DCWT : Image Compression Method Based On Wavelet and DCT Transforms

Dr. Ahmed Tariq Sadiq - University of Technology
Abdulla Mohamed Awad - University of Al-Anbar

الخلاصة

في هذا البحث سنتعامل مع ضغط الصور باستخدام تحويل الجيب تمام المتقطع وتحويل Haar المويجي. الخصائص الرياضية تحويل الجيب تمام المتقطع وتحويل Haar المويجي استخدمت لاستكشاف هذا المجال كطريقة لضغط الصور الثابتة لتحسين التقنيات الموجودة وللخروج بطريقة جديدة لضغط الصور. الطريقة المقترحة اثبتت انها افضل كفاءة من تحويل الجيب تمام المتقطع وتحويل Haar الميجي كل على حدة. الدمج بين التحويلين اعطى تحسيناً ملحوظاً في الاداء. استخدمت عدة طرق للمقارنة بين النتائج لضغط الصور القياسية ومقارنتها مع الصور الاصلية. في الطريقة المقترحة تم الوصول الى نسبة ضغط تعادل تقريباً 1:22 وهذا الرقم يعتبر أفضل من الطرق الموجودة.

Abstract

This paper deals with image compression using Discrete Cosine Transform (DCT) and Haar wavelets transforms (WT). The mathematical properties of Haar wavelets and DCT are covered to explore the field of still image compression methods, which improve on existing techniques and develop new image compressors. DCWT is more efficient than corresponding DCT and wavelet transform functions. Incorporating DCT and wavelet transform is shown to improve the performance of the DCT and wavelet (Haar, IWT). Several methods are compared in terms of their ability to compress standard images and the fidelity of the reproduced image to the original image. In the new proposed method the compression rate is approximately equal to 1:22 and this rate is more efficient than that of the still image compression methods.

1-Introduction

Image compression is useful and necessary in a variety of applications. These applications can be divided into two types: transmission and storage. Transmission involves sending a file over a channel and compression reduces the number of bits to be transmitted and making the transmission process more efficient. Storing a file in a compressed form typically requires fewer bits. All the images on the web are compressed, typically in the JPEG or GIF formats, most modems use compression.
There are two types of compression, lossless compression, which can reconstruct the original image exactly from the compressed image, and lossy compression, which can only reconstruct an approximation of the original image. Lossless compression is typically used for text, and lossy compression is used for images and sound where a little bit of loss in resolution is often undetectable, or at least acceptable.

2-The Discrete Cosine Transform.

The DCT breaks down the source image into (N x N) matrix or block . In practice (N) most often equals 8 because a larger block, though it would probably give better compression often takes a great deal of time to perform DCT calculations creating an unreasonable tradeoff.

As a result, DCT implementations typically break the image down into more manageable (8 x 8) blocks. Then we apply the discrete cosine transform on the matrix. The mathematical functions for a two-dimensional DCT are:

$$y(k,l) = \frac{c(k)c(l)}{4} \sum_{i=0}^{7} \sum_{j=0}^{7} x(i,j) \cos\left\{\frac{(2i+1)k\pi}{16}\right\} \cos\left\{\frac{(2j+1)l\pi}{16}\right\}$$

Where k,l=0,…………,7 and c(k)=

$$c(k) = \begin{cases} \frac{1}{\sqrt{2}}, & k=0 \\ 1, & k<>0 \end{cases}$$

There is an Inverse DCT (IDCT):

$$x(k,l) = \sum_{i=0}^{7} \sum_{j=0}^{7} y(i,j) \frac{c(k)c(l)}{4} \cos\left\{\frac{(2i+1)k\pi}{16}\right\} \cos\left\{\frac{(2j+1)l\pi}{16}\right\}$$

3-HAAR Transform

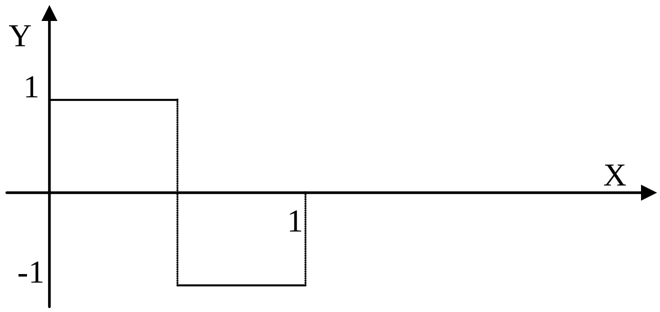
Wavelets are functions defined over a finite interval. The basic idea of the wavelet transform is to represent an arbitrary function f(x) as a linear combination of a set of such wavelets or basis functions.

These basis functions are obtained from a single prototype wavelet called the mother wavelet by dilations (scaling) and translations (shifts).

The purpose of wavelet transform is to change the data from time-space domain to time-frequency domain, which makes better compression results.

The simplest form of wavelets is the Haar wavelet function, which can be defined as: [6]

$$\psi(x) = \begin{cases} 1 & 0 \le x < 1/2 \\ -1 & 1/2 \le x < 1 \\ 0 & otherwise \end{cases}$$



The ‘Discrete Cosine Transform (DCT)’ and the ‘Wavelet Transform’ are examples of ortho-normal transforms that are used in image compression.

The transform is rarely applied to the whole image, it usually deals with small regions or image blocks independently. This has the advantage of exploiting local similarities within the image but also leads to a ‘blocking artifact’ effect found in this type of method. The blocks do not have to be a fixed size or shape, but they are usually non-overlapping. Applying an overlapping blocked system duplicates data already contained in other image blocks, and hence can be wasteful.

ISO has done extensive testing and has developed a standard set of quantization values that cause impressive degrees of compression. To determine the value of the quantum step sizes, the user inputs a single quality factor, which should range from one to about twenty-five. Values larger than twenty-five would work, but the picture quality degrades far enough at quality level 25.

The quality level sets the difference between adjoining bands of the same quantization level. These bands are oriented on diagonal lines across the matrix, so quantization levels of the same value are all roughly the same distance from the origin.

To calculate the quantization matrix we use this formula:

$$Q(i, j) = 1 + (1+i+j) \quad \textit{Quality factor}$$

An example of what the quantization matrix looks like is with a quality factor of two. If quality factor = 2 then the table is as shown below :

3	***5***	***7***	***9***	***11***	***13***	***15***	***17***
5	***7***	***9***	***11***	***13***	***15***	***17***	***19***
7	***9***	***11***	***13***	***15***	***17***	***19***	***21***
9	***11***	***13***	***15***	***17***	***19***	***21***	***23***
11	***13***	***15***	***17***	***19***	***21***	***23***	***25***
13	***15***	***17***	***19***	***21***	***23***	***25***	***27***
15	***17***	***19***	***21***	***23***	***25***	***27***	***29***
17	***19***	***21***	***23***	***25***	***27***	***29***	***31***

Quantization Matrix

4-Merging (Haar Wavelets and DCT) Image Compression

This proposed algorithm applies (HAAR and DCT) transform to image compression so that after we apply the Haar transform to image, the DCT is applied to the first quarter of image and all the values in (LH,HL and HH) are discarded .

The image that is to be compressed is processed in 8x8 blocks. The DCT is applied to each of these blocks. The coefficient set that is obtained has to be quantized by scaling of each coefficient block with a known matrix after scaling of coefficients, many of these will be zero, that means they are no more important. The rest of coefficients represent the image in compressed form. The compression rate obtained using this method depends on the values of the elements of the quantization matrix. As long as these elements have high values, we will obtain more coefficients with value equal to zero, so the compression rate will be higher and this will increase also the loss of information in compression process.

A- The DCWT Image Compression Coding Algorithm

(1)-Read the image data.

(2)-The two-dimensional HAAR Wavelet is computed for image block.

(3)-The 2D-FDCT is computed for the first quarter of the coefficient matrix [low-low frequency] in step (2).

(4)-The majority of the coefficients are discarded by selecting a suitable threshold.

(5)- Reorder the DCT coefficients of the image array in the first quarter by zig-zag order.

(6)- Reorder the DCT coefficients of the image array in the first quarter from 2-dimensional array to one dimensional array, which has the values of rows in location row mod 8=0 only and discard all the values in other rows.

(7)- Apply RLE to the zero's value.
(8)-Apply shift-coding to get image data in stream of bits in a binary file.

Figure (1) shows the block diagram of the above algorithm.

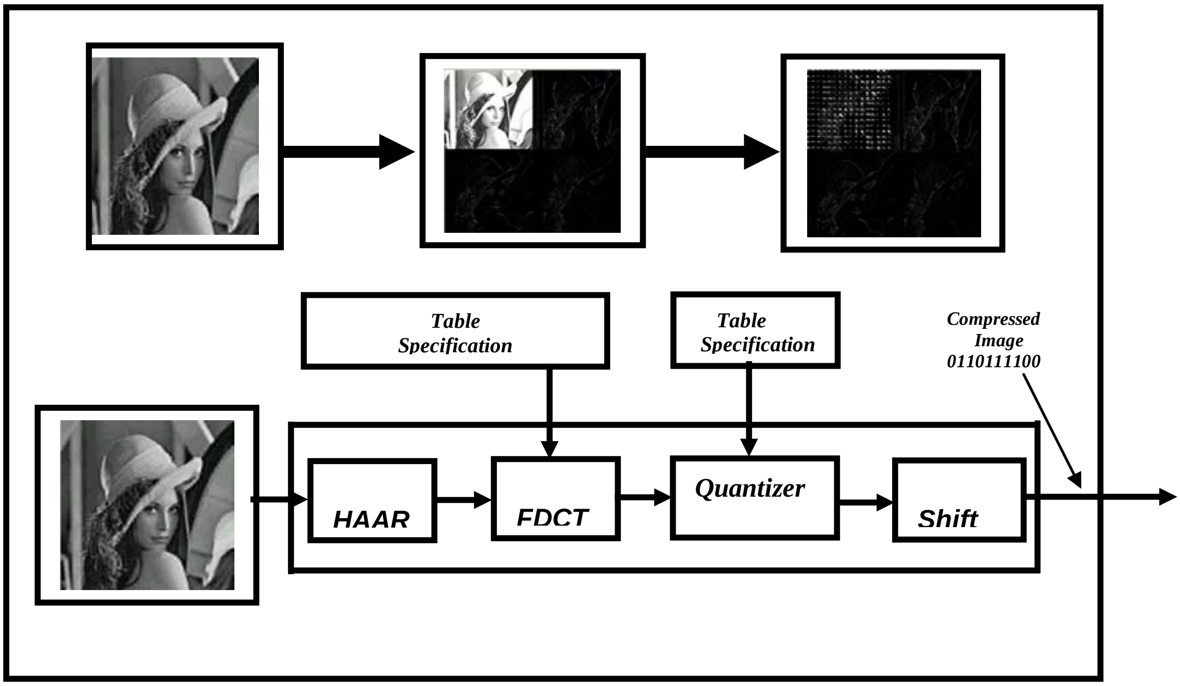


Figure (1)

B-The DCW

(1)-Receive the coefficient from the binary file.
(2)-Apply shift-decoding to get image data in decimal representation.
(3)- Apply decoding RLE to the zero's value.
(4)- Reorder the coefficients of the image array from 1-dimensional array to 2-dimensional array.
(5)-The selected threshold is multiplied by the image coefficients.
(6)- Reorder the DCT coefficients of the image array in the the first quarter from zig-zag order.
(7)-The 2-D IDCT is computed for first quarter of image coefficient array to get [low-low frequency].
(8)-The 2-D inverse HAAR Wavelet is computed for image block

Figure (2) shows the block diagram of decoding DCWT.

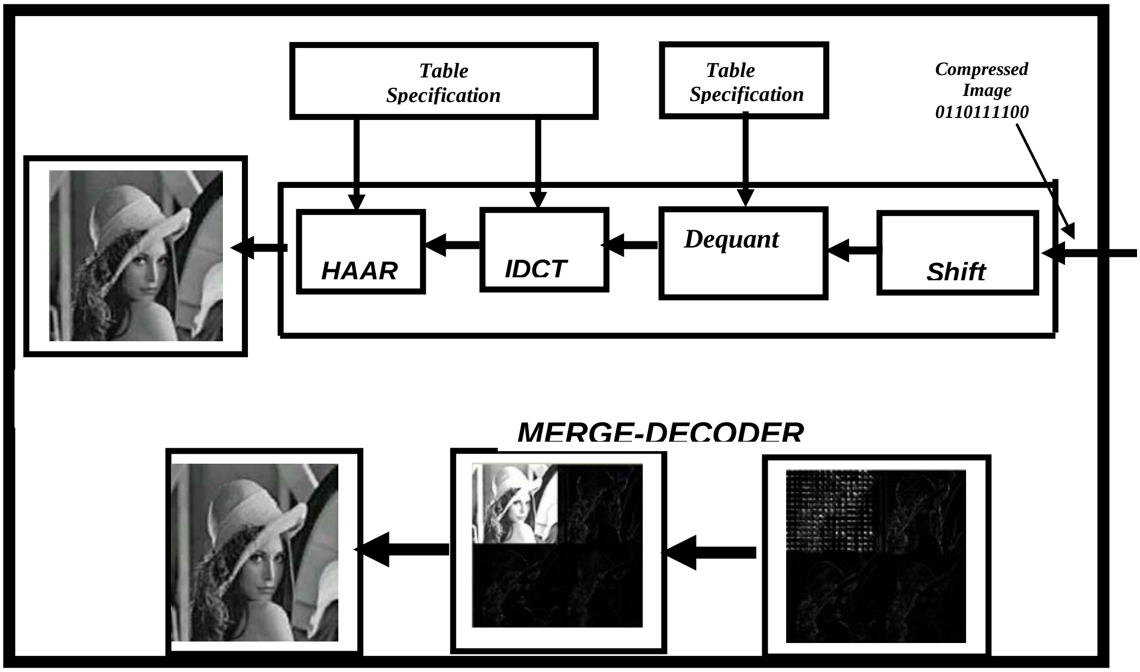


Figure (2)
The DCWT Decoder Block Diagram

5-Discussion

The compression results produced are discussed. They include the subjective and objective qualities of reconstructed images in this method, timing of composition and decomposition for different wavelet types, and timing of DCT algorithm for different compression ratios.

Two popular measures of distortion are the squared error measure and the absolute difference measure, which is called difference distortion measures. If X{n} is the source output and Y{n} is the reconstructed sequence, the squared error measure is given by

$$d(x,y)=(x-y)^2$$

And the absolute difference measure is given by

$$d(x,y)=\left|x-y\right|$$

Practically, it is difficult to examine the difference on a term-by-term basis. Some average measures are used to summarize the information. The most often used average measure is the average of squared error measure.

The average squared value of the source output and the MSE is called the signal-to-noise-ratio (SNR).

The peak-signal-to-noise ratio (PSNR) is calculated by the following equation:

$$PSNR(dB) = 10\log_{10}\frac{x_{peak}^2}{(MSE)^2}$$

PSNR is the most commonly used value to evaluate the objective image compression quality. Tables (1) and (2) show the experimental results.

Table (1)
Compression Results by Using Different Wavelets and DCT

Compression Method	**Image**	**Compression Ratio**	**Compression File size**	**PSNR**
Discrete Cosine Transform	Lena	1:13	5.14 KB	25
	Housebreak	1:13	5 KB	27
Haar Wavelet	Lena	1:7	9.6 KB	28
	Housebreak	1:8	8.13 KB	30
IWT Transform	Lena	1:10	6.5 KB	28
	Housebreak	1:14	4.45 KB	30
DCWT	Lena	1:22	2.98 KB	24
	Housebreak	1:22	2.87 KB	25.5

Table (2)

Compression factor by DCT, Wavelets and DCWT

Image Size (KB)	**DCT**		**Haar Wavelet**		**IWT**		**DCWT**	
	Com File size	**Com. factor**	**Com File size**	**Com. factor**	**Com File size**	**Com. factor**	**Com File size**	**Com. factor**
1	**0.109**	**9.4**	**0.34**	**3**	**0.37**	**3**	**0.060**	**17**
4	**400**	**10.2**	**1.01**	**4**	**1.1**	**4**	**0.220**	**18.5**
16	**1.4**	**11.7**	**3.93**	**5.6**	**3.19**	**5.13**	**0.851**	**19.25**
64	**5.22**	**12.5**	**9.6**	**6.8**	**10.6**	**6.2**	**2.98**	**22**
256	**19.5**	**13.4**	**27**	**9.7**	**25.7**	**10.2**	**11.2**	**23.4**

In Figure (3), when we look at the DCT, Wavelet (Haar and Integer) and DCWT transforms and compare their implementation, we find that DCWT gives a superior compression over DCT and Wavelet as shown in the above figure. A comparison between all the above methods has shown they hold a promise.

The DCWT algorithm is a general-purpose image compression algorithm. In this algorithm, first the Haar-Wavelet transform is applied and in the second the DCT is applied to the first quarter of the image in the upper left corner (low-low filter).

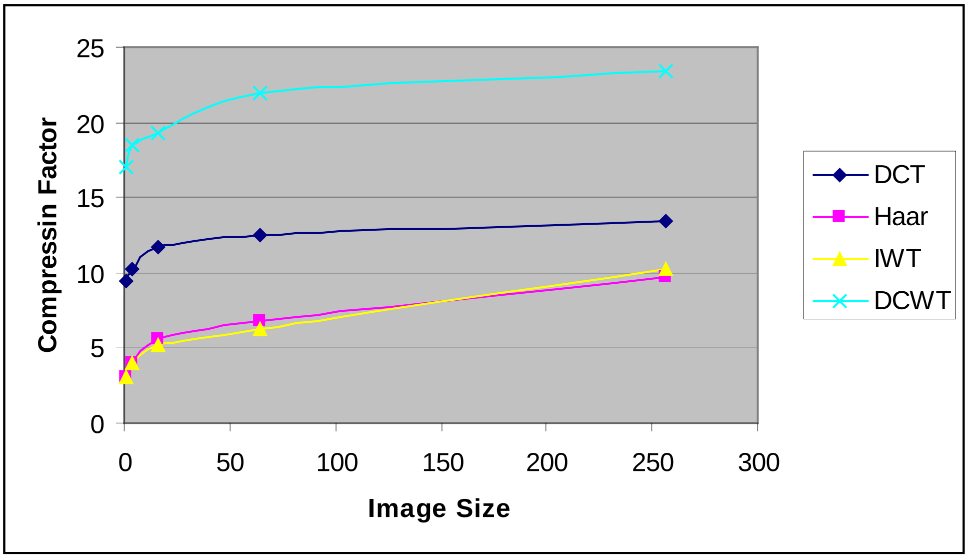


Figure (3)

The Relation between Image Size and Compression Factor

6-Conclutions

The experimental results for this new algorithm show that the algorithm using DCWT can achieve a much higher compression ratio than image compression methods using only DCT or Wavelet transforms. There are several important points drawn from this comparison :

1- The DCT is one of efficient methods for image compression but it takes time.
2- Haar Wavelet gives the worst compression rates but it is a fast method.
3- DCWT gives the best compression ratio and it needs less time than DCT but this time is longer than that of wavelet (Haar and Integer) transforms but this time is acceptable if we look at it from compression ratio point of view.

References

[1] David Bethel, ”***Optimization of Still Image Compression Techniques***”, Ph.D. Thesis, University of Bath, 1997.

[2] Mark Nelson, ”***The Data Compression Book***”, P.347-373/M&T Publishing, 1991.

[3] Scott E Umbaugh, ”***Computer Vision and Image Processing***”, Prentice-Hall NJ, 1998.

[4] David Salmon, ”***Data Compression***”, Prentice-Hall, 2000.

[5] Eric E Majani, ”***Algorithm For Lossy /Progressive Loss less Image Compression***”, NASA Tech. Brief Vol. -21.NO.12, Item # 146, December, 1997.

[6] Panrong Xiao, ”***Image Compression by Wavelet Transform***”, M. Sc. Thesis, East Tennessee State University, August, 2001.

[7] K. Sayood, “***Introduction to Data Compression***”, Second edition, Academic Press, 2000.

[8] C.S Burrus R.A. Gopinath, and H.Guo. “***Introduction to Wavelets and Wavelet Transforms***”, Englewood Cliffs, NJ: Prentice Hall, 1998.

[9] H. Gu, “***Image Compression Using the Haar Wavelet Transform***”, M.Sc. Thesis, East Tennessee State University, 2000.

[10] G. Strang, “***The Discrete Cosine Transform***”, Society for Industrial and Applied Mathematics, 1999.

[11] P. Morton and A. Petersen, “**I*mage Compression Using the Haar Wavelet Transform***”, College of the Redwoods, 1997.