International Journal of Scientific & Engineering Research Volume 10, Issue 2, February-2019
ISSN 2229-5518

Digital Image Compression using DCT Algorithm: An Improvement

Rajasekar P, Srinivas L N B

Abstract— Image Compression is a method in which the size of a digital image is reduced yet the visibility of the image is maintained. It is done to reduce the cost for storing as well as transmitting the image. There are many kinds of compression techniques for compressing different types and sizes of images. Depending on the user's requirement to get the desired result, compression might be done at a high rate which can even reduce the quality of the image to a great extent. There are many kinds of algorithms present to deal with such problems and which can compress an image to the user's desired compression rate. These algorithms are categorized into two different compression types: lossy and lossless. Here we are going to use DCT Algorithm, which comes under lossy compression, i.e. it does compress the image but compromises with the image quality too and it takes a lot of time to compress an image. Therefore, this gives a lot of scope of doing a research to improve the DCT Algorithm.

Index Terms—DCT, Lossless and lossy compression, Quantization, Normalization, Zero padding, Lookup table, PSNR ratio

1 Introduction

The use of images can be done in a variety of areas and applications such as in defense, medicine, forensics, as well as in the regular day-to-day use by people. However, these images contain pixels and the range of sizes can vary to a greater extent. So, it depends on the requirements of the users, how they want to transform the image. In general, digital image compression is defined as how to lessen or minimize the size of an image, a video, or any kind of graphic file, to an extent where it cannot be accepted. This helps in storage stabilisation, i.e. to store more and more images in a confined storing area by reducing the size of the image. Also, it means to remove or reduce the irrelevant and redundant data so that we can store the image effectively and efficiently.

There are many ways by which image compression can be done, like scalar/vector quantization [1], transform coding, etc. But when it is about low bit rate, transform coding is more efficient than the others. There are two compression types:

1. Lossless Image Compression
2. Lossy Image Compression

- *Srinivas L N B is currently working as Assitant Professor in the Department of Information Technology, School of Computing, SRM Institute of Science and Technology, Tamilnadu, India. E-mail: srinivas.l@ktr.srmuniv.ac.in*
- *Rajasekar P is currently working as Assitant Professor in the department of Information Technology, School of Computing, SRM Institute of Science and Technology, Tamilnadu, India. E-mail: rajasekar.p@ktr.srmuniv.ac.in*

LOSSLESS IMAGE COMPRESSION

Lossless image compression, also known as reversible compression, is the one where data is not degraded. Here the rate of reduction of image data or image quality is a lot lesser compared to the lossy image compression techniques. If the data is lost, the original data can be recovered from the compressed data. Also, there is a minimal loss in data which leads to smaller compression rates. This technique is also called noiseless image compression as there is no addition of noise. This kind of technique can be used in areas where one cannot afford to lose important data. The coding techniques used here are: Area coding, Huffman encoding, Run Length encoding and LZW coding [10].

LOSSY IMAGE COMPRESSION

Lossy image compression, also known as irreversible compression, is the one where there is a lot of compromise done regarding the image quality. This is done by partially discarding the data to represent the image. The results shown by this technique is visible by the naked eye because it does compress the image but also degrades its quality to a great extent. We can reduce a lot of image data using lossy compression, unlike lossless compression. This technique is used mostly where the image quality does not matter that much as what matters is the sense of the image. It should also not be degraded to such an extent that it cannot be compared to the original image at all. The low quality of the image helps to protect it against image theft.

ISSN 2229-5518

The processes used here, like quantization, results in the loss of information and entropy encoding. Then decoding and dequantization is done and finally we can get the reconstructed image. The coding techniques used are Vector quantization, Sub-band coding, Transform coding, Block Truncation coding [6] and Fractal coding.

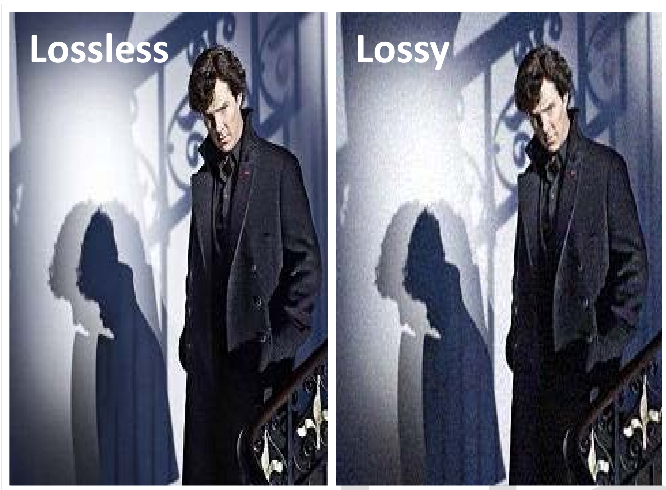

Fig. 1. Comparison between lossless and lossy image compression

Thus, image compression helps users to compress digital images according to their requirements. This means they keep as much information intact as needed and discard the useless or redundant data by applying one of these above-mentioned techniques.

DCT Algorithm is a very user-friendly technique which we are using in our research to compress the digital images. But it is a lossy compression technique which compresses the image by discarding many parts of the image data, thus reducing the image quality to a large extent. Here we are going to improve the process by incorporating a strategy which helps us to compress the image at our own specific rate to produce desired results, i.e. image getting compressed as well as minimal difference in the quality of the original and compressed image. This helps us to maintain a clear analysis of different images of different sizes along with their PSNR ratio as well as compare it with the original DCT algorithm.

DCT EXPERIMENT AND DISCUSSION

DCT or Discrete Cosine Transform is a lossy image compression algorithm which transforms an image into separate parts. In the whole process, the image is first broken into 8x8 blocks of pixels and then DCT is applied to each block. Now each block is compressed through quantization and then it is de-quantized and finally after applying inverse DCT the image is reconstructed as shown in the below image.

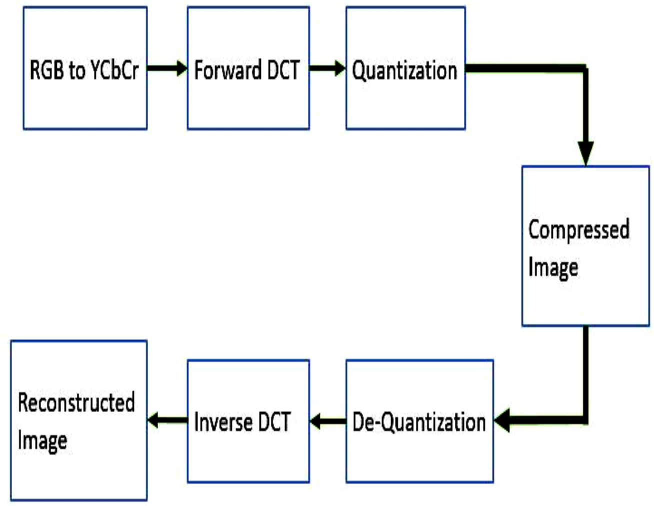

Fig.2. Architecture of the process of image compression

DCT is a Fourier related transform and the only difference is that here it uses real numbers. The image is transformed from a spatial to a frequency domain and the image is represented as sinusoids of different frequencies. In this process, normalization [7] is done after applying the forward DCT [8] or DCT-II. The formula for forward DCT is:

$$X_k = \sum_{n=0}^{N-1} x_n \cos\left[\frac{\pi}{N}\left(n+\frac{1}{2}\right)k\right] \quad k = 0, \dots, N-1$$

Here, N represents real numbers from x_0 to x_{N-1} and these are transformed into N real numbers X_0 to X_{N-1}.

While doing normalization after the forward DCT, the image turns into strands of black and white colours and they hold the pixel values ranging from 0 to 255, where 0 represents pure black and 255 represents pure white colour. As there are a lot of 8x8 blocks of pixels, we are referring to only one. The least frequency values are present on the top left corner of the matrix and the frequency increases from left to right, top to bottom and diagonally, and here only the top left values holds greater importance. Now the image is re-quantized to eleven bits where the data is stored in the form of 2's complement. The data is actually never stored in an eleven bit format and that is why this process creates noise (data loss). After that, inverse DCT is applied using the formula:

$$X_k = \frac{1}{2}x_0 + \sum_{n=1}^{N-1} x_n \cos\left[\frac{\pi}{N}n\left(k+\frac{1}{2}\right)\right] \quad k = 0,\ldots,N-1$$

After that the reconstructed image is formed followed with some loss of data because of this lossy algorithm technique. The process may take time depending on the colours in image.

The following table shows a sample of forward DCT implementation matrix followed with the resultant coefficient values:

TABLE 1
INPUT MATRIX

140	144	147	140	140	155	179	175
144	152	140	147	140	148	167	179
152	155	136	167	163	162	152	172
168	145	156	160	152	155	136	160
162	148	156	148	140	136	147	162
147	167	140	155	155	140	136	162
136	156	123	167	162	144	140	147
148	155	136	155	152	147	147	136

TABLE 2
OUTPUT MATRIX

186	-18	15	-9	23	-9	-14	19
21	-34	26	-9	-11	11	14	7
-10	-24	-2	6	-18	3	-20	-1
-8	-5	14	-15	-8	-3	-3	8
-3	10	8	1	-11	18	18	15
4	-2	-18	8	8	-4	1	-7
9	1	-3	4	-1	-7	-1	-2

As seen in the DCT output matrix [11], it has the highest value of 186 which may be confusing as whether it is of lower frequency or not. But as we move away from this value, the values become lower in magnitude and this can show how only the lower frequency values hold greater importance than the higher frequency values which are represented by the lower right coefficients. Here the coefficients are normalized so that the unimportant data can be discarded, thus getting the result as a compressed image.

RESULT

When a digital image is converted into a spectral image that is when image compression is required and is implemented. During this process, the coloured or colourless image is converted into a grayscale image. After this the image is fragmented into their respective Red, Green and Blue planes. These individual planes are extracted one by one and converted into 8x8 block structure. From this 2-D matrix data, all the 1-D arrays are extracted and forward DCT is applied on every block which produces the coefficient data. The unnecessary coefficients are eliminated and by retaining the required coefficients, the data is normalised. On this, inverse DCT is applied and the 1-D arrays are converted into 2-D arrays of 8x8 matrix structure. These represent the spectral planes and these planes are combined to form a composite image. A process called zero padding [15] is implemented so that each dimension of the lower parts of the image is compressed to form a multiple of 8. This zero padding is cut short from the spectral planes which is provided as a result image. The parts of all images together constitute to form a single image i.e. a composite image which comes before the final reconstructed image is composed.

CONCLUSION

In the improved version of DCT the image is split into blocks of 8x8 matrices. Then a spectral coefficient data of 8x8 blocks is produced by performing forward DCT on the split image of 8x8. It then produces an 8x8 pixel data by performing inverse DCT on the spectral 8x8 blocks. The composites of all images together constitute to form a single image. In earlier version, the forward and inverse DCT both were not applied to the 8x8 blocks. The latest improved version is much faster because we have directly taken the values from the cosine lookup table and it is easy to implement and has compression size improvement as compared to earlier version because in earlier version the image was not broken down into blocks, but rather forward DCT was applied on the entire image.

ISSN 2229-5518

DCT algorithm's basic process is to express a finite sequence of data points in terms of a sum of oscillating cosine functions. Instead of calculating and computing the cosine values, the improved version fetches the cosine value from the incorporated 8x8 cosine table which makes the program run faster and eventually takes less performance time. Process called zero padding is implemented so that each dimension of the right and bottom part of the image is compressed to form a multiple of the number 8. This zero padding is cut short or trimmed from the spectral planes that is provided as a result image.

TABLE 3

COMPARISON TABLE BETWEEN DCT IMPLEMENTATION AND THE IMPROVED METHOD

Images	DCT Implementation (in ms)	Improved Method (in ms)
Sherlock.jpg (37kb)	**46091 (21kb)**	**623 (17kb)**
Flower.jpg (29kb)	**96662 (28kb)**	**672 (21kb)**
Guniea.jpg (15kb)	**16456 (8kb)**	**469 (8kb)**
Puppy.jpg (25kb)	**40427 (15kb)**	**529 (14kb)**
Tiger.jpg (44kb)	**38731 (20kb)**	**517(18kb)**

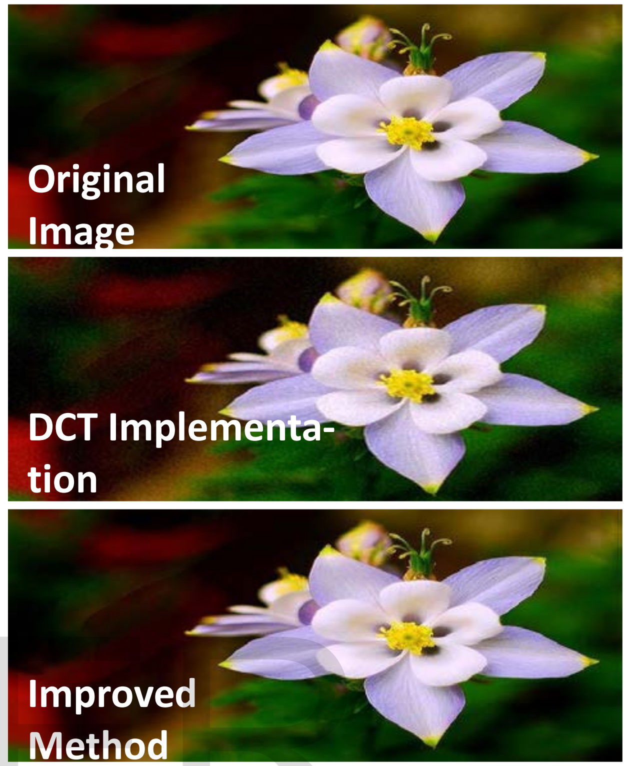

Fig. 3. Comparison between DCT and Improved implementation

REFERENCES

[1] Messaoudi and K. Srairi, "Colour image compression algorithm base on the DCT transform using difference lookup table", in IEEE Vol. 52 No. 20 pp. 1685–1686, 2016

[2] Navpreet Saroya, Prabhpreet Kaur, "Analysis of IMAGE COMPRESSION Algorithm Using DCT and DWT Transforms", in ISSN:2277 128X, Volume 4, Issue 2, IJARCSSE, February 2014

[3] Anurag Bhatt ,Dr. Ashutosh Kumar Bhatt , "Image Compression Algorithms under JPEG with Lapped Orthogonal Transform and Discrete Cosine Transformation", e-ISSN: 2278-067X, p-ISSN: 2278-800X, Volume 7, Issue 3 (May 2013), PP. 06-10, IJERD, 2013

[4] Boucetta, A., and Melkemi, K.E.: 'DWT based-approach for color image compression using genetic algorithm', Image and Signal Processing Springer Berlin Heidelberg, 2012, pp. 476–484

[5] Deshlahra, Archana, G. S. Shirnewar, and A. K. Sahoo. "A Comparative Study of DCT, DWT & Hybrid (DCT-DWT) Transform." (2013).

[6] Delp, E.J., and Mitchell, O.R.: 'Image compression using block truncation coding', IEEE Trans Commun., 1979, 27, pp. 1335–1342

[7] Douak, F., Benzid, R., and Benoudjit, N.: 'Color image compression algorithm based on the DCT transform combined to an adaptive block scanning', AEU-International Journal of Electronics and Communications., 2011, 65, pp. 16–26

[8] Sugreev Kaur and Rajesh Mehra, "HIGH SPEED AND AREA EFFICIENT 2D DWT PROCESSOR BASED IMAGE COMPRESSION Signal & Image Processing : An International Journal(SIPIJ) Vol.1,No.2, December 2010.

[9] Pearlman, W., Islam, A., Nagaraj, N., and Said, 'Efficient, low-complexity image coding with a set partitioning embedded block coder', IEEE Transactions on Circuits and Systems for Video Technology., 2004, 14, pp. 1219–1235

[10] Andrew B. Watson," Image Compression Using the Discrete Cosine Transform", Mathematical Journal, 4(1), 1994, p. 81-88

[11] Anitha s," Image Compression Using Discrete Cosine Transform & Discrete Wavelet Transform", International Journal of Scientific & Engineering Research Volume 2, Issue 8, August-2011 1 ISSN 2229-5518

[12] Salam Benchikh and Michael Corinthios, "A Hybrid Image Compression Technique Based on DWT and DCT Transforms", IEEE, Department of Electrical Engineering, Ecole Poly technique de Montreal, Montreal, Qc, Canada

[13] R. Gonzalez, R. Woods, "Digital image processing," 2nd edition, prentice Hall, New jersey, 2002.

[14] Y. Q. SHi, H. Sun, "Image and video compression for multimedia engineering," 2nd edition, CRC press, Taylor & Francis group, 2008

[15] M. Rabbani, R. Joshi, "An overview of the JPEG 2000 still image compression standard," Elsevier, Signal processing: Image Communication, pp. 3-48, 2002.