Transmission and Reception of Image using Encryption and Decryption Technique

Anita Khandelwal[1], Bhavneesh Malik[2]
[1]M.Tech Scholar, RIEM Rohtak
[2]Assistant Professor, HOD ECE Department, R.I.E.M, MDU, Rohtak

***Abstract* -** Image compression is the application of Data compression on digital images. Digital images contain large amount of Digital information that need effective techniques for storing and transmitting large volume of data. Image compression techniques are used for reducing the amount of data required to represent a digital image. An Image can be compressed with use of Discrete Cosine Transformation (DCT), quantization encoding are the steps in the compression of the JPEG image format. The 2-D Discrete Cosine transform is used to convert the 8×8 blocks of image into elementary frequency components .The frequency components(DC and AC) are reduced to zero during the process of quantization which is a lossy process .The quantized frequency components are coded into variable length codeword's using encoding process. Distortion between the original image and reconstructed image is measured with PSNR (peak signal to noise ratio) with different compression factors. The compression ratio and PSNR values are different for different images.. It is found that performance will not remain same for different images even though compression factor was same.

Keywords *- Compression Ratio, Compression, Quantization, PSNR.*

I. INTRODUCTION

Compressing an image is significantly different than compressing raw binary data. Of course, general purpose compression programs can be used to compress images, but the result is less than optimal. This is because images have certain statistical properties which can be exploited by encoders specifically designed for them. Also, some of the finer details in the image can be sacrificed for the sake of saving a little more bandwidth or storage space. This also means that lossy compression techniques can be used in this area. Uncompressed multimedia (graphics, audio and video) data requires considerable storage capacity and transmission bandwidth. Despite rapid progress in mass-storage density, processor speeds, and digital communication system performance, demand for data storage capacity and data-transmission bandwidth continues to outstrip the capabilities of available technologies. The recent growth of data intensive multimedia-based web applications have not only sustained the need for more efficient ways to encode signals and images but have made compression of such signals central to storage and communication technology. For still image compression, the `Joint Photographic Experts Group' or JPEG standard has been established by ISO (International Standards Organization) and IEC (International Electro-Technical Commission). The performance of these coders generally degrades at low bit-rates mainly because of the underlying block-based Discrete Cosine Transform (DCT) scheme.

Like other transforms, the Discrete Cosine Transform (DCT) attempts to decorrelate the image data. After decorrelation each transform coefficient can be encoded independently without losing compression efficiency. This section describes the DCT and some of its important properties.

A. The One-Dimensional DCT

The most common DCT definition of a 1-D sequence of length N is

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

for $u = 0,1,2,\ldots,N-1$. Similarly, the inverse transformation is defined as

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

for $x = 0,1,2,\ldots,N-1$. In both equations (1) and (2) $\alpha(u)$ is defined as

The transform possesses a high energy compaction property which is superior to any known transform with a fast computational algorithm. [**1-51** The transform also possesses a circular convolution-multiplication relationship which can readily be used in linear system theory. ***[6***

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

It is clear from **(1)** that for u=0

$$C(u=0)=\sqrt{\frac{1}{N}}\sum_{x=0}^{N-1}f(x).$$

The first transform coefficient is the average value of the sample sequence. In literature, this value is referred to as the DC Coefficient. All other transform coefficients are called the AC Coefficients

B. The two-Dimensional DCT

The objective of this document is to study the efficacy of DCT on images. This necessitates the extension of ideas presented in the last section to a two-dimensional space. The 2-D DCT is a direct extension of the 1-D case and is given by

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

…. (1)

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

…. (2)

For $u, v = 0, 1, 2\ldots N-1$ and $\alpha(u)$ and $\alpha(v)$ are defined in **(3)**. The inverse transform is defined as

For $x, y = 0, 1, 2, \ldots, N-1$. The 2-D basis functions can be generated by multiplying the horizontally oriented 1-D basis functions (shown in Figure 3) with vertically oriented set of the same functions.

Types of Compression Systems: There are two types of compression systems

1. Lossy compression system: Lossy compression techniques can be used in images where some of the finer details in the image can be sacrificed for the sake of saving a little more bandwidth or storage space.
2. Loss less compression system: Lossless Compression System which aim at minimizing the bit rate of the compressed output without any distortion of the image. The decompressed bit-stream is identical to original bit-stream.

II. PROPERTIES OF DCT

A. Decorrelation

The principle advantage of image transformation is the removal of redundancy between neighboring pixels. This leads to uncorrelated transform coefficients which can be encoded independently. The normalized autocorrelation of the images before and after DCT is shown in Figure 5. Clearly, the amplitude of the autocorrelation after the DCT operation is very small at all lags. Hence, it can be inferred that DCT exhibits excellent decorrelation properties

B. Energy Compaction

Efficacy of a transformation scheme can be directly gauged by its ability to pack input data into as few coefficients as possible. This allows the quantizer to discard coefficients with relatively small amplitudes without introducing visual distortion in the reconstructed image. DCT exhibits excellent energy compaction for highly correlated images.

C. Separability

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

…… (3)

This property, known as separability, has the principle advantage that C (u, v) can be computed in two steps by successive 1-D operations on rows and columns of an image. This idea is graphically illustrated in Figure 8. The arguments presented can be identically applied for the inverse DCT computation

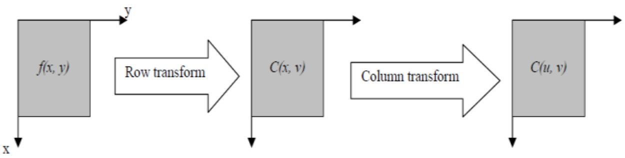


Fig.1: Separable, row-column decomposition

$$Z = AXA^T$$

…… (3)

$$a(k,n)=\sqrt{\frac{2}{N}}c(k)\cos\left[\frac{2\pi(2n+1)k}{4N}\right]$$

$$k,n=0,1,\ldots,N-1$$

$$c(0)=\sqrt{\frac{1}{2}} \quad \text{and} \quad c(k)=1 \quad \text{for} \quad k\neq 0$$

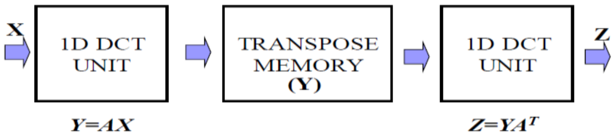


Fig.2: transition of DCT to IDCT Process

D. Symmetry

Another look at the row and column operations in Equation **6** reveals that these operations are functionally identical. Such a transformation is called a *symmetric transformation*. A separable and symmetric transform can be expressed in the form

$$T = AfA,$$

Where A is an $N\times N$ symmetric transformation matrix with entries $a(i, j)$ given by

$$a(i,j)=\alpha(j)\sum_{j=0}^{N-1}\cos\left[\frac{\pi(2j+1)i}{2N}\right],$$

…… (4)

And f *is* the $N\times N$ image matrix

This is an extremely useful property since it implies that the transformation matrix5 can be precomputed offline and then applied to the image thereby providing orders of magnitude improvement in computation efficiency.

E. Orthogonality

In order to extend ideas presented in the preceding section, let us denote the inverse transformation of **(7)** as

$$f = A^{-1}TA^{-1}.$$

…… (5)

DCT basis functions are orthogonal. Thus, the inverse transformation matrix of A is equal to its transpose i.e. $A^{-1}= A^T$. Therefore, and in addition to its decorrelation characteristics, this property renders some reduction in the pre-computation complexity

III. IMPLEMENTATION & PROCESS

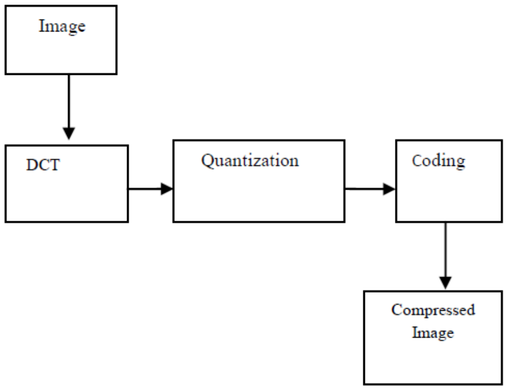


Fig.3: image Compression using DCT

The DCT transform the data from the spatial domain to the frequency domain. The spatial domain can give us an idea about the amplitude of the color as we move through space. The frequency domain explains that the amplitude of the color is varying fast from one pixel to the other pixel in an image data.

$$D(i,j) = \frac{1}{4} C(i)C(j) \sum_{x=0}^{7} \sum_{y=0}^{7} P(x,y) \cos\left[\frac{(2x+1)i\pi}{16}\right] \cos\left[\frac{(2y+1)j\pi}{16}\right]$$

$$C(u) = \begin{cases} \frac{1}{\sqrt{2}} & if\ u = 0 \\ 1 & if\ u > 0 \end{cases} \quad \ldots\ldots (6)$$

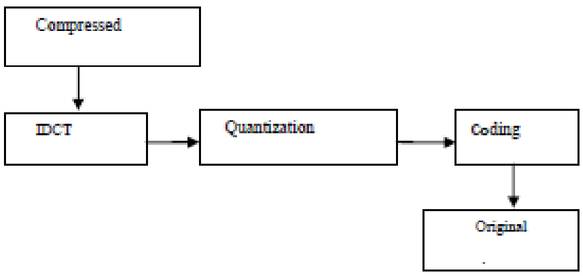


Fig.4: Image decompression using DCT

$$D(i,j) = \frac{1}{\sqrt{2N}} C(i)C(j) \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} P(x,y) \cos\left[\frac{(2x+1)i\pi}{2N}\right] \cos\left[\frac{(2y+1)j\pi}{2N}\right] \quad \ldots\ldots (7)$$

Quantization refers to the process of estimation of the continuous set of values in the image data with a finite set of values. The input values are taken as the original data, and the output values are constantly a finite number of levels [1]. The quantization is functions that have a set of output values are distinct, and generally finite. The quantization process is done by essentially dividing each component in the frequency domain by a constant for that component and then rounding nearest integer value. It is one of the lossy operations in the whole process of compression.

The main goal of quantization is to decrease most of the less important high frequency coefficients to zero, the more the zeros we can generate, better the image will compress. Quantization involves dividing each of the coefficients by an integer value between 1 and 255 and rounding off. The quantization table is selected to reduce the precision of each coefficient to not more than necessary. The quantization table is conceded along with the compressed file. Another expediency of this method is that it allows the user to customize the various level of compression at runtime to fine tune the quality or compression ratio. The following is the quantization table can be show as:

TABLE.1 QUANTIZATION MATIX TABLE

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

- 2-D DCT = 1-D DCT (row) → 1-D DCT (column)

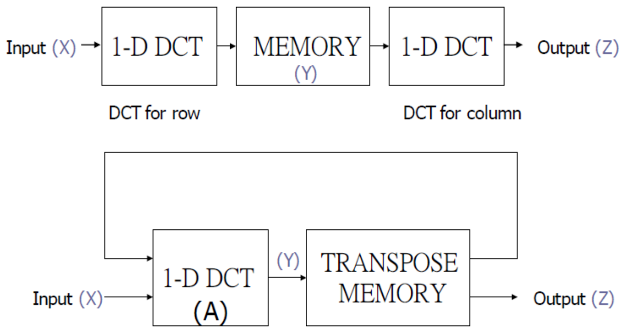


Fig.5: Use transposes memory

Transform coding constitutes an integral component of contemporary image/video processing applications. Transform coding relies on the premise that pixels in an image exhibit a certain level of correlation with their neighboring pixels. Similarly in a video transmission system, adjacent pixels in consecutive frames show very high correlation. Consequently, these correlations can be exploited to predict the value of a pixel from its respective neighbors. A transformation is, therefore, defined to map this spatial (correlated) data into transformed (uncorrelated) coefficients. Clearly, the transformation should utilize the fact that the information content of an individual pixel is relatively small i.e., to a large extent visual contribution of a pixel can be predicted using its neighbors. A typical image/video transmission system is outlined in Figure 1. The objective of the source encoder is to exploit the redundancies in image data to provide compression. In other words, the source encoder reduces the entropy, which in our case means decrease in the average number of bits required to represent the image. On the contrary, the channel encoder adds redundancy to the output of the source encoder in order to enhance the reliability of the transmission. In the source encoder exploits some redundancy in the image data in order to achieve better compression. The transformation sub-block de correlates the image data thereby reducing inter pixel redundancy. The transformation is a lossless operation; therefore, the inverse transformation renders a perfect reconstruction of the original image. The quantize sub-block utilizes the fact that the human eye is unable to perceive some visual information in an image. Such information is deemed redundant and can be discarded without introducing noticeable visual artifacts.

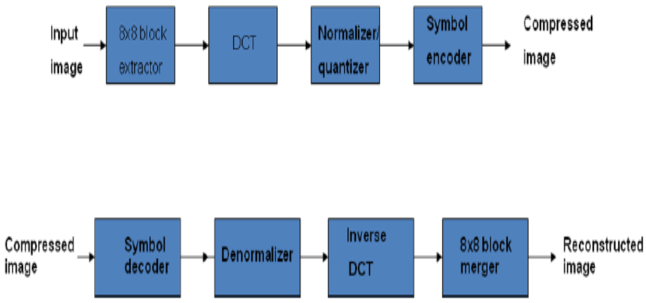


Fig.6: Compress and Reconstruction of Image process

IV. ALGORITHM

A. Proposed DCT Algorithm

The following is a general overview of the JPEG process.

- The image is broken into 8x8 blocks of pixels.
- Working from left to right, top to bottom, the DCT is applied to each block.
- Each block is compressed through quantization.
- The array of compressed blocks that constitute the image is stored in a drastically reduced amount of space.
- When desired, the image is reconstructed through decompression, a process that uses the inverse Discrete Cosine Transform (IDCT).

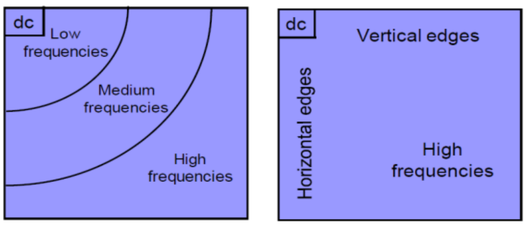


Fig.7: Overal l structure of Pixel

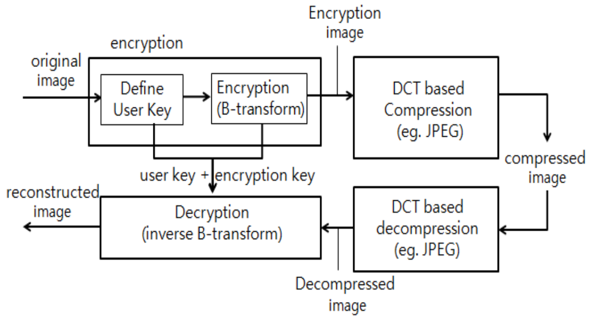


Fig.8: overall Algorithm for construction and reconstruction of image

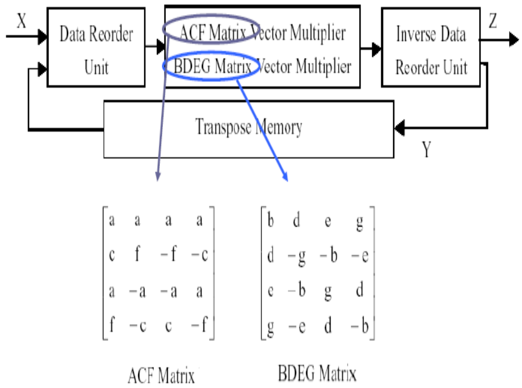


Fig.9: Overall structurfor matric composition

DCT

$$X(k) = e(k) \sum_{n=0}^{N-1} x(n) \cos\left[\frac{(2n+1)\ \pi\ k}{2N}\right],\ k = 0,1\ \ldots\ N-1$$

IDCT

$$x(n) = \frac{2}{N} \sum_{k=0}^{N-1} e(k) X(k) \cos\left[\frac{(2n+1)\ \pi\ k}{2N}\right],\ n = 0,1\ \ldots\ N-1$$

$$e(k) = \frac{1}{\sqrt{2}}\ \text{if}\ k = 0,\ e(k) = 1\ \text{othervise}$$

…… *(8)*

- Algorithm(Transmitting)
  1. First divide the original or target image into blocks and apply DCT(Discrete
  2. Cosine Transformation) on each blocks
  3. Then rotate the DCT blocks keep the direction of rotation as key for reconstructing the image
  4. Then cover the original image with another random image
  5. The random image is also divided into blocks and applies DCT on each block and the original image is covered by random image and it is send to the destination. This process is called embedding
- Algorithm (Receiver)
  6. The random image is taken out .This process is called extraction
  7. Using the rotation key the DCT blocks are reconstructed for the source image
  8. Then to each block we apply inverse DCT The image is reconstructed

B. Four Point DCT EXAMPLE

$$X(k)=e(k)\sum_{n=0}^{3} x(n)\ \cos\left[\frac{(2n+1)\ \pi\ k}{8}\right],\ k=0,1,2,3$$

$$X(0)=\frac{1}{\sqrt{2}}\sum_{n=0}^{3} x(n)=\frac{1}{\sqrt{2}}x(0)+\frac{1}{\sqrt{2}}x(1)+\frac{1}{\sqrt{2}}x(2)+\frac{1}{\sqrt{2}}x(3)$$

$$X(1)=\sum_{n=0}^{3} x(n)\cos\left[\frac{(2n+1)\ \pi}{8}\right]=x(0)\cos\frac{\pi}{8}+x(1)\cos\frac{3\pi}{8}+x(2)\cos\frac{5\pi}{8}+x(3)\cos\frac{7\pi}{8}$$

$X(2)=......$

$X(3)=......$

Coefficients

$$\begin{bmatrix}X(0)\\X(1)\\X(2)\\X(3)\end{bmatrix}=\begin{bmatrix}\frac{1}{\sqrt2} & \frac{1}{\sqrt2} & \frac{1}{\sqrt2} & \frac{1}{\sqrt2}\\ \cos\frac{\pi}{8} & \cos\frac{3\pi}{8} & \cos\frac{5\pi}{8} & \cos\frac{7\pi}{8}\\ \cos\frac{2\pi}{8} & \cos\frac{6\pi}{8} & \cos\frac{10\pi}{8} & \cos\frac{14\pi}{8}\\ \cos\frac{3\pi}{8} & \cos\frac{9\pi}{8} & \cos\frac{15\pi}{8} & \cos\frac{21\pi}{8}\end{bmatrix}\times\begin{bmatrix}x(0)\\x(1)\\x(2)\\x(3)\end{bmatrix}$$

$$\begin{bmatrix}X(0)\\X(1)\\X(2)\\X(3)\end{bmatrix}=\begin{bmatrix}\frac{1}{\sqrt2} & \frac{1}{\sqrt2} & \frac{1}{\sqrt2} & \frac{1}{\sqrt2}\\ \cos\frac{\pi}{8} & \cos\frac{3\pi}{8} & -\cos\frac{3\pi}{8} & -\cos\frac{\pi}{8}\\ \cos\frac{2\pi}{8} & -\cos\frac{2\pi}{8} & -\cos\frac{2\pi}{8} & \cos\frac{2\pi}{8}\\ \cos\frac{3\pi}{8} & -\cos\frac{\pi}{8} & \cos\frac{\pi}{8} & -\cos\frac{3\pi}{8}\end{bmatrix}\times\begin{bmatrix}x(0)\\x(1)\\x(2)\\x(3)\end{bmatrix}$$

Symmetric or anti-symmetric rows

$$\begin{bmatrix}X(0)\\X(1)\\X(2)\\X(3)\end{bmatrix}=\begin{bmatrix}\frac{1}{\sqrt2} & \frac{1}{\sqrt2} & \frac{1}{\sqrt2} & \frac{1}{\sqrt2}\\ \cos\frac{\pi}{8} & \cos\frac{3\pi}{8} & -\cos\frac{3\pi}{8} & -\cos\frac{\pi}{8}\\ \cos\frac{2\pi}{8} & -\cos\frac{2\pi}{8} & -\cos\frac{2\pi}{8} & \cos\frac{2\pi}{8}\\ \cos\frac{3\pi}{8} & -\cos\frac{\pi}{8} & \cos\frac{\pi}{8} & -\cos\frac{3\pi}{8}\end{bmatrix}\times\begin{bmatrix}x(0)\\x(1)\\x(2)\\x(3)\end{bmatrix}$$

$$\begin{bmatrix}X(0)\\X(1)\\X(2)\\X(3)\end{bmatrix}=\begin{bmatrix}c_2 & c_2 & c_2 & c_2\\ c_1 & c_3 & -c_3 & -c_1\\ c_2 & -c_2 & -c_2 & c_2\\ c_3 & -c_1 & c_1 & -c_3\end{bmatrix}\times\begin{bmatrix}x(0)\\x(1)\\x(2)\\x(3)\end{bmatrix}$$

N-1 (3) Coefficients C_1, C_2, C_3

$$X(0)=[x(0)+x(3)+x(1)+x(2)]\times c_2$$

$$X(1)=[x(0)-x(3)]\times c_1+[x(1)-x(2)]\times c_3$$

$$X(2)=[x(0)+x(3)-(x(1)+x(2))]\times c_2$$

$$X(3)=[x(0)-x(3)]\times c_3-[x(1)-x(2)]\times c_1$$

C. Defination of new variable

$$\begin{cases}X(0)=[P_0+P_1]\times c_2\\ X(1)=M_0\times c_1+M_1\times c_3\\ X(2)=[P_0-P_1]\times c_2\\ X(3)=M_0\times c_3-M_1\times c_1\end{cases},\ \text{where}\ \begin{cases}P_0=x(0)+x(3)\\ M_0=x(0)-x(3)\\ P_1=x(1)+x(2)\\ M_1=x(1)-x(2)\end{cases}$$

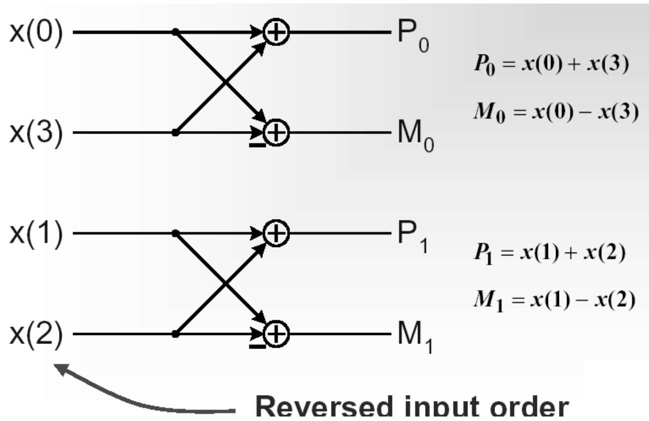


Fig.10 : Butterfly Diagram for DCT

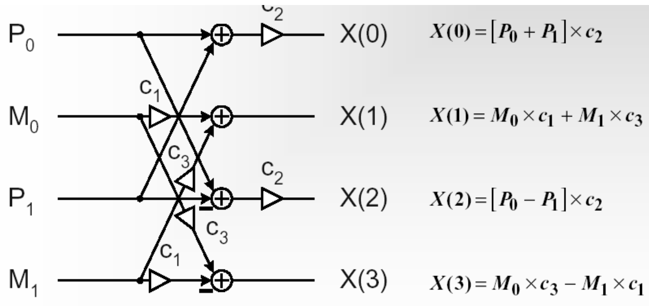
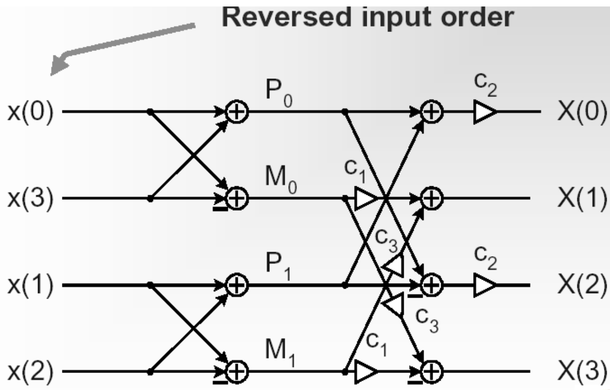


Fig.11 : Second Diagram of butter fly



D. Eight Point DCT EXAMPLE

$$\begin{bmatrix}X(0)\\X(1)\\X(2)\\X(3)\\X(4)\\X(5)\\X(6)\\X(7)\end{bmatrix}=\begin{bmatrix}c_4 & c_4 & c_4 & c_4 & c_4 & c_4 & c_4 & c_4\\ c_1 & c_3 & c_5 & c_7 & c_9 & c_{11} & c_{13} & c_{15}\\ c_2 & c_6 & c_{10} & c_{14} & c_{18} & c_{22} & c_{26} & c_{30}\\ c_3 & c_9 & c_{15} & c_{21} & c_{27} & c_1 & c_7 & c_{13}\\ c_4 & c_{12} & c_{20} & c_{28} & c_4 & c_{12} & c_{20} & c_{28}\\ c_5 & c_{15} & c_{25} & c_3 & c_{13} & c_{23} & c_1 & c_{11}\\ c_6 & c_{18} & c_{30} & c_{10} & c_{22} & c_2 & c_{14} & c_{26}\\ c_7 & c_{21} & c_3 & c_{17} & c_{31} & c_{13} & c_{27} & c_9\end{bmatrix}\begin{bmatrix}x(0)\\x(1)\\x(2)\\x(3)\\x(4)\\x(5)\\x(6)\\x(7)\end{bmatrix}$$

$$\begin{bmatrix} X(0)\\ X(1)\\ X(2)\\ X(3)\\ X(4)\\ X(5)\\ X(6)\\ X(7)\end{bmatrix} = \begin{bmatrix} c_4 & c_4 & c_4 & c_4 & c_4 & c_4 & c_4 & c_4\\ c_1 & c_3 & c_5 & c_7 & -c_7 & -c_5 & -c_3 & -c_1\\ c_2 & c_6 & -c_6 & -c_2 & -c_2 & -c_6 & c_6 & c_2\\ c_3 & -c_7 & -c_1 & -c_5 & c_5 & c_1 & c_7 & -c_3\\ c_4 & -c_4 & -c_4 & c_4 & c_4 & -c_4 & -c_4 & c_4\\ c_5 & -c_1 & c_7 & c_3 & -c_3 & -c_7 & c_1 & -c_5\\ c_6 & -c_2 & c_2 & -c_6 & -c_6 & c_2 & -c_2 & c_6\\ c_7 & -c_5 & c_3 & -c_1 & c_1 & -c_3 & c_5 & -c_7\end{bmatrix} \begin{bmatrix} x(0)\\ x(1)\\ x(2)\\ x(3)\\ x(4)\\ x(5)\\ x(6)\\ x(7)\end{bmatrix}$$

Fig.12 : Row-Column Method Example

E. Reduce an 8x8 matrix computation to two 4x4 matrix computation

DCT

$$\begin{bmatrix} Y(0)\\ Y(2)\\ Y(4)\\ Y(6)\end{bmatrix} = \begin{bmatrix} a & a & a & a\\ c & f & -f & -c\\ a & -a & -a & a\\ f & -c & c & -f\end{bmatrix} \begin{bmatrix} X(0)+X(7)\\ X(1)+X(6)\\ X(2)+X(5)\\ X(3)+X(4)\end{bmatrix}$$

$$\begin{bmatrix} Y(1)\\ Y(3)\\ Y(5)\\ Y(7)\end{bmatrix} = \begin{bmatrix} b & d & e & g\\ d & -g & -b & -e\\ e & -b & g & d\\ g & -e & d & -b\end{bmatrix} \begin{bmatrix} X(0)-X(7)\\ X(1)-X(6)\\ X(2)-X(5)\\ X(3)-X(4)\end{bmatrix}$$

IDCT

$$\begin{bmatrix} Y(0)\\ Y(1)\\ Y(2)\\ Y(3)\end{bmatrix} = \begin{bmatrix} a & c & a & f\\ a & f & -a & -c\\ a & -f & -a & c\\ a & -c & a & -f\end{bmatrix} \begin{bmatrix} X(0)\\ X(2)\\ X(4)\\ X(6)\end{bmatrix} + \begin{bmatrix} b & d & e & g\\ d & -g & -b & -e\\ e & -b & g & d\\ g & -e & d & -b\end{bmatrix} \begin{bmatrix} X(1)\\ X(3)\\ X(5)\\ X(7)\end{bmatrix}$$

$$\begin{bmatrix} Y(7)\\ Y(6)\\ Y(5)\\ Y(4)\end{bmatrix} = \begin{bmatrix} a & c & a & f\\ a & f & -a & -c\\ a & -f & -a & c\\ a & -c & a & -f\end{bmatrix} \begin{bmatrix} X(0)\\ X(2)\\ X(4)\\ X(6)\end{bmatrix} - \begin{bmatrix} b & d & e & g\\ d & -g & -b & -e\\ e & -b & g & d\\ g & -e & d & -b\end{bmatrix} \begin{bmatrix} X(1)\\ X(3)\\ X(5)\\ X(7)\end{bmatrix}.$$

$$Y = AX$$

$$A = \begin{bmatrix} a & a & a & a & a & a & a & a\\ b & d & e & g & -g & -e & -d & -b\\ c & f & -f & -c & -c & -f & f & c\\ d & -g & -b & -e & e & b & g & -d\\ a & -a & -a & a & a & -a & -a & a\\ e & -b & g & d & -d & -g & b & -e\\ f & -c & c & -f & -f & c & -c & f\\ g & -e & d & -b & b & -d & e & -g\end{bmatrix}$$

$$\begin{bmatrix} a\\ b\\ c\\ d\\ e\\ f\\ g\end{bmatrix} = \sqrt{\frac{2}{N}} \begin{bmatrix} \cos\frac{\pi}{4}\\ \cos\frac{\pi}{16}\\ \cos\frac{\pi}{8}\\ \cos\frac{3\pi}{16}\\ \cos\frac{5\pi}{16}\\ \cos\frac{3\pi}{8}\\ \cos\frac{7\pi}{16}\end{bmatrix}$$

DCT :

$$\begin{bmatrix} Y(0)\\ Y(2)\\ Y(4)\\ Y(6)\end{bmatrix} = \begin{bmatrix} a & a & a & a\\ c & f & -f & -c\\ a & -a & -a & a\\ f & -c & c & -f\end{bmatrix} \begin{bmatrix} X(0)+X(7)\\ X(1)+X(6)\\ X(2)+X(5)\\ X(3)+X(4)\end{bmatrix}$$

$$\begin{bmatrix} Y(1)\\ Y(3)\\ Y(5)\\ Y(7)\end{bmatrix} = \begin{bmatrix} b & d & e & g\\ d & -g & -b & -c\\ e & -b & g & d\\ g & -e & d & -b\end{bmatrix} \begin{bmatrix} X(0)-X(7)\\ X(1)-X(6)\\ X(2)-X(5)\\ X(3)-X(4)\end{bmatrix}$$

IDCT :

$$\begin{bmatrix} Y(0)\\ Y(1)\\ Y(2)\\ Y(3)\end{bmatrix} = \begin{bmatrix} a & c & a & f\\ a & f & -a & -c\\ a & -f & -a & c\\ a & -c & a & -f\end{bmatrix} \begin{bmatrix} X(0)\\ X(2)\\ X(4)\\ X(6)\end{bmatrix} + \begin{bmatrix} b & d & e & g\\ d & -g & -b & -e\\ e & -b & g & d\\ g & -e & d & -b\end{bmatrix} \begin{bmatrix} X(1)\\ X(3)\\ X(5)\\ X(7)\end{bmatrix}$$

$$\begin{bmatrix} Y(7)\\ Y(6)\\ Y(5)\\ Y(4)\end{bmatrix} = \begin{bmatrix} a & c & a & f\\ a & f & -a & -c\\ a & -f & -a & c\\ a & -c & a & -f\end{bmatrix} \begin{bmatrix} X(0)\\ X(2)\\ X(4)\\ X(6)\end{bmatrix} - \begin{bmatrix} b & d & e & g\\ d & -g & -b & -e\\ e & -b & g & d\\ g & -e & d & -b\end{bmatrix} \begin{bmatrix} X(1)\\ X(3)\\ X(5)\\ X(7)\end{bmatrix}$$

Pixel Domain

52	55	61	66	71	61	64	73
63	59	66	90	109	585	69	72
62	59	68	113	144	104	66	73
63	58	71	122	154	106	70	69
67	61	68	104	126	88	68	70
79	65	60	70	77	68	58	75
85	71	64	59	55	61	65	83
87	79	69	58	65	76	78	94

DCT ↓ ↑ IDCT

Frequency Domain (DC → 609; AC)

609	-29	-62	25	55	-20	-1	3
7	-21	-62	9	11	-7	-6	6
-46	8	77	-25	-30	10	7	-5
-50	13	35	-15	-9	6	0	3
11	-8	-13	-2	-1	1	-4	1
-10	1	3	-3	-1	0	2	-1
-4	-1	2	-1	2	-3	1	-2
-1	-1	-1	-2	-1	-1	0	-1

DC: $F(0,0) = (1/8) \Sigma\Sigma f(m,n)$

related to the average value of the block

Fig.13: Real time scenario

V. CRYPTOGRAPHY & OUTCOMES

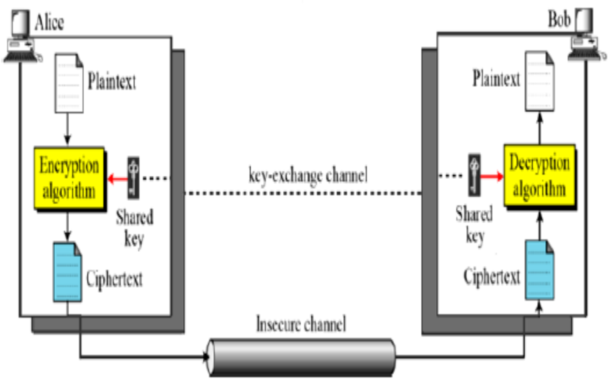


Fig.14: Encryption and decryption process

At first we divide original images to be transmitted into small square blocks and apply two-dimensional discrete cosine transform to each block and we obtain DCT components of each block.

For a fast communication, we would like to reduce the amount of transmitting data. Consequently, compression of the DCT components is required. In each block, most of DCT components have high energies in low frequency bands, we only use low frequency components through a simple low pass filter that is, left-up corners of each block with size of NC × NC are selected and higher frequency components are dropped. As a result of this process we can compress the transmitting images.

The DCT represents each block of image samples as a weighted sum of 2-D cosine functions (basis functions)

DCT Algorithm Classification

- Direct 2-D Method :- The 2-D transforms, DCT and IDCT, to be applied directly on the N x N input data items
- Row-Column Method
  - i. The 2-D transform can be carried out with two passes of 1-D transforms
  - ii. The separability property of 2-D DCT/IDCT allows the transform to be applied on one dimension (row) then on the other (column)
  - iii. Requires 2N instances of N-point 1-D DCT to implement an N x N 2-D DCT

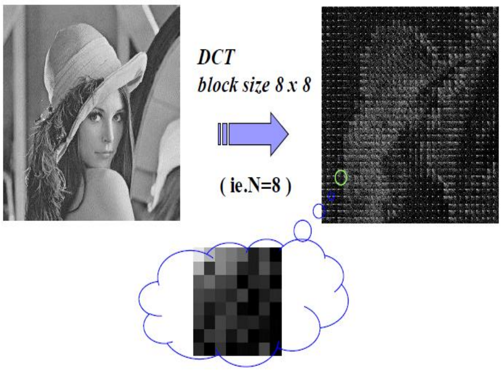


Fig.15: Outcomes of the program

Fig.16: Input Image

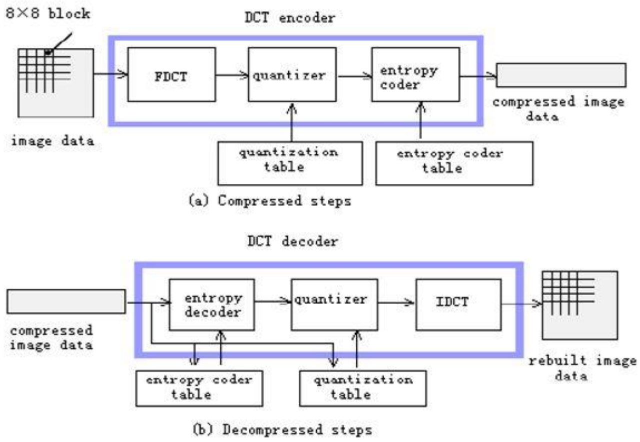


Fig.17: Block diagram of Compression and Decompression

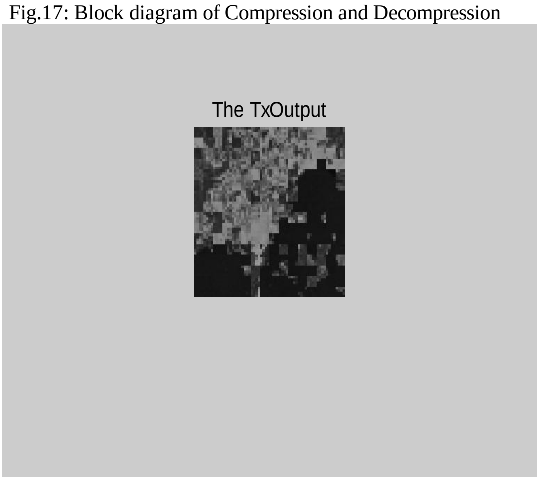


Fig.18: Compression Image

Fig.19: Decompression Image

VI. RESULT AND FUTURE SCOPE

Because the existing encryption methods do not consider compression, to compress the encrypted image, the new compression algorithm was needed. According to this necessity, several algorithms were proposed.

In this paper we proposed a new image encryption and compression method based on Embedding and Discrete

Cosine Transform (DCT). Using DCT images can be compressed. For encryption, DCT blocks of transmitted images are rotated and mixed with a random image to hide them. The compression efficiency of them is worse than the standard compression technique. So, we proposed the novel image encryption method which can improve the compression efficiency for the DCT-based standard compression method.

By reducing the discontinuity of the image using B-transform, the compression efficiency of proposed method is better than the standard compression method for the normal image. In future we can use RGB content as a key and use to encrypt and decrypt the image using RGB content.

ACKNOWLEDGMENT

This research was supported by the RIEM, Rohtak affiliated to MDU Rohtak under their strategic award program. I am thankful to Professor Bhavneesh Malik, HOD ECE Department for their support and proper giddiness.

VII. REFERENCES

[1] Lakhani, G, Modified JPEG Huffman coding, IEEE Transactions Image Processing, 12(2), 2003 pp. 159 – 169.

[2] S. Haykin. Communication Systems. John Wiley and Sons, Inc., 4th edition , pp 660-693

[3] T.Sreenivasulureddy, K.Ramani, S. Varadarajan and B.C.Jinaga, (2007,). "image Compression Using Transform Coding Methods", IJCSNS International Journal of Computer Science and Network Security, VOL. 7

[4] Abrams, M., and Podell, H. "Cryptography" Potentials, IEEE Page No 36-38.Issue:1,Volume: 20, Feb-Mar,2001J.

[5] Garfinkel, S.L; "Public Key Cryptography" , Computer, IEEE, Volume: 29, Issue:6, June 1996.

[6] Ahmed, N., T. Natarajan, and K. R. Rao. 1974. On image processing and a discrete cosine transform. IEEE Transactions on Computers C-23(1): 90-93.

[7] Image Encryption Using Advanced Hill Cipher Algorithm Bibhudendra Acharya1, Saroj Kumar Panigrahy2, Sarat Kumar Patra3, and Ganapati Panda3, Published in International Journal of Recent Trends in Engineering, Vol. 1, No. 1, May 2009

[8] en.wikipedia.org/wiki/Cryptography

[9] en.wikipedia.org/wiki/MATLAB

[10] users.ece.gatech.edu/~skhire3/files/**DCT**_image_enc.pdf

[11] en.wikipedia.org/wiki/Discrete_cosine_transform

[12] A. M. Eskicloglu, "Protecting intellectual property in digital multimedia networks," *IEEE Trans. on Computer*, vol. 36, no. 7, pp. 39-45, July 2003.

[13] N. S. Kulkarni, I. Gupta, and S. N. Kulkarni, "A robust image encryption technique based on random vector," in *Proc. 1st Int. Conf. on Emerging Trends in Engineering and Technology (ICETET)*, July 2008, pp. 15-19.

Anita khandelwal pursuing **MTECH** from Rohtak Institute of Engg. Management, Rohtak(2013-2015) and obtained **B.Tech in Electronics and Communication Engineering from R.I.E.M.** College affiliated to Maharshi Dayanand University in 2012. In 2009, I have passed **polytechnic Diploma in ECE Deptt**. From Chottu Ram Polytechnic, Rohtak, Affiliated to HSBTE (Panchkula). His main research interests include: Signal Processing and Image Processing.

Bhavneesh Malik received the Masters degree in Electronics and Telecommunications from **JRN Rajasthan Vidyapeeth University,** Udaipur, Rajasthan, India in 2006. He received his B.E degree in **Electronics and Communication from Maharishi Dayanand University (MDU), Rohta**k, India in 2003. At present he is working as Head of Deptt. ECE R.I.E.M., MDU (Rohtak). His main research interests includes: Wireless Sensor Networks, Wireless Body Area Networks, MAC Protocols, Next generation Wireless Networks and Image Processing. He has teaching experience of more than 10 years.