International Journal of Computer Applications (0975 – 8887)
Volume 175– No.19, September 2020

Cosine Transformation of Image Coordinates for Reduction of Image Loss due to Compression and Decompression

Md. Ashek-Al-Aziz
Associate Professor
University of Development Alternative
Dhaka, Bangladesh

Abdullah-Hil Muntakim
Assistant Professor
University of Development Alternative
Dhaka, Bangladesh

ABSTRACT
Discrete Cosine Transformation (DCT) is a popular image compression technique, but it is lossy compression because of image losses found after decompression. In order to reduce the loss, a new algorithm with model of Cosine Transformation of Image Coordinates (CTIC) has been proposed. The earlier method with Matlab standard built in function and proposed model – both have been implemented over 100 Google images. It has been found that CTIC method has much less loss than in the existing DCT method's Matlab built in functions dct2(.) and idct2(.). Also PSNR and Energy Compaction Ratio have been calculated for both the image transformation techniques.

General Terms
Image Analysis, Image Processing, Image Compression and Decompression, Multimedia Communications

Keywords
Image, Compression, Decompression, Discrete Cosine Transformation, Inverse DCT, CTIC, ICTIC

1. INTRODUCTION
Data are to be compressed before transmission through medium of communication and decompressed followed by arriving at the receiving end due to better utilization of bandwidth. Like text, audio and video data, images are also to be processed for compression before transmitting over the channel and decompressed by the receiver to get the original one. Discrete Cosine Transformation (DCT) [1-2] is one of the popular method for compression and decompression. Forward DCT is applied to image at the time of compression at the sender end and Inverse DCT is at the time of decompression at the receiver end. Mathworks has implemented forward and inverse DCT both in their Matlab functions – dct2(.) and idct2(.) respectively. Both these functions receives image as parameter and returns compressed and decompressed image respectively. A gray scale image is nothing but a two dimensional matrix. At the time of compression, dct2(.) function receives a two dimensional image matrix, apply DCT over the gray image and produce a compressed image which is also an encrypted image. Similarly, idct2(.) function can have the compressed and encrypted image as input substituting within the parenthesis as parameter and produce a decompressed and decrypted image which is supposed to be the exact copy of the original one. Due to lossy compression technique of DCT [3], it retrieved image shall not be the same of the original one rather it should be carbon copy. However, we get such result of DCT implementation in Matlab practice.

A color image can be converted into gray image using rgb2gray(.) function in Matlab. Therefore, the result of idct2(.) function after execution of dct2(.) over 'Lenna.png' image, it has be found that there has been 100% loss after decompression which is not expected. Moreover, we have got 81.44% loss in case of 'cameraman.jpg' image which is shown in Fig 1 (a) and (b).

The aim of the research work is to reduce the loss after decompression. To do so, we are to propose new method of Cosine Transformation of Image Co-ordinates (CTIC), implement using Matlab, and compare the results of proposed method outcome with existing built in function results.

Fig 1(a): Original cameraman.jpg image

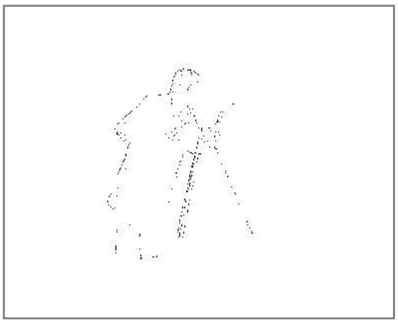

Fig 1(b): Resulting image of idct2(.) function execution after dct2(.)

2. RELATED WORKS
Alotaibi and Elrefaei (2019) implemented DCT with IWT (Integer Wavelet Transform) methods together to increase the robustness and imperceptibility in text-image watermarking [4] but not image compression. DCT is used as image encryption technique as information security tool by Shaheen et. al. (2018) but it failed to outperform DWT (Discrete Wavelet Transform) method [5]. A hybrid method consists of

DWT and DCT has been applied to compress 3D image by Elharar et. al. (2007) [6]. DCT was proved to be used in pattern recognition with Wiener filtering method and its performance was very close to Karhunen-Loeve-Transform (KLT) [7]. Fracastoro et. al. (2018) proposed a method of Steerable DCT (SDCT) that included the directional information during transformation which outperformed conventional DCT [8]. Zeng, and Fu (2008) introduced a new framework - first transform consists of following a direction other than the vertical or horizontal, then produced coefficients are arranged in a way so that the second transform is applied to the coefficients which are best aligned with each other [9]. In order to reduce the noise on the image, DCT based filtering method has been applied which showed better results than state of the art filters [10]. Wu and Tai (2001) regarded the DCT as band pass filter – small size image blocks were considered equal size bands and at the time of decompression, those bands were gathered and similarity property was used reduce the bit rate [11]. Byeong G. Lee (1984) proposed a new algorithm named 2^m-point DCT reducing the number of multiplications in conventional DCT which was practically usable for acoustic signals but not for image compressions [12]. 2^m3^n point DCT was realized by mix-radix algorithm for which radix-3 and radix-6 algorithm was found better than radix-2 [13]. Aburdene et. al. (1995) derived recursive algorithm for computation of DCT and Inverse DCT using Clenshaw's recurrence formula [14] which is useful for signal communication with less computation. A fast recursive algorithm for DCT was shown earlier by Hou (1987) [15]. Gilbert Strang (1999) established DCT-1 through DCT-4[16]. DCT-1 and DCT-2 was proposed by Wang and Hunt (1983) [17]. Fractional DCT (FDCT) got attention of Cariolaro et. al (2002) instead of DCT for image compression [18]. A simple convolution-multiplication relationship for DCT filtering was established by Chitprasert and Rao (1990) [19]. Obukhov and Kharlamov (2008) implemented DCT8×8 block CUDA programming for efficient execution [20]. Hafed and Levine (2001) showed application of face recognition using DCT [21]. Bhandari et. al. (2011) used DCT with Singular Value Decomposition (SVD) to enhance the low contrast satellite image [22]. Similar approach used by Kumar et. al. (2012) for same application area implemented normalized difference vegetation index method [23]. Narasimha and Peterson (1978) compared performances of N-point DCT and 2N-point DFT (Discrete Fourier Transformation) and found that N-point DCT is faster [24]. A high fidelity has been achieved by Zhao an Yuan (1994) using fractals and DCT in image compression [25]. Robinson and Kecman (2003) incorporated Support Vector Machine (SVM) with DCT to compress images and produced better image quality after decompression [26]. A fast pruning algorithm was developed by Zhongde Wan (1991) [27]. Projection on Convex Set (POCS) and Constrained Least Square (CLS) methods outperformed Block DCT in the work of Yang et. al. (1993) [28]. Discrete Sine Transform was demonstrated over DCT by Wang (1982) [29]. Mandyam et. al. (1997) made entropy reduction in DCT based system of image compression and decompression than in the other JPEG compression methods [30]. Though significant improvement is shown, entropy is not far reduced that ensures zero loss in image compression and decompression.

3. METHOD

3.1 Existing Method

Forward DCT: P(i,j) is a representation of 2D matrix of an image. This matrix is divided by some 8×8 blocks. Transformation is carried out using following mathematical equation to get the transformed image matrix F(i,j).

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

where, $C(i),C(j) = \frac{1}{\sqrt{2}}$ for i,j=0 and $C(i),C(j) = 1$ for all other values of i and j [31].

Inverse DCT: P(i,j) is a representation of 2D matrix of an image. This matrix is divided by some 8×8 blocks. Transformation is carried out using following mathematical equation to get the transformed image matrix F(i,j).

$$F(i,j) = \frac{1}{4}\sum_{i=0}^{7}\sum_{j=0}^{7}C(i)C(j)F(i,j)\cos\frac{(2x+1)i\pi}{16}\cos\frac{(2y+1)j\pi}{16}$$

where, $C(i),C(j) = \frac{1}{\sqrt{2}}$ for i,j=0 and $C(i),C(j) = 1$ for all other values of i and j [31].

All 8×8=64 values of input block contribute to the values of output block and since cos0=1, the value of upper left most corner of transformed matrix is just average of the whole block. This is called DC coefficient and rest others are AC coefficient [31].

3.2 Proposed Method

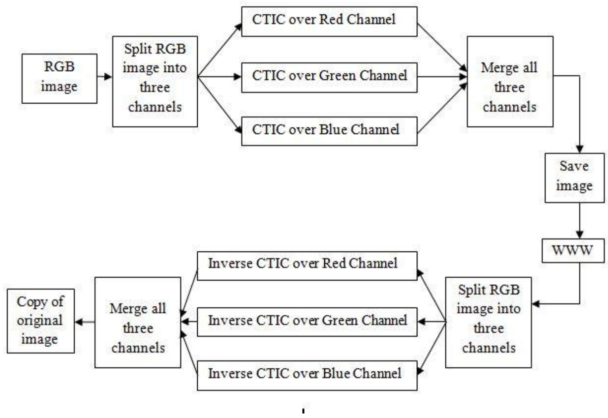

Fig 2: Proposed CTIC model

A model is proposed to depict the roll of application of proposed method. In this model, the image which is to transmitted over communication channel should splitted into three channel Red, Green and Blue first, then the proposed algorithm to be executed over each channel separately. After that transformed channels can be merged to form a single file and then go through communication medium. At the receiving end the received image is to be split again and inverse transformation is applied on each channel. After inverse transformation, three different channels are again merged together to form the original image.

3.3 Proposed CTIC Algorithm

3.3.1 For Compression

```
[m n] ← sizeof(original image)
Split the color channels, apply following loops each
individual channel
For i=1 to m do
        For j=1 to n do
                P(i,j) ← P(i,j)/100
        End For
End For
For i=1 to m do
        For j=1 to n do
                F(i,j) ← P(i,j)*cos(2*π*i)*cos(2*π*j)
        End For
End For
```

3.3.2 For decompression

```
[m n] ← sizeof(compressed image)
Split the color channels, apply following loops each
individual channel
For i=1 to m do
        For j=1 to n do
                P(i,j) ←
F(i,j)/(cos(2*π*i)*cos(2*π*j))
        End For
End For
For i=1 to m do
        For j=1 to n do
                P(i,j) ← P(i,j)*100
        End For
End For
```

4. TOOLS AND DATASET

For experimental setup a computer machine with Intel Pentium P 6200, 2.13 Gigahertz Microprocessor and 2 Gigabyte DDR 3 RAM has been taken. Matlab Version 7.12.0.635 (R2011a) is used for code implementation. 100 images from Google's image repository have been used as dataset for experiment which didn't need any permission. For similarity measure, the online portal www.imgonline.com.ua/eng/similarity-percent.php was used. For calculation of Pick to Signal Noise Ratio (PSNR) and Energy Compaction Ratio (ECR), 'Octave Online' package was used since Matlab Version 7.12.0.635 (R2011a) doesn't have 'psnr' command. Energy Compaction Ratio is nothing but the ratio of arithmetic mean and geometric mean of the image pixels.

5. RESULTS

Matlab built in function dct2(.) and idct2(.) have been encoded in the program written for DCT implementation as well as matlab code for proposed CTIC model has been developed, fed 100 images as inputs, produce resulting images and saved into disk. Then the similarity between the original image and resulting image of idct2(.) function has been checked, also checked the similarity between the original images and resulting images of inverse CTIC algorithm code, subtract from 100 to convert the similarity measure into dissimilarity or loss. The loss for both the function cases has been compared and a comparative chart as shown in Fig 4 has been produced. Few of the best results are shown in Fig 3. Fig 5 shows the comparative result of PSNR. Fig 6, Fig 7 and Fig 8 show the comparative results of Energy Compaction Ratio of both DCT and CTIC transforms over the red channels, green channels and blue channels of 100 images respectively.

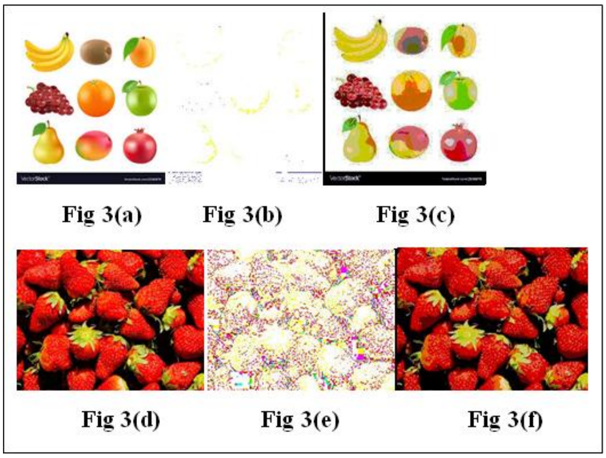

Fig 3: Few of the best results. (a) and (d) are the original images. (b) and (e) are the result of inverse transformation by DCT. (c) and (f) are the result of inverse transformation of CTIC

6. CONCLUSION AND RECOMMENDATION

Proposed model is versatile because it will work upon all type of color/rgb image file. Moreover, we have successfully implemented proposed CTIC method using Matlab and found that proposed method has reduced image loss after decompression – it has reduced image loss from 69.12% to 7.31% on an average. This is a significant improvement. We have also checked the Pick to Noise Signal Ratio (PSNR) and we have got much higher PSNR in CTIC than in DCT as shown in Fig 5. In case of Energy Compaction Ratio, CTIC has got higher ratio in red channels, green channels and blue channels of the color images in 63%, 65% and 52% cases respectively. Our next mission is to make the image loss 0%.

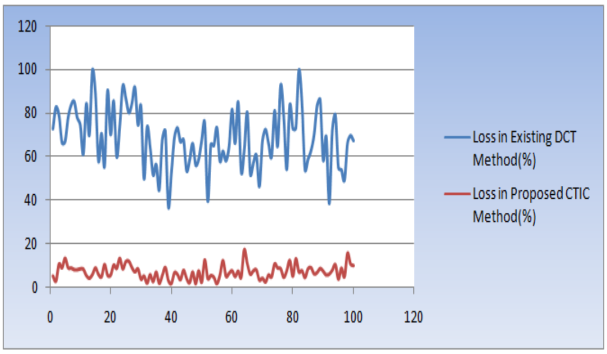

Fig 4: Comparative results of existing DCT method/dct2(), idct2() function and proposed CTIC method

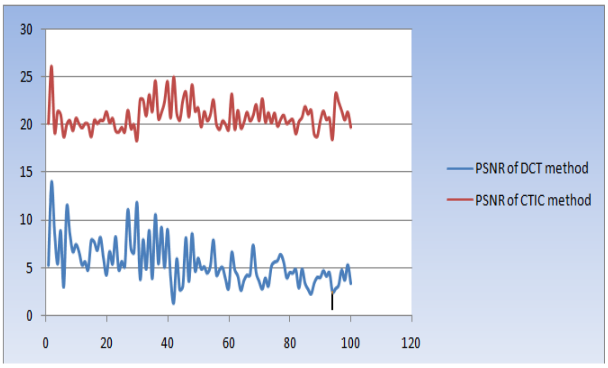

Fig 5: Comparative results of existing DCT method and proposed CTIC method for PSNR

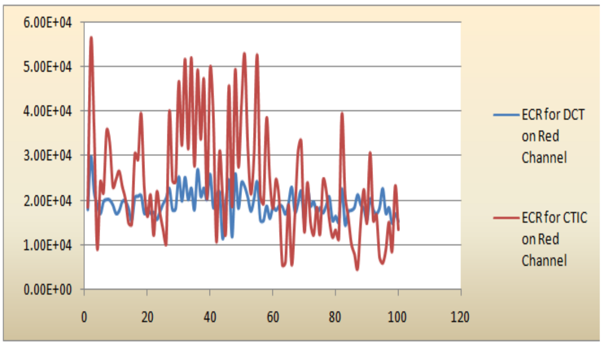

Fig 6: Comparative results of Energy Compaction Ratio for DCT and CTIC transform on Red Channels of the images

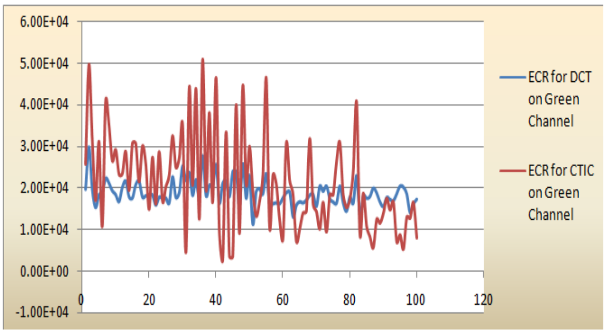

Fig 7: Comparative results of Energy Compaction Ratio for DCT and CTIC transform on Green Channels of the images

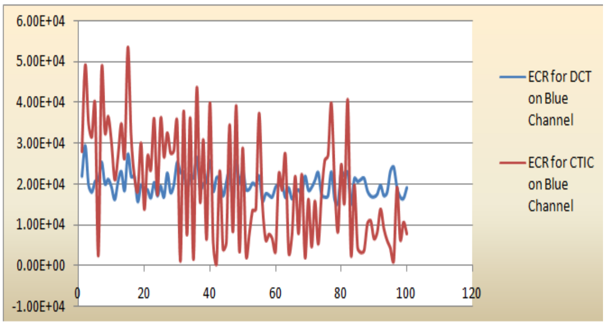

Fig 8: Comparative results of Energy Compaction Ratio for DCT and CTIC transform on Blue Channels of the images

7. REFERENCES

[1] Thota N. R. and Devireddy S. K., Image Compression Using Discrete Cosine Transform, Georgian Electronic Scientific Journal: Computer Science and Telecommunications, 3(17), 2008

[2] Watson A. B., Image Compression Using the Discrete Cosine Transform, Mathematica Journal, 4(1), 1994, 81-88

[3] Raid A.M., Khedr W.M., El-dosuky M. A. and Ahmed W., Jpeg Image Compression Using Discrete Cosine Transform - A Survey, International Journal of Computer Science & Engineering Survey (IJCSES), 5(2), April, 2014

[4] Alotaibi R. A., Elrefaei L. A., Text-image watermarking based on integer wavelet transform (IWT) and discrete cosine transform (DCT), Applied Computing and Informatics, 15, 2019, 191–202

[5] Shaheen A. M., Sheltami T. R., Al‑Kharoubi T. M., Shakshuki E., Digital image encryption techniques for wireless sensor networks using image transformation methods: DCT and DWT, Journal of Ambient Intelligence and Humanized Computing, Springer-Verlag GmbH Germany, 2018

[6] Elharar E., Stern A., Hadar O. and Javidi B., A Hybrid Compression Method for Integral Images Using Discrete Wavelet Transform and Discrete Cosine Transform, Journal Of Display Technology, 3(3), 2007

[7] Ahmed N., Natarajan T., and Rao K. R., Discrete Cosine Transform, IEEE Transactions on Computers, January, 1974

[8] Fracastoro G., Fosson S. M., Magli E., Steerable Discrete Cosine Transform, IEEE Transactions on Image Processing, 26(1), October 24, 2018

[9] Zeng B. and Fu J., Directional Discrete Cosine Transforms—A New Framework For Image Coding, IEEE Transactions On Circuits And Systems For Video Technology, 18(3), March, 2008

[10] Pogrebnyak O. and Lukin V. V.**,** Wiener discrete cosine transform-based image filtering, Journal of Electronic Imaging 21(4), Oct-Dec, 2012

[11] Wu Y. G. and Tai S. C., Medical Image Compression by Discrete Cosine Transform Spectral Similarity Strategy, IEEE Transactions On Information Technology In Biomedicine, 5(3), September, 2001

[12] Lee B. G., A New Algorithm to Compute the Discrete Cosine Transform, IEEE Transactions On Acoustics, Speech, And Signal Processing, 32(6), December, 1984

[13] Chan Y. H. and Siu W. C., Mixed-Radix Discrete Cosine Transform, IEEE Transactions On Signal Processing, 41(II), November, 1993

[14] Aburdene M. F., Zheng J. and Kozick R. J., Computation of Discrete Cosine Transform Using Clenshaw's Recurrence Formula, IEEE Signa L Processing Letters, 2(8), August, 1995

[15] Hou H. S., A fast recursive algorithm for computing the discrete cosine transform, IEEE Transactions On Acoustics, Speech, And Signal Processing, 35(10), October, 1987

[16] Strang G., The Discrete Cosine Transform, SIAM Review, 41(1), Society for Industrial and Applied Mathematics, 1999, 135–147

[17] Wang Z. and Hunt B. R., The Discrete Cosine Transform—-A New Version, International Conference on Acoustics, Speech, and Signal Processing (ICASSP), IEEE, 1983

[18] Cariolaro G., Erseghe T. and Kraniauskas P., The Fractional Discrete Cosine Transform, IEEE Transactions On Signal Processing, 50(4), April, 2002

[19] Chitprasert B. and Rao K. R., Discrete Cosine Transform Filtering, Signal Processing, 19, 1990, Elsevier, 233-245,

[20] Obukhov and Kharlamov, Discrete Cosine Transform for 8x8 Blocks with CUDA, NVIDIA, 2008

[21] Hafed Z. M. and Levine M. D., Face Recognition Using the Discrete Cosine Transform, International Journal of Computer Vision 43(3), 2001, 167–188

[22] Bhandari A. K., Kumar A. and Padhy P. K., Enhancement of Low Contrast Satellite Images using Discrete Cosine Transform and Singular Value Decomposition, International Journal of Computer, Electrical, Automation, Control and Information Engineering, 5(7), 2011

[23] Kumar A., Bhandari A. K., Padhy P., Improved normalised difference vegetation index method based on discrete cosine transform and singular value decomposition for satellite image processing, IET Signal Process., 6(7), 2012, 617–625

[24] Narasimha M. J. and Peterson A. M., On the Computation of the Discrete Cosine Transform, IEEE Transactions On Communications, 26(6), June, 1978

[25] Zhao Y. and Yuan B., Image compression using fractals and discrete cosine transform, Electronics Letters, 30(6), 17th March, 1994, 474-475

[26] Robinson J. and Kecman V., Combining Support Vector Machine Learning with the Discrete Cosine Transform in Image Compression, IEEE Transactions On Neural Networks, 14(4), July, 2003

[27] Wan Z., Pruning the Fast Discrete Cosine Transform, IEEE Transactions on Communications, 39(5), May, 1991

[28] Yang Y., Galatsanos N. P., Katsaggelos A. K., Regularized Reconstruction to Reduce Blocking Artifacts of Block Discrete Cosine Transform Compressed Images, IEEE Transactions On Circuits And Systems For Video Technology, 3(6), December, 1993

[29] Wang Z., A Fast Algorithm for the Discrete Sine Transform Implemented by the Fast Cosine Transform, IEEE Transactions On Acoustics, Speech, And Signal Processing, 30(5), October, 1982

[30] Mandyam G., Ahmed N. and Magotra N., Lossless Image Compression Using the Discrete Cosine Transform, Journal Of Visual Communication And Image Representation, 8(1), March, 1997, 21–26

[31] Halsall F., Multimedia Communications Applications, Networks, Protocols and Standards, Pearson Education, 1988, 116-17.