УДК 621.3

Ayman M.M. Ahmed
Author contact information: a_ymn2002@yahoo.com

INVESTIGATION OF ASSOCIATIVE PROPERTY OF MATRIX MULTIPLICATION IN DISCRETE COSINE TRANSFORMS FOR FIXED-POINT CALCULATION SYSTEMS

Досліджено, як зміна порядку множення матриць у двовимірному дискретному косинусному перетворенні та обчисленнях з фіксованою комою для стиску зображень впливає на якість відтвореного зображення.

They investigate how matrix multiplication order change in the discrete cosine transform and fixed-point calculation for image compression affects the resolution of the reconstructed image.

Introduction

During applying the discrete cosine transform (DCT) in an image compression system based on fixed-point calculation (for systems that deal with the integers only, while the exponent part of the number is ignored), questions appears here are; "does this property affects the resolution of the reconstructed image or not?" and if so, in what way it affects? And is there a difference in results for floating point calculations? All these questions are answered in this paper.

It is important to investigate such questions; to be considered while designing image compression systems (DCT based) where resolution and image quality is a significant issue; such as remote sensing satellites and high image quality transmission systems DCT based [1,2].

Discrete cosine transform [1] is applied in an image using the equation: [$dctr := dct \cdot img \cdot dct^T$] where:

- DCT: is discrete cosine transforming matrix.

$$dct := \begin{pmatrix} 0.3536 & 0.3536 & 0.3536 & 0.3536 & 0.3536 & 0.3536 & 0.3536 & 0.3536 \\ 0.4904 & 0.4157 & 0.2778 & 0.0975 & -0.0975 & -0.2778 & -0.4157 & -0.4904 \\ 0.4619 & 0.1913 & -0.1913 & -0.4619 & -0.4619 & -0.1913 & 0.1913 & 0.4619 \\ 0.4157 & -0.0975 & -0.4904 & -0.2778 & 0.2778 & 0.4904 & 0.0975 & -0.4157 \\ 0.3536 & -0.3536 & -0.3536 & 0.3536 & 0.3536 & -0.3536 & -0.3536 & 0.3536 \\ 0.2778 & -0.4904 & 0.0975 & 0.4157 & -0.4157 & -0.0975 & 0.4904 & -0.2778 \\ 0.1913 & -0.4619 & 0.4619 & -0.1913 & -0.1913 & 0.4619 & -0.4619 & 0.1913 \\ 0.0975 & -0.2778 & 0.4157 & -0.4904 & 0.4904 & -0.4157 & 0.2778 & -0.0975 \end{pmatrix}$$

Resulted from the formula

The two-dimensional DCT of an M-by-N matrix A is defined as follows.

$$B_{pq} = \alpha_p \alpha_q \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} A_{mn} \cos\frac{\pi(2m+1)p}{2M} \cos\frac{\pi(2n+1)q}{2N}, \quad \begin{matrix} 0 \le p \le M-1 \\ 0 \le q \le N-1 \end{matrix}$$

$$\alpha_p = \begin{cases} 1/\sqrt{M}, & p = 0 \\ \sqrt{2/M}, & 1 \le p \le M-1 \end{cases} \qquad \alpha_q = \begin{cases} 1/\sqrt{N}, & q = 0 \\ \sqrt{2/N}, & 1 \le q \le N-1 \end{cases}$$

- img: is fragment of image 8X8 (Pixels).
- dct^T: Is the transpose of the dct matrix

$$dct^T = \begin{pmatrix} 0.354 & 0.49 & 0.462 & 0.416 & 0.354 & 0.278 & 0.191 & 0.098 \\ 0.354 & 0.416 & 0.191 & -0.098 & -0.354 & -0.49 & -0.462 & -0.278 \\ 0.354 & 0.278 & -0.191 & -0.49 & -0.354 & 0.098 & 0.462 & 0.416 \\ 0.354 & 0.098 & -0.462 & -0.278 & 0.354 & 0.416 & -0.191 & -0.49 \\ 0.354 & -0.098 & -0.462 & 0.278 & 0.354 & -0.416 & -0.191 & 0.49 \\ 0.354 & -0.278 & -0.191 & 0.49 & -0.354 & -0.098 & 0.462 & -0.416 \\ 0.354 & -0.416 & 0.191 & 0.098 & -0.354 & 0.49 & -0.462 & 0.278 \\ 0.354 & -0.49 & 0.462 & -0.416 & 0.354 & -0.278 & 0.191 & -0.098 \end{pmatrix}$$

Resulted from matrix transpose and represent the following formula:

The DCT is an invertible transform, and its inverse is given by

$$A_{mn} = \sum_{p=0}^{M-1} \sum_{q=0}^{N-1} \alpha_p \alpha_q B_{pq} \cos\frac{\pi(2m+1)p}{2M} \cos\frac{\pi(2n+1)q}{2N}, \quad \begin{matrix} 0 \le m \le M-1 \\ 0 \le n \le N-1 \end{matrix}$$

$$\alpha_p = \begin{cases} 1/\sqrt{M}, & p = 0 \\ \sqrt{2/M}, & 1 \le p \le M-1 \end{cases} \qquad \alpha_q = \begin{cases} 1/\sqrt{N}, & q = 0 \\ \sqrt{2/N}, & 1 \le q \le N-1 \end{cases}$$

In the applications that use a fixed-point digital signal processors (DSP) or field programmable gate array (FPGA) based on fixed-point calculation, only the integers are used.

The equation $dctr := dct \cdot img \cdot dct^T$ may be implemented inside the DSP in two ways:

1- Multiply the first two matrices $dct \cdot img$, and then the output matrix is multiplied with dct^T.

2- Multiply the last two matrices $img \cdot dct^T$, and then the output matrix is multiplied with dct.

The main question here is ***“does this changing in the order of multiplication makes effect?”*** and if so ***“in what degree it effects?”***

Software TOOLS

Software has been developed in **MathCAD** to allow fast results, simultaneous visualization and immediate comparison.

Software Methodology

The algorithm implemented in the MathCAD reads the image file; divide it into fragments of 8X8 according to number of the rows or columns; the lowest is taken.

A DCT is applied to each fragment individually and a masked matrix Q is used to eliminate the lowest frequencies in the transformed matrices, then the inverse DCT is applied in each matrix individually to get the original matrices and recollecting these matrices we get a decompressed image.

In each way of applying DCT two ways (left and right) have been implemented as explained above; and in each variant the mean error is estimated and these results are exported to the “MS-EXCEL” and a chart of the means error in each block is drawn.

The output data is compared between the two ways (left and right) and a conclusion will be taken based on this.

The output results are taken to the “MS-Excel” to apply statistics and comparison; the file in Excel will be like that:

r	w	IF(r>w,1,IF(r<w,-1,0))
0.875	0.9844	-1
0.875	0.875	0
0.75	0.8281	-1
0.9844	1.0313	-1
0.8594	0.9375	-1
0.9531	1.0313	-1
0.9063	0.9219	-1
0.875	1	-1
0.875	0.875	0
0.9375	1.0313	-1
0.9375	0.9063	1
0.9063	0.9063	0
0.875	1.0313	-1
1	0.8438	1
0.9063	0.9063	0
0.9063	0.8438	1
0.5938	0.625	-1
0.8125	0.9063	-1
0.9375	0.9375	0
0.8281	0.875	-1
0.875	0.75	1
0.8281	0.8906	-1
1.0313	0.9219	1
0.9063	0.9063	0
0.9375	0.9375	0
0.8906	0.8125	1
0.8594	0.8438	1
1	0.875	1
0.8906	1.0156	-1
0.8906	0.9219	-1
0.9375	0.9219	1
1.0625	0.9375	1
0.8125	0.8594	-1
1.0156	0.875	1
0.9063	0.9063	0
1.0313	0.8906	1
1.0781	0.8281	1
1.1094	0.8438	1
0.8125	0.875	-1

Where:

r: Mean error in matrix multiplication of first two matrices $dct \cdot img$, and then the output matrix is multiplied with dct^T,it is the mean difference between the original image and the constructed image.(will call it ***left way***)

w: Mean error in matrix multiplication of last two matrices $img \cdot dct^T$, and then the output matrix is multiplied with dct.(call it ***right way***)

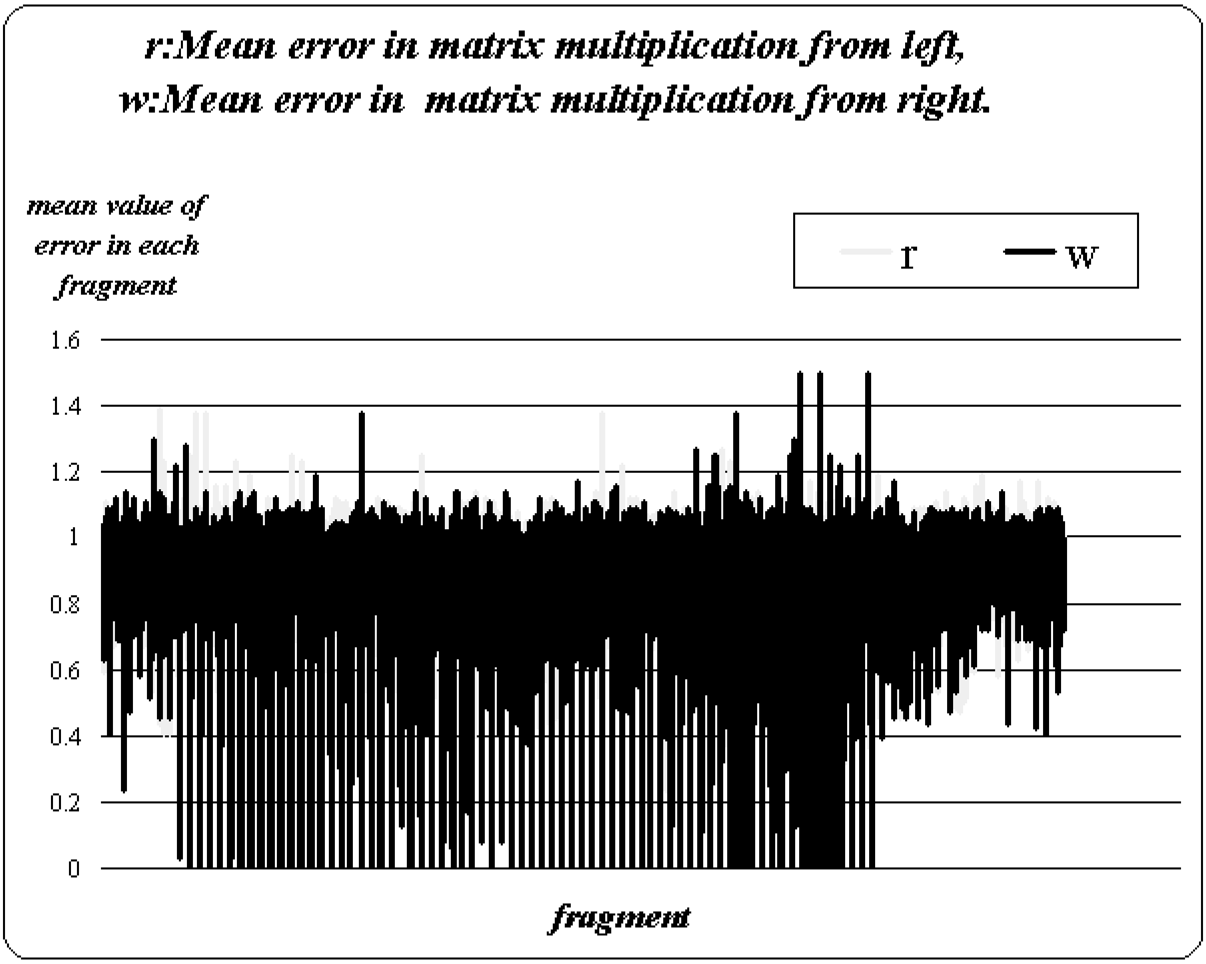


Figure 1. Comparison between the two multiplication directions r,w represent the mean error in each fragment

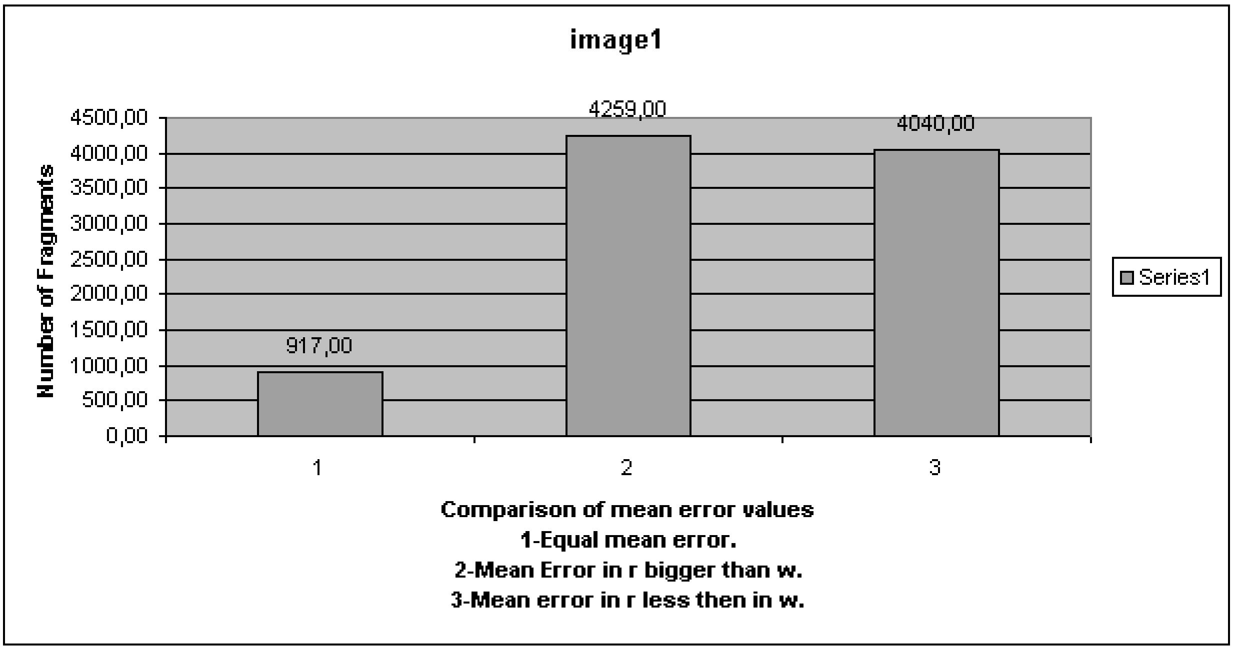


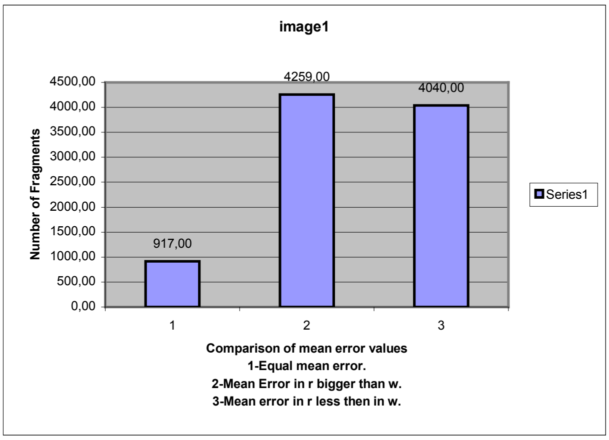
Figure 2. number of fragments with mean error

Number of zeros	***917.00***
Number of +ve ones	***4259.00***
Number of -ve ones	***4040.00***

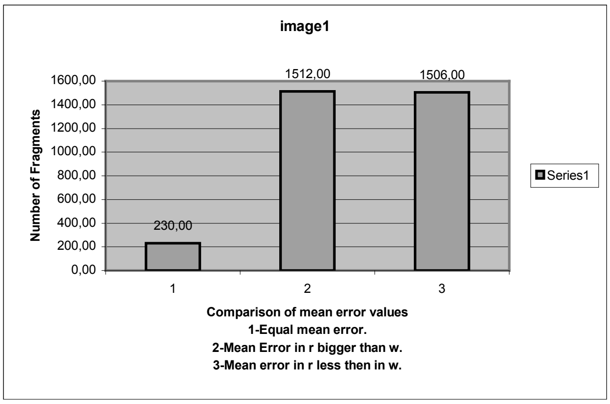
Where

Number of zeros	Represent Number of fragments where the difference between the original image and the reconstructed image is the same in two multiplication directions ***right way*** and ***left way.***
Number of +ve ones	Represent Number of fragments where difference between the original image and the reconstructed image, in ***left way*** is greater than those in ***right way***
Number of -ve ones	Represent Number of fragments where difference between the original image and the reconstructed image, in ***left way*** is less than those in ***right way***

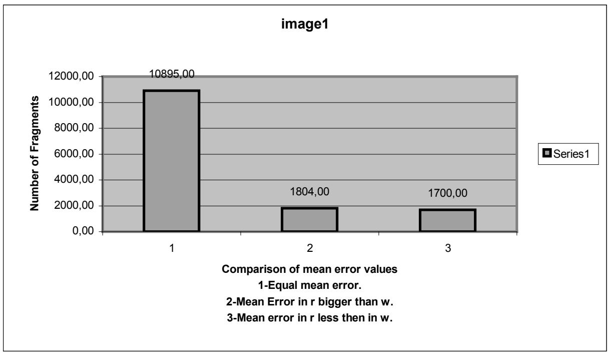
Results



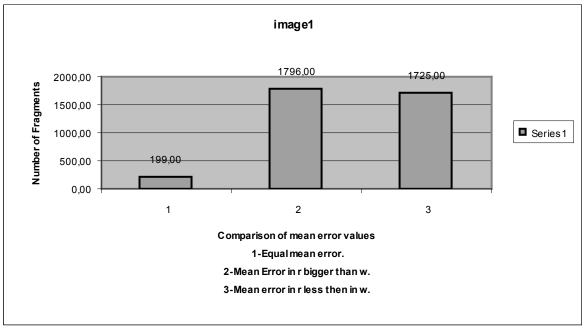
Number of zeros	***917.00***
Number of +ve ones	***4259.00***
Number of -ve ones	***4040.00***



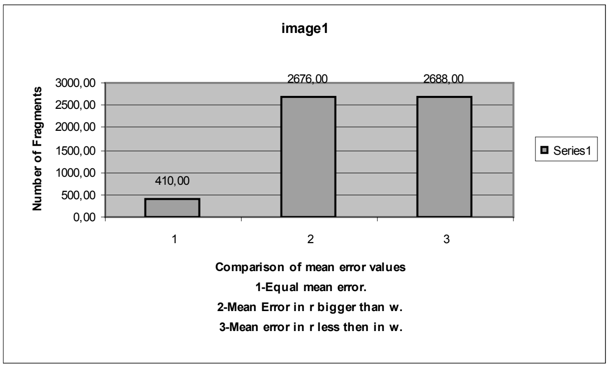
Number of zeros	***230.00***
Number of +ve ones	***1512.00***
Number of -ve ones	***1506.00***



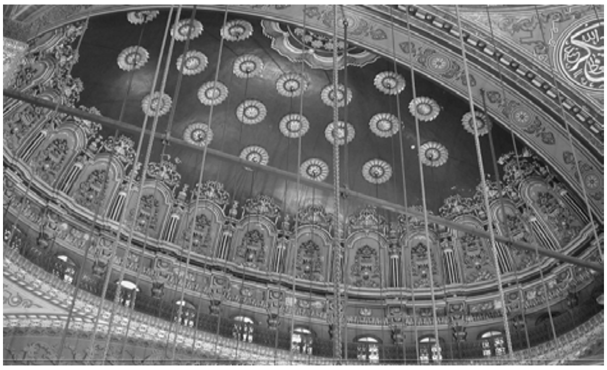
Number of zeros	*10895.00*
Number of +ve ones	*1804.00*
Number of -ve ones	*1700.00*

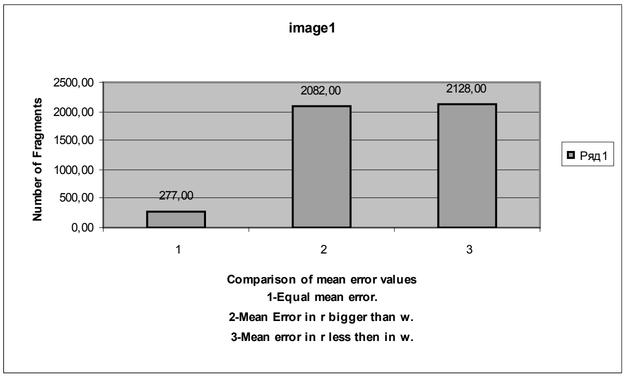


Number of zeros	*199.00*
Number of +ve ones	*1796.00*
Number of -ve ones	*1725.00*



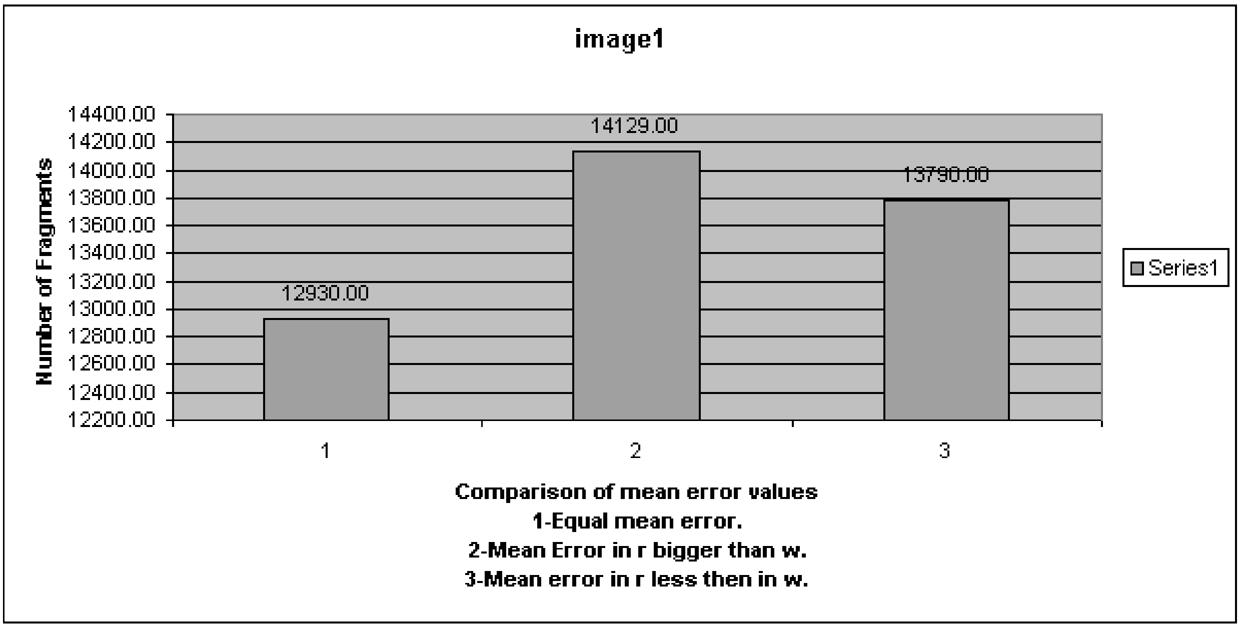
Number of zeros	*410.00*
Number of +ve ones	*2676.00*
Number of -ve ones	*2688.00*





Number of zeros	***277.00***
Number of +ve ones	***2082.00***
Number of -ve ones	***2128.00***

General Evaluation:



Number of zeros	***12930.00***
Number of +ve ones	***14129.0***
Number of -ve ones	***13790.00***

It means that the average error in matrix multiplication from left to right is statistically higher than the average errors resulted from the right to left matrix multiplication for 40849 fragments.

Conclusion

Based on statistical calculation for 40849 matrices, it was seen that applying discrete cosine transform (DCT) in a fixed point calculation system on a matrix (image), the probability of getting better average error from the original image is higher when multiplying the matrix with the transpose matrix of DCT then multiply with DCT matrix itself. Whenever the image contains wide areas with the same level, the two ways (left and right) are almost equivalent.

1. Basics of JPEG http://www.whisqu.se/per/docs/article.htm from "Denis Bukharov". 2. Jpeg2000: Image Compression Fundamentals, Standards, and Practice (by David S. Taubman, Michael W. Marcellin) http://www.amazon.com.