

# Information on Image Compression Using Cosine Transformation without Content Degradation

**Abdullah K-K. A.**

Department of Computer Science  
Olabisi Onabanjo University  
Ago Iwoye, Ogun State, Nigeria.  
+2348060046592,  
uwaizabdullah9@gmail.com

**Robert A.B.C.**

Department of Computer Science  
University of Ibadan  
Ibadan Oyo State, Nigeria  
abc.robert@ui.edu.ng  
+2347033701248

## ABSTRACT

Compression is an act of reducing the size of data or information, however, information compression minimise the size in bytes of file content without degrading the quality of the information to an unacceptable level. The reduction in file size allows more information to be stored in a given amount of memory space. It reduces the time required for information to be processed over the Internet or Web pages. In this paper, we demonstrate and critically discuss information compression using Cosine Transform with dimensions that are not in binary form even though compression factor are the same, performance would not remain the same for different images.

**Keywords:** Compression, Cosine Transform, Quantization, Degradation

---

## Aims Research Journal Reference Format:

Abdullah K-K. A.& Robert A.B.C. (2016): Information On Image Compression Using Cosine Transformation Without Content Degradation. *Advances in Multidisciplinary Research Journal*. Vol 2, No.2 Pp 307-314

---

## 1. INTRODUCTION

Compression reduced the quality of data in representing file and image content without degradation in quality of original information content. The influence of information compression on image and computer graphics applications, particularly those generating digital photographs and other complex colour images can generate very large file sizes. The problem faced on storage space and the requirements to speedily transmit information on image data over the Internet have therefore led to the development of different techniques of image compression. This reduces the physical size of files, local, global mean square error and increase the qualitatively the perception visualization of high and low contrast text pattern. Using quantization frequency compression method may not achieve the objective due to lack of standard in compression quality scales. The methods of information compression are classified into two categories namely: Lossy and Lossless Compression. Cosine Transform would be used for the compression of the information on images with a lossy compression.

The Cosine Transform works by separating information on image into differing frequencies during a step called quantization where part of compression actually occurred. However, the less important frequencies are discarded, hence the term lossy is achieved. According to Kumar et al., [1] described information on compressing image using Discrete Cosine Transform with lossy compression techniques and data loss did not affect the image clarity. The most important frequencies are only used to retrieve the information on the image in the decompression process. As a result, reconstructed information on the image contain some distortion, however, the level of distortion is adjustable [2]. Cosine Transform (CT) offers a new paradigm for information compression on image such as transmission of large images over the internet, maximum usage of resources and transmission of images over the internet in lesser time. Compression ratios changes significantly among compressed images but the use of fixed compression parameter may lead to statistical reduction in detecting an average compression ration.

## 2. RELATED WORKS

Some compression techniques have been developed to compress images without degrading the content which are based on Cosine Transform with binary form most especially 8 x 8 matrices [3]. In [4], compression analysis on binary form was used to compute the Discrete Cosine Transform (DCT). This algorithm was comprehended using Matlab code and modified to perform better when implemented in hardware description language. However using wavelength technique can also examine a set of wavelet function for the implementation of image compression but the image compression information content was assessed and compare with cosine transform [5]. It shows that it provide a good reference for application. Furthermore, Fractal Image Compression (FIC) [6] and Partitioned Iterated Function Systems are widely used techniques prior to the advent of Cosine Transform for real world image compression due to the inherent fractal nature of many natural images [7].

Fractal encoding is a mathematical process to encode any given image as a set of mathematical data that describes the properties of the image. Fractal encoding relies on the fact that all objects contain information in the form of similar, repeating patterns called attractor. The encoding process of an image is given by approximating the smaller range blocks, from the larger domain blocks until an affine contractive map is found. The number of transformations required for approximate in any given image depends on image complexity. The encoding process has intense computations, since large number of iterations is required to find the fractal patterns in the image. Fractal Compression still provides a high degree of compression.

Actually, compressed image size does not generally increase linearly with increase in image size; it increases at a lower rate. Compression is non symmetrical that is the time taken for encoding process is more than the decoding. The decoding process is much simpler as it interprets the fractal codes into the image. Decompression takes significantly less time than compression [8]. With Cosine Transform [9] that widely used lossy form image compression is centred on the Discrete Cosine Transform (DCT) and is reversible. The DCT works by separating images into parts of differing frequencies and the energy compartment can be represented within a relatively narrowed range frequency.

### 2.1 The Encoding Cosine Transform

The 2-D CT [10] for input data size MxN is defined as follows: Given a time domain data sequence  $f(i,j)$ , where  $0 \leq i \leq M-1$ ,  $0 \leq j \leq N-1$ , the transformation of the sequence into a frequency domain data sequence  $F(u,v)$ , where  $0 \leq u \leq M-1$ ,  $0 \leq v \leq N-1$  is defined by the function in the equation for DCT as:

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

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

Where:

M = Input row data set,

N = Input column data set.

x = Row Index time domain where  $0 \leq x \leq M-1$ ,

y = Column index time domain where  $0 \leq y \leq N-1$ .

i = Frequency row index

j = Frequency column index

D(i,j) = Domain frequency coefficient

Decoding the compressed image is done by using Inverse Cosine Transform (ICT) which is given as

$$F(i, j) = \frac{2}{\sqrt{MN}} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} C(u)C(v)F(u, v) \cos\left[\frac{(2m+1)u\pi}{2M}\right] \cos\left[\frac{(2\pi+1)v\pi}{2N}\right] \quad (3)$$

$$D(i, j) = \frac{1}{4} C(i)C(j) \sum_{x=0}^7 \sum_{y=0}^7 P(x, y) \cos\frac{2x+1)i\pi}{20} \cos\frac{(2y+1)j\pi}{20} \quad (4)$$

In equations 1 and 2 define nxn CT and considered M and N are considered to be the same as 256 (8x8). In image compression the input data set is usually partitioned into basic square blocks of data.

### 3. PROPOSED METHODOLOGY

The proposed method would still use the same process as in the literature to calculates the entry ( $i, j^{\text{th}}$ ) of the transformed image from the pixel values of the original image matrix. For the study, a 10 x10 matrix block would still be used for N equals 10 where r and y range from 0 to 9, therefore D (i,j) would be as in Equation 3. The methodology includes the following processes: Implementing the Cosine Transform equation by breaking the image into 10 x 10 blocks of pixels. A Cosine Discrete is applied to each block of pixels. However, quantization process is applied to compress each block. Then the array of the compressed blocks constitute the image is stored in a reduced amount of memory space. Decoding the compression process, the image is reconstructed through decompression, a process that uses the Inverse Cosine Transform (ICT).

$$\text{original} = \begin{bmatrix} 554 & 523 & 523 & 523 & 509 & 509 & 509 & 531 & 600 & 524 \\ 592 & 580 & 536 & 554 & 507 & 527 & 570 & 521 & 500 & 526 \\ 654 & 598 & 554 & 580 & 514 & 541 & 560 & 511 & 499 & 528 \\ 639 & 580 & 536 & 554 & 514 & 518 & 550 & 541 & 573 & 540 \\ 580 & 554 & 536 & 567 & 520 & 522 & 540 & 515 & 523 & 532 \\ 528 & 536 & 523 & 536 & 519 & 521 & 530 & 521 & 554 & 534 \\ 523 & 505 & 510 & 549 & 513 & 515 & 520 & 574 & 521 & 536 \\ 510 & 512 & 523 & 523 & 501 & 519 & 510 & 551 & 531 & 538 \\ 523 & 591 & 554 & 566 & 504 & 524 & 510 & 541 & 525 & 540 \\ 534 & 521 & 539 & 554 & 521 & 573 & 541 & 518 & 527 & 522 \end{bmatrix}$$

Cosine Transform is designed to work on pixel values ranging from -128 to 127 for 8 x 8 image compression, the original block are "leveled off" by subtracting 512 from each entry. This results in the following matrix M

$$M = \begin{bmatrix} 42 & 11 & 11 & 11 & .3 & -3 & -3 & 19 & 88 & 12 \\ 80 & 68 & 24 & 42 & -5 & .15 & 58 & 9 & -12 & 14 \\ 142 & 86 & 42 & 68 & 2 & 29 & 48 & -1 & -13 & 16 \\ 127 & 68 & 24 & 42 & 5 & 6 & 38 & 29 & 61 & 18 \\ 68 & 42 & 24 & 55 & 9 & 10 & 28 & 3 & 11 & 20 \\ 16 & 24 & 11 & 24 & 7 & 9 & 18 & 9 & 42 & 22 \\ 11 & -7 & -2 & 37 & 1 & 3 & 8 & 62 & 9 & 24 \\ -2 & 0 & 11 & 11 & 11 & 7 & -2 & 39 & 19 & 26 \\ 11 & 79 & 42 & 54 & -8 & 12 & -2 & 29 & 13 & 28 \\ 21 & 9 & 27 & 42 & 9 & 61 & 29 & 6 & 15 & 10 \end{bmatrix}$$

The Cosine transform is resulted in matrix multiplication in equation 5

$$D = TMT^T \quad (5)$$

The equation 4 matrix M is multiplied on the left by the CT matrix T and this transforms the rows. However, the columns are then transformed by multiplying on the right by the transpose of the CT matrix. This yields the following matrix result. The block matrix now consists of 100 cosine transform coefficients,  $D_{i,j}$ , where i and j range from 0 to 9.

$$D_{ij} = \begin{bmatrix} 249.9 & 72.7 & 74.8 & 10.3 & -54.5 & 19.3 & 44.9 & 21.2 & -42.7 & -42.1 \\ 31.3 & 66.9 & 57.0 & 54.0 & 19.8 & 16.8 & -2.7 & 16.5 & -26.9 & 36.5 \\ -6.1 & -11 & -12.8 & -13.8 & -7.3 & -20.4 & -18.8 & 16.3 & 1.9 & -7.3 \\ -56.0 & -43.1 & -16.6 & -4.3 & -33.8 & 11.4 & -23.3 & -15.5 & 24.9 & 4.1 \\ -26.8 & -48.9 & -23.1 & -20.7 & 14.6 & 16.1 & -30.9 & 25.8 & -20.3 & -14.4 \\ 16.2 & -32.6 & 22.5 & 20.8 & -29.7 & 3.3 & -6.3 & 2.1 & -25.5 & 31.8 \\ -1.3 & -42.1 & 5.1 & -15.9 & 6.8 & 48.5 & -7.1 & 26.8 & 16.8 & 9.1 \\ 19.2 & 10.0 & 27.7 & 15.9 & 10.2 & 7.3 & -27.7 & 3.7 & -12.6 & 2.1 \\ 8.6 & -10.6 & -18.2 & -8.3 & -5.7 & 20.2 & 12.5 & 10.6 & 18.0 & -18.3 \\ 6.9 & 29.3 & -9.9 & -11.8 & 0.8 & -0.4 & 28.8 & -13.1 & 7.7 & -20.4 \end{bmatrix}$$

### 3.1 Quantization Process

The resulted 10x10 block of CT coefficients is ready for compression by quantization process. This would enable the to decide on the quality levels ranging from 1 to 100, where 1 gives the poorest image quality and highest compression, while 100 gives the best quality and lowest compression. As a result, the quality/compression ratio can be tailored to suit different needs. Human visualization are used for subjected which resulted into JPEG standard quantization matrix. With a 50 quality level, the matrix renders high quality compression.

$$Q_{50} = \begin{bmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 & 71 & 81 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 65 & 50 & 45 \\ 14 & 13 & 26 & 24 & 40 & 57 & 69 & 56 & 43 & 30 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 & 64 & 32 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 & 70 & 63 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 & 81 & 76 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 107 & 100 & 86 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 & 105 & 94 \\ 80 & 104 & 108 & 177 & 124 & 131 & 128 & 121 & 100 & 82 \\ 84 & 122 & 131 & 146 & 137 & 137 & 140 & 138 & 95 & 86 \end{bmatrix}$$

A scalar multiple of JPEG standard quantization matrix can be used for If however, another level of quality and compression is desired, scalar multiples of the JPEG standard quantization matrix may be used for quality level greater than 50 but with less compression, higher image quality. For standard quantization matrix to be achieved, it is multiplied by 100-quality level divide by 50 and for less than 50 quality and more compression, lower image quality, the standard quantization matrix is multiplied by 50/quality level.

The scaled quantization matrix is then rounded and clipped to have positive integer values ranging from 1 to 255. Quantization is then achieved by dividing each element in the transformed image matrix D by the corresponding element in the quantization matrix, and then rounding to the nearest integer value. In this study, quantization matrix level 50 i.e.  $Q_{50}$  is used.

$$C_{ij} = \text{round} \left[ \frac{D_{ij}}{Q_{ij}} \right]$$

$$C = \begin{bmatrix} 16 & 7 & 8 & 1 & -2 & 1 & 1 & 0 & 1 & -1 \\ 3 & 6 & 4 & 5 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -4 & -3 & -1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ -2 & -2 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Note that the coefficient situated near the upper-left corner correspond to the lower frequencies to which the human eye is most sensitive of the image block while the zeros represent the less important, higher frequencies that have been discarded, giving rise to the lossy part of compression. As stated earlier, only the remaining nonzero coefficients will be used to reconstruct the image.

### 3.2 Decompression

Reconstruction of image is by decoding the bit stream representing the quantized matrix C. each element of C is then multiplied by the corresponding element of the quantization matrix originally used.

$$R_{i,j} = Q_{i,j} \times C_{i,j}$$

$$R_{ij} = \begin{bmatrix} 339 & 210 & 228 & 395 & 715 & 1328 & 1627 & 1721 & 1740 & 1751 \\ 348 & 386 & 456 & 592 & 857 & 1377 & 1432 & 1322 & 1170 & 942 \\ -28 & -30 & -38 & -53 & -91 & -144 & -149 & -118 & -107 & -62 \\ -132 & -115 & -135 & -201 & -282 & -500 & -556 & -572 & -542 & -552 \\ -70 & -59 & -64 & -94 & -140 & -253 & -291 & -308 & -285 & -282 \\ 4 & -1 & -4 & 3 & -2 & -18 & -19 & -4 & 21 & 36 \\ -12 & -12 & -14 & -19 & -26 & -58 & -60 & -65 & -50 & -45 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

The ICT is applied to matrix R, which is rounded to the nearest integer. However, 512 is added to each element of the result, giving the decompressed JPEG version N of the original 10x10 image block M.

$$N = \text{roundup}(T^T R T) + 512$$

$$N = \begin{bmatrix} 1749 & -130 & 417 & 478 & 528 & 482 & 508 & 548 & 506 & 503 \\ 2705 & -482 & 338 & 456 & 857 & 1377 & 1432 & 1322 & 511 & 478 \\ 3176 & -907 & 325 & 439 & 541 & 436 & 525 & 586 & 518 & 460 \\ 2626 & 679 & 415 & 449 & 549 & 455 & 530 & 568 & 513 & 476 \\ 1354 & -91 & 560 & 494 & 546 & 497 & 530 & 501 & 510 & 508 \\ 202 & 455 & 667 & 551 & 530 & 532 & 531 & 498 & 514 & 535 \\ -226 & 392 & 804 & 560 & 530 & 548 & 540 & 479 & 1204 & 535 \\ 512 & 512 & 512 & 512 & 512 & 512 & 512 & 512 & 512 & 512 \\ 512 & 512 & 512 & 512 & 512 & 512 & 512 & 512 & 512 & 512 \\ 512 & 512 & 512 & 512 & 512 & 512 & 512 & 512 & 512 & 512 \end{bmatrix}$$

$$original = \begin{bmatrix} 554 & 523 & 523 & 523 & 509 & 509 & 509 & 531 & 600 & 524 \\ 592 & 580 & 536 & 554 & 507 & 527 & 570 & 521 & 500 & 526 \\ 654 & 598 & 554 & 580 & 514 & 541 & 560 & 511 & 499 & 528 \\ 639 & 580 & 536 & 554 & 514 & 518 & 550 & 541 & 573 & 540 \\ 580 & 554 & 536 & 567 & 520 & 522 & 540 & 515 & 523 & 532 \\ 528 & 536 & 523 & 536 & 519 & 521 & 530 & 521 & 554 & 534 \\ 523 & 505 & 510 & 549 & 513 & 515 & 520 & 574 & 521 & 536 \\ 510 & 512 & 523 & 523 & 501 & 519 & 510 & 551 & 531 & 538 \\ 523 & 591 & 554 & 566 & 504 & 524 & 510 & 541 & 525 & 540 \\ 534 & 521 & 539 & 554 & 521 & 573 & 541 & 518 & 527 & 522 \end{bmatrix}$$

$$Depression = \begin{bmatrix} 1749 & -130 & 417 & 478 & 528 & 482 & 508 & 548 & 506 & 503 \\ 2705 & -482 & 338 & 456 & 857 & 1377 & 1432 & 1322 & 511 & 478 \\ 3176 & -907 & 325 & 439 & 541 & 436 & 525 & 586 & 518 & 460 \\ 2626 & 679 & 415 & 449 & 549 & 455 & 530 & 568 & 513 & 476 \\ 1354 & -91 & 560 & 494 & 546 & 497 & 530 & 501 & 510 & 508 \\ 202 & 455 & 667 & 551 & 530 & 532 & 531 & 498 & 514 & 535 \\ -226 & 392 & 804 & 560 & 530 & 548 & 540 & 479 & 1204 & 535 \\ 512 & 512 & 512 & 512 & 512 & 512 & 512 & 512 & 512 & 512 \\ 512 & 512 & 512 & 512 & 512 & 512 & 512 & 512 & 512 & 512 \\ 512 & 512 & 512 & 512 & 512 & 512 & 512 & 512 & 512 & 512 \end{bmatrix}$$

#### 4. RECOMMEDATION

The result shows that compressing image in binary form show a remarkable 70% of the CT coefficients were discarded prior to image block decompression/reconstruction. While the decompressed matrix has higher coefficient than the original matrix, this implies that CT coefficients were not discarded. Furthermore, in the decompressed matrix negative values gives a bad image if used to compress an image.

#### 5. CONCLUSION

Compressing Image with information content using Cosine Transform (CT) was implement using Scikit Learn tool (SKLEARN) and has gone a long way because it is becoming more refined and increasingly high, It has been noticed that a CT of higher matrix is better for compression but has to be run in binaries in other to have a better quality compression that is the matrix be in binary form  $2^2$ ,  $2^3$  or  $2^4$  etc.

## REFERENCES

- [1] Nageswara, R. T., Srinivasa, K. D.. 2008. Image Compression Using Discrete Cosine Transform. Georgian Electronic Scientific Journal: Computer Science and Telecommunications 3:
- [2] Jacquin A. E., 1992. Image coding based on a fractal theory of iterated contractive image transformation, IEEE Transaction on Image Processing, Jan, 1.1:18-30.
- [3] Bouguezel, S., Ahmad, M.O. and Swamy M.N.S., 2009. A fast 8×8 transform for image compression," IEEE International Conference on Microelectronics (ICM'2009), Marrakech, Morocco, Dec..74-77.
- [4] Saroya, N. and Kaur P. 2014 Analysis of image compression algorithm Using DCT and DWT Transforms, International Journal of Advanced Research in Computer Science and Software Engineering Febtruary, 4.2:897:900.
- [5] Sonja Grigic, Mislav Grigic and Branko Zovko-Cihlar, 2001 . Performance Analysis of Image Compression using Wavelets ., IEEE Transactions on Industrial electronics , June 48.. 3:.682- 695.
- [6] Saupe, D. and Jacob, S. 1997. Variance based Quad-trees in fractal Image compression," Electronic Letters , 33..1: 46-48.
- [7] Micheal. F. B. 1996. Fractal Image Compression", Notices of the AMS, June 57-662.
- [8] Mitra, S.K. Murthy, C. A. and Kundu, M. K. 2000 Partitioned Iterated Function System: A New tool for digital imaging", IETE Journal of Research, Sep-Oct 16. 5: 279-298.
- [9] 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. 7. 7.
- [10] Virender, P. Priyanka, , Munish, K. 2012. Analysis of Image Compression Techniques using DCT International Journal of Engineering, Science and Metallurgy Sept 2..3: 790-793