calculate the addition and subtraction of, first and fifth elements, Second and sixth elements, third and seventh elements, fourth and eighth elements. To get new matrices BR6, BG6, BB6 then for every rows of BR6, BG6, BB6 we calculate the addition and subtraction of first and fifth elements, second and sixth elements, third and seventh elements, fourth and eighth elements. To get matrices RSR3 ,RSG3 ,RSB3 . Then we combine these three matrices to get the RGB image

$$SR_3=\begin{bmatrix} 178.77 & 178.77 & 167.02 & 167.02 & 140.12 & 140.11 & 123.11 & 123.11 \\ 131.77 & 131.77 & 143.52 & 143.52 & 129.61 & 129.61 & 112.61 & 112.61 \\ 173.77 & 173.77 & 160.02 & 160.02 & 177.86 & 177.86 & 160.86 & 160.86 \\ 148.77 & 148.77 & 162.52 & 162.52 & 166.61 & 166.61 & 149.61 & 149.61 \\ 123.27 & 123.27 & 136.02 & 136.02 & 145.23 & 145.23 & 140.48 & 140.48 \\ 110.52 & 110.52 & 97.77 & 97.77 & 97.98 & 97.98 & 136.73 & 136.73 \\ 95.77 & 95.77 & 95.77 & 95.77 & 100.48 & 100.48 & 117.48 & 117.48 \\ 95.77 & 95.77 & 95.77 & 95.77 & 100.48 & 100.48 & 117.48 & 117.48 \end{bmatrix}$$

ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 5, Issue 6, December 2015

RSG_3=

165.47	155.84	148.34	145.34	136	124.38	122.13	121.5
95.47	92.84	97.34	109.34	111.5	99.88	90.13	81.5
99.09	85.47	84.47	96.47	105.63	108.5	103.25	102.13
60.84	62.22	80.22	92.22	102.88	95.75	79.5	71.38
63.91	73.53	82.41	90.66	93.5	91.88	87.25	84.38
68.91	71.53	65.41	57.66	59.5	74.88	96.25	108.38
66.84	72.97	79.09	79.34	79.31	86.19	94.56	99.19
65.97	72.09	68.72	68.97	73.69	80.56	88.94	93.56

RSB_3=

153.28	141.28	137.72	133.72	128.56	117.69	115.75	112
76.28	71.28	77.72	88.72	94.56	83.69	72.75	63
58.28	48.28	45.72	53.72	69.81	72.94	68.25	65
11.28	16.28	29.72	44.72	52.81	46.94	29.25	19
14.47	23.47	33.78	39.53	40.88	39.5	35.19	30.19
26.47	28.47	23.78	13.53	13.88	27.5	46.19	59.19
34.84	42.84	48.66	48.91	52.13	58.25	61.94	60.44
46.09	49.09	48.91	44.16	48.88	55	58.69	57.19

FIG (2) REPRESENTS COMBINE THE THREE MATRICES TO GET THE RECONSTRUCTED IMAGE

IV. THE QUALITY OF THE RECONSTRUCTED IMAGES MEASURED USING (MSE, PSNR)

Let *f(x,y)* represent an input image and let *r(x,y)* denote an estimate or approximation of *f(x,y)* that results from decompressing the input. For any value of *x* and *y*, the error *e(x,y)* between *f(x,y)* and *r(x,y)* can be defined as:

$$e(x,y)= f(x,y) - r(x,y)$$

noise is the error introduced through a lossy compression of the image[].

So that the total error between the two images is

$$\sum_{x=1}^{N} \sum_{y=1}^{N} [f(x,y) - r(x,y)]$$

Where the images are of size N×N.

A. The Mean Square Error (MSE)

It is defined as the commutative square error between the original and reconstructed images. MSE is found by taking the sum of squared error divided by the total number of pixels in the image as follow [6]:

$$MSE = \frac{1}{N \times N} \sum_{x=1}^{N} \sum_{y=1}^{N} [f(x,y) - r(x,y)]^2$$

The unit measure of MSE is decibels (dB).

B. The Peak Signal to Noise Ratio (PSNR)

In practice, the peak signal-to-noise ratio (PSNR) is used to measure the difference between two images. It is defined as:

$$PSNR = 20\log_{10}\left[\frac{(2^K - 1)}{\sqrt{MSE}}\right]$$

Where 2^K is the number of gray levels (typically $256=2^8$ for 8 bits) then,

$$PSNR = 20\log_{10}\left[\frac{255}{\sqrt{MSE}}\right]$$

The unit measure of PSNR is decibels (dB).The PSNR is very common in image processing. The values for PSNR in lossy image are between 30 and 50 dB. It gives better number in 40; however, the subjective judgment of the viewer also is regarded as an important measure, perhaps, being the most important measure [6].

The error between the original image and reconstructed image in the example are:

E_R=

8.23	-4.77	-2.02	-9.02	10.89	-4.11	1.89	-1.11
4.23	-0.77	-10.52	-0.52	14.39	2.39	1.39	-10.61
-0.77	-10.77	-2.02	4.98	-2.86	-0.86	9.14	3.14
-7.77	-2.77	-4.52	8.48	10.39	2.39	3.39	-9.61
-3.27	4.73	2.98	7.98	-0.23	-2.23	-2.48	-7.48
-4.52	-3.52	5.23	-3.77	-3.98	11.02	-6.73	6.27
-13.77	-2.77	6.23	8.23	7.52	17.52	6.52	9.52
-14.77	-8.77	-4.77	-5.77	-3.48	6.52	-3.48	-4.48

E_G=

2.53	0.16	-1.34	-1.34	2	2.63	-2.13	-2.5
2.53	0.16	-1.34	-1.34	1.5	3.13	-2.13	-2.5
-1.09	2.53	-0.47	-3.47	-0.63	0.5	2.75	-0.13
0.16	3.78	-2.22	0.78	-1.88	-0.75	1.5	-1.38
3.09	1.47	3.59	2.34	0.5	0.13	0.75	1.63
3.09	1.47	3.59	2.34	0.5	0.13	0.75	1.63
-8.84	-3.97	-3.09	-0.34	1.69	2.81	0.44	-2.19
-3.97	-3.09	1.28	1.03	-1.69	2.44	-1.94	-7.56

E_B=

0.72	0.72	-0.72	-0.72	1.44	0.31	-1.75	0
0.72	0.72	-0.72	-0.72	0.44	1.31	-1.75	0
0.72	0.72	-0.72	-0.72	-0.81	-0.94	1.75	0
0.72	0.72	-0.72	-0.72	-0.81	-0.94	1.75	0
0.53	-0.47	0.22	0.47	0.13	-0.5	-0.19	-0.19
0.53	-0.47	0.22	0.47	0.13	-0.5	-0.19	-0.19
-2.84	-1.84	0.34	0.09	-1.13	-1.25	-0.94	0.56
-1.09	-0.09	2.09	1.84	1.13	2	1.31	-0.19

Then

MSE_R= 48.5198, MSE_G= 6.3286, MSE_B= 0.9614.

$PSNR_R$= 31.2716, $PSNR_G$=40.1177, $PSNR_B$= 48.3016.

PSNR=(31.2716+40.1177+48.3016)/3=39.897≈40is the best number.

We considered a small matrix because it is difficult to show

ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 5, Issue 6, December 2015

the results on the large size of matrices like a matrix of size 512×512.Now we have RGB image of size 512×512 $=2^9 \times 2^9$. First we will extend the inverse Haar matrix up to the size of the RGB image. Later, we will apply the inverse Haar matrix to the image. The whole procedure will be carried out with the help of MATLAB R2015b . In our case the three matrices are of the dimension 512×512 $=2^9 \times 2^9$, so inverse Haar matrix method compressed the RGB image in ninth steps the first step compresses to 256×256 $= 2^8 \times 2^8$ the second step compresses to 128×128 $= 2^7 \times 2^7$ the third step compresses to 64×64 $= 2^6 \times 2^6$ the fourth step compresses to 32×32 $= 2^5 \times 2^5$ the fifth step compresses to 16×16 $= 2^4 \times 2^4$ the sixth step compresses to 8×8 $= 2^3 \times 2^3$ the seventh step compresses to 4×4 $= 2^2 \times 2^2$ the eighth step compresses to 2×2 $= 2^1 \times 2^1$ the ninth step compresses to 1×1 $= 2^0 \times 2^0$ then chosen a non-negative small value called threshold for three matrices make the PSNR near 40 . The elements of the matrices will be replaced by zero if the absolute value of elements in matrices are less than or equal to the threshold values and we get sparse matrices after that we can sent the image compression (sparse matrices) by mail in internet or load it in website. To decompressed the image after downloaded from the internet, we will used Haar matrix to reconstruction image in ninth steps the first step reconstructs to 2×2 $= 2^1 \times 2^1$ the second step reconstructs to 4×4 $= 2^2 \times 2^2$ the third step reconstructs to 8×8 $= 2^3 \times 2^3$ the fourth step reconstructs to 16×16 $= 2^4 \times 2^4$ the fifth step reconstructs to 32×32 $= 2^5 \times 2^5$ the sixth step reconstructs to 64×64 $= 2^6 \times 2^6$ the seventh step reconstructs to 128×128 $= 2^7 \times 2^7$ the eighth step reconstructs to 256×256 $= 2^8 \times 2^8$ the ninth step reconstructs to 512×512 $=2^9 \times 2^9$ and we have the reconstructed image.

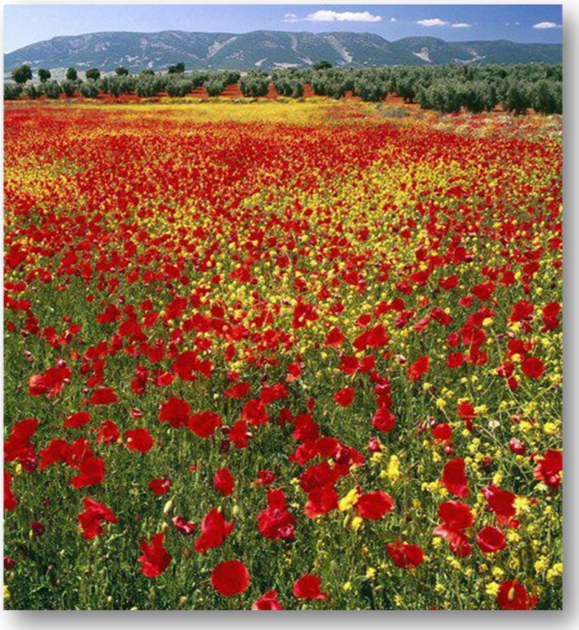

FIG (3) REPRESENTS RBG IMAGE

FIG (4) REPRESENTS THE RECONSTRUCTED IMAGE

T_R= 1.25 , T_G= 1.25, T_B= 1.1719

CR_R=1.2257, CR_G=1.2287, CR_B=1.2444

CR=1.2257+1.2287+1.2444=3.6988

MSE_R= 14.5115, MSE_G= 14.6908, MSE_B= 18.1484

$PSNR_R$=36.5137, $PSNR_G$=36.4604, $PSNR_B$= 35.5424

PSNR=(36.5137+36.4604+35.5424)/3=36.

APPENDIX

Program in Matlab (Lossy compression by Haar matrix)
It is compress columns and rows together in same step (matrix 512*512)

```
clear;close all;
IH1=imread('Buttercups.png');
   figure,imshow(IH1);
   %   read compression matrices
    A1 = csvread('A1.dat');
    A2 = csvread('A2.dat');
    A3 = csvread('A3.dat');
    A4 = csvread('A4.dat');
    A5 = csvread('A5.dat');
    A6 = csvread('A6.dat');
    A7 = csvread('A7.dat');
    A8 = csvread('A8.dat');
    A9 = csvread('A9.dat');
   %  transpose compression matrices
   AT1 = transpose(A1);
   AT2 = transpose(A2);
   AT3 = transpose(A3);
   AT4 = transpose(A4);
   AT5 = transpose(A5);
   AT6 = transpose(A6);
   AT7 = transpose(A7);
```

ISSN: 2277-3754
ISO 9001:2008 Certified
International Journal of Engineering and Innovative Technology (IJEIT)
Volume 5, Issue 6, December 2015

```
AT8 = transpose(A8);
AT9 = transpose(A9);
NIHCOM = IH1;
NIHDCOM = IH1;
nZT=0;
nZS=0;
SPSNR=0;
for gg = 1:3
%    read band number = gg
IH=IH1(:,:,gg);
IH = double(IH);
% compression image matrix
   IHCOM1 = A1 * IH * AT1;
   IHCOM2 = A2 * IHCOM1 * AT2;
   IHCOM3 = A3 * IHCOM2 * AT3;
   IHCOM4 = A4 * IHCOM3 * AT4;
   IHCOM5 = A5 * IHCOM4 * AT5;
   IHCOM6 = A6 * IHCOM5 * AT6;
   IHCOM7 = A7 * IHCOM6 * AT7;
   IHCOM8 = A8 * IHCOM7 * AT8;
   IHCOM9 = A9 * IHCOM8 * AT9;
   nZT =  nnz(IHCOM9);
   % THRESHOLDING image matrix
   cv = 1;
   fff = min(abs(IHCOM9));
   TS = max(fff)* cv;
   TSV(gg) = TS
   BW = IHCOM9 > TS;
   BW = BW .* IHCOM9;
   BS = IHCOM9 < -TS;
   BS = BS .* IHCOM9;
   NIHCOM = BW + BS;
   nZS =  nnz(NIHCOM);
   %   Decompress image matrix
   IHDCOM9 = A9\NIHCOM/AT9;
   IHDCOM8 = A8\IHDCOM9/AT8;
   IHDCOM7 = A7\IHDCOM8/AT7;
   IHDCOM6 = A6\IHDCOM7/AT6;
   IHDCOM5 = A5\IHDCOM6/AT5;
   IHDCOM4 = A4\IHDCOM5/AT4;
   IHDCOM3 = A3\IHDCOM4/AT3;
   IHDCOM2 = A2\IHDCOM3/AT2;
   IHDCOM1 = A1\IHDCOM2/AT1;
   NIHDCOM(: , :,gg) =IHDCOM1;
   ERRVAL = (IH - IHDCOM1).^2;
   SUMERR = sum(sum(ERRVAL,2));
   MSE = SUMERR/262144
   PSNR = 20 * log10(255/sqrt(MSE))
   SPSNR(gg) = PSNR
   CCR=nZT/nZS;
   CR(gg)=CCR
   FCR=sum(CR)
end
FPSNR = sum(SPSNR)/3;
figure,imshow(NIHDCOM);
title(['PSNR =(' num2str(SPSNR) ') FPSNR = '
num2str(FPSNR) ' TS =(' num2str(TSV) ') CR = '
num2str(CR)] );
filename =['comp_CR_' num2str(CR) '_PSNR = '
num2str(FPSNR) '.png'];
imwrite(NIHDCOM,filename,'png');
end.
```

V. CONCLUSIONS

1. When we want to compressed or decompressed the rows we multiplication from right and when we want to compressed or decompressed the columns we multiplication from left.
2. The number of steps in compression or reconstruction depended on the dimension of image.
3. The compression ratio can change when the number of applied steps is changed (increasing the number of steps leads to increasing the compression ratio).
4. Increasing the thresholds values leads to increasing the compression ratio.
5. The root mean square error can be reduced or Enlarged by changing the thresholds values.

REFERENCES

[1] Salomon D., "Data Compression the complete reference", Department of Computer Science, California State University ,Springer Verlag, Fourth Edition, 2007.

[2] Strang G. ,"Introduction to Linear Algebra", Fourth Edition, 2009.

[3] Lang S.," Linear Algebra ", Springer Verlag New York Inc., Third Edition, 1987.

[4] kolman B.," Introductory linear algebra with applications", Fourth Edition,1984.

[5] Frazier M. W.," An Introduction to Wavelets Through Linear Algebra ", Springer verlag New York,Inc. ,1999.

[6] Umbaugh S. E., "Computer Vision and Image processing", Prentice Hall Inc, 1998.

[7] Miri R. and Hussain A.," Reconstruction of lossless compressed fingerprint and x-ray images using Haar matrix technique " ,International Journal of Engineering and Applied Sciences (IJEAS),Vol.2,Iss.11,pp31-34,November 2015