Transform Based Hybrid Image Compression Techniques in Conjunction with Fractal Image Compression Scheme.

Sumegha Yadav  
Department Of Electronics & Communication, Netaji Subhas Institute of Technology  
(University Of Delhi)  
New Delhi, India  
sumegha19@gmail.com,  

Dr. Tarun Kumar Rawat  
Department Of Electronics & Communication, Netaji Subhas Institute of Technology  
(University Of Delhi)  
New Delhi, India  
drtkrawat@gmail.com

Keywords—DCT, DWT, Fractal Image Compression, PSNR, SSIM, encoding time, Hybrid Techniques

Abstract—In a distributed environment large image files remain a major bottleneck within systems. Compression is an important component of the solutions available for creating file sizes of manageable and transmittable dimensions. Conventional method of employing DCT or the state-of art by using Fractals and Wavelets, each method is unique and has it’s independent advantages and drawbacks. The following paper has proposed three techniques and their respective hybrid models derived from above listed schemes, that will help usher the technology into a new sphere. Below are enlisted two hybrid Image Compression methods wiz DCT based Fractal Image Compression, Wavelet based Fractal Image Compression.
I. INTRODUCTION

Fractal shapes occur universally in the natural world. The term fractal was coined in 1975 by Benoît Mandelbrot, from the Latin word ‘fractus’, meaning "broken" or "fractured." In colloquial usage, a fractal is a shape that is recursively constructed or self-similar, that is, a shape that appears similar at all scales of magnification and is therefore often referred to as "infinitely complex". One of the most important foundations for fractal image compression is the concept of iterated function systems (IFS). Through IFS we are able to systematically reproduce fractals which occur in nature. The approach pioneered by Michael F. Barnsley is to use the similarities on different scales throughout images to assist in compression. Michael F. Barnsley's discovery of the fractal transform in 1988 was preceded by B. Mandelbrot's development of fractal geometry. The fractal transform for image-data compression was introduced first by Barnsley and Demko[3]. However, the algorithm requires extensive computations [4]. The first practical fractal image compression scheme was introduced by Jacquin [1] and Jacobs et al. [2] called the Jacquin–Fisher algorithm using block-based transformations and an exhaustive search strategy. Their scheme was a better version of the system patented by Barnsley [5],[6]. The procedure of the Jacquin–Fisher algorithm consists of:

1) use of a two-level (8 × 8 and 4 × 4) square block-size partitioning method to partition the original image into blocks called range blocks;
2) classification of the image blocks into shade, midrange, and edge blocks;
3) an exhaustive search of a “virtual codebook” obtained from a pool of domain blocks of the same class and also from a set of pool-of-blocks transformations.

There are two important issues which need to be improved in the Jacquin–Fisher algorithm. First, the cost of an exhaustive search of a pool of domain blocks is too high. Second, a good classification algorithm needs to be obtained. An algorithm that aims at reducing the search space and also to decrease the order of the fractal function.

To address this particular problem we hereby propose incorporation of DCT and Wavelet with Fractal algorithm respectively, both of which being transform based coding techniques.

II. RELATED WORK

To decrease the computational cost during the domain block search, some efforts have been directed on reducing the comparing complexity for making encoding faster. The classification or clustering methods [7, 8, 9, 10], which speed the search up by restricting the search space to a subset of the domain block pool in which features extracted from the blocks are represented.

Organizing the domain blocks into a tree structure gives faster searching as compared to a linear search [7,8]. This approach can provide a reasonable speed-up while maintaining the same quality of compression. Fast fractal image compression using spatial correlation [12] is another method. DCT based classification have been used for speed up the comparison time [10,11]. Another approach is to minimize the search by excluding domain blocks. Examples include Jacobs's method of skipping adjacent domain blocks [12], Tong's method an adaptive search algorithm based on the standard deviation(STD) difference between range and domain blocks[13]. However in this paper we propose two algorithms, both
having the baseline of Fractal algorithm but using DCT in one and DWT in another in a way that would help in reducing the long encoding times of alone Fractal algorithm while maintaining the same image quality and PSNR ratios.

III. FRACTALS AND ITERATED FUNCTION SYSTEMS

Iteration is a process, or set of rules, which one repeatedly applies to an initial state. One could even define an iteration as a repetitive task. As described by Kominek[16], the metaphor of a Multiple Reduction Copying Machine is an elegant way to introduce Iterated Function Systems. In the example demonstrated in Figure 1 we will produce the Sierpinski Triangle, one of the most well known Iterated Function Systems. To reach this attractor there are three rules which, when composed together, act as the lenses in the copy machine.

Figure 1. Multiple Reduction Copying Machine using three rules to produce the Sierpinski Triangle.

Each rule, or lens, reduces the original seed by half the original size and translates the new image to a new location.

A. Metric Spaces, Mappings, Transformations

A metric space \((X,d)\) is a set \(X\) together with a real-valued function \(d:X \times X \rightarrow R\), which measures the distance between pairs of points \(x\) and \(y\) in \(X\). \(d\) is called a metric on the space \(X\) when it has the following properties [2]:

- \(d(x,y) = d(y,x), \quad \forall x, y \in X\)
- \(d(x,y) \geq 0, \quad \forall x, y \in X\)
- \(d(x,y) = 0 \iff x = y, \quad \forall x, y \in X\)
- \(d(x,y) \leq d(x,z) + d(z,y), \quad \forall x, y, z \in X\)

The Euclidean Plane, \(R^2\), along with the Euclidean metric, \(d(x,y) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2}\), \(\forall x, y \in R^2\), is an example of a metric space.

Affine transformations on \(R\) are transformations of the form where \(a\) and \(b\) are real constants. If \(a < 1\), then this transformation contracts the line toward the origin. If \(a > 1\), the line is stretched away from the origin. If \(a < 0\), the line is flipped 180° about the origin. The line is translated, or shifted, by an amount \(b\). If \(b > 0\), then the line is shifted to the right. If \(b < 0\) the line is translated to the left. We will consider affine transformations on the Euclidean plane.
Let $w: \mathbb{R}^2 \to \mathbb{R}^2$ be of the form $w(x, y) = (ax + by + e, cx + dy + f) = (x', y')$ where $a, b, c, d, e$, and $f$ are real numbers, and $(x', y')$ is the new coordinate point. This transformation is a two-dimensional affine transformation.

We can also write this same transformation with the equivalent notations:

$$w(x) = w\left(\begin{pmatrix} x \\ y \end{pmatrix}\right) = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} e \\ f \end{pmatrix} = Ax + T,$$

where $A$ is a real matrix and $T = \begin{pmatrix} e \\ f \end{pmatrix}$ represents translations.

**B. Convergence and Contractions**

A sequence $\{x_n\}_{n=1}^{\infty}$ of points in a metric space $(X, d)$ is said to converge to a point $x \in X$ if for any given number $\varepsilon > 0$, there is an integer $N > 0$ such that $d(x_n, x) < \varepsilon$ for all $n > N$.

A transformation $f: X \to X$ on a metric space $(X, d)$ is called contractive, or a contraction mapping, if there is a constant $0 \leq s < 1$ such that $d(f(x), f(y)) \leq (s)(d(x, y)) \forall x, y \in X$. Iterated Function Systems set the foundation for Fractal Image Compression. If these mappings are contractive, applying the IFS to a seed image will eventually produce an attractor of that map.

**IV. DISCRETE COSINE TRANSFORM**

Like other transforms, the Discrete Cosine Transform (DCT) attempts to de-correlate the image data. Transform compression is based on a simple premise: when the signal is passed through a certain transform, the resulting data values will no longer be equal in their information carrying roles.

In particular, the low frequency components of a signal are more important than the high frequency components. Basic equations involving DCT are: for 1-D DCT:-

$$\hat{x}_v = \sum_{k=0}^{N-1} x_k C_v \cos\left(\frac{(2k+1)v\pi}{2N}\right), \quad v = 0, ..., N-1,$$

$$x_k = \sum_{v=0}^{N-1} \hat{x}_v C_v \cos\left(\frac{(2k+1)v\pi}{2N}\right), \quad k = 0, ..., N-1,$$

where $C_0 = \sqrt{1/N}$ and $C_k = \sqrt{2/N}$ if $k \neq 0$.

$\hat{x}$ is usually referred to as the forward cosine transform, while $x$ is referred to as the backward cosine transform.

Similarly for 2-D DCT:-

$$\hat{f}_{uv} = \sum_{j,k=0}^{N-1} f_{jk} C_u \cos\left(\frac{(2j+1)u\pi}{2N}\right) C_v \cos\left(\frac{(2k+1)v\pi}{2N}\right),$$

$$f_{jk} = \sum_{u,v=0}^{N-1} \hat{f}_{uv} C_u \cos\left(\frac{(2j+1)u\pi}{2N}\right) C_v \cos\left(\frac{(2k+1)v\pi}{2N}\right),$$

where $C_0 = \sqrt{1/N}$ and $C_k = \sqrt{2/N}$ if $k \neq 0$.
is usually referred to as the forward cosine transform, while \( \hat{f} \) is referred to as the backward cosine transform. DCT-based image compression relies on two techniques to reduce the data required to represent the image. The first is quantization of the image's DCT coefficients; the second is entropy coding of the quantized coefficients.

V. DISCRETE WAVELET TRANSFORM

What is a ‘wavelet’ actually? Very correctly as Wikipedia puts it: “A wavelet is a wave-like oscillation with an amplitude that begins at zero, increases, and then decreases back to zero. It can typically be visualized as a "brief oscillation". Generally, wavelets are purposefully crafted to have specific properties that make them useful for signal processing. Wavelets can be combined, using a "reverse, shift, multiply and sum" technique called convolution, with portions of an unknown signal to extract information from the unknown signal.”

Haar defined a complete set of orthogonal system of functions in \( T^s[0,1], S \in [0,\infty] \).

\[
\text{Haar}(0,t) = 1, t \in [0,1];
\]

Original Haar definition is as follows:

\[
\text{Haar}(1,t) = \begin{cases} 1, t \in \left[0, \frac{1}{2}\right), \\ -1, t \in \left[\frac{1}{2},1\right] \end{cases}
\]

The wavelet methods are strongly connected with classical basis of the Haar functions; scaling and dilation of a basic wavelet can generate the basis Haar functions. Mother wavelet : \( \varphi(t) \), we can obtain the rest of the function for the transformation i.e child wavelet \( \varphi_{a,b}(t) \) by scaling and translating the mother wavelet.

\[
\varphi_{a,b}(t) = \frac{1}{\sqrt{2}} \varphi \left( \frac{t-b}{a} \right)
\]

\[
w_{a,b} = \langle \varphi_{a,b}, f(t) \rangle = \int_{-\infty}^{\infty} \varphi_{a,b}(t)f(t) dt
\]

VI. DCT BASED FRACTAL IMAGE CODER

As the proposal goes, we would first structure an encoding mechanism amalgamating both DCT and Fractal in a manner that encoding times of Fractal algorithm are reduced while preserving the proven excellent PSNR ratios of Fractal Image Compression algorithm and image quality.

A. Compression process

A domain pool is formed by segmenting the image into 8x8 non-overlapping blocks. DCT was parallel employed on each of the domain pool block.

Quantisation being a pillar of DCT algorithm was then applied using a quantization matrix. Zero coefficients are removed by scanning in Zig-zag manner. Now we enter into the domain of Fractal encoding stage. Domain pool is formed by segmenting the image into 8x8 non-overlapping blocks. Averaging of domain block by taking 2x2 pixels at a time, within the domain block. New scaled domain block of
size 4×4 is formed. Subtract average of the domain from each entry in the domain block. Thus resulting scaled domain block is D. 8 different dihedral transformations of each domain block in an eight dimensional matrix (M) are stored. Offset of the range blocks is stored, which we will add back to the image later.

Algorithm cycles through each domain block and tests each symmetry that is stored in M, along with the four possible gray scales for the best transformation that will map to a given range block. Best map is found, the location of that domain block, the best symmetry of the domain block, the best scaling, and the offset is saved in the five dimensional matrix.

Fractal parameters, stored in a five dimensional matrix are decoded first and by using repeated transform we get an arbitrary image by fractals. A DCT domain fractal approximation of the original image is produced by this process. Conversely, we get the difference image by decoding the Huffman code and de-quantizing it.

VI. DWT BASED FRACTAL IMAGE CODER

Wavelet transform (Mallat, 1989; Marta et al., 2003) approaches to image compression exploit redundancies in scale. Wavelet transform data can be organized into a sub tree structure that can be efficiently coded. In Hybrid fractal-wavelet techniques domain-range transformation idea of fractal encoding is ap-

Figure 2 Block Diagram for DCT based Fractal Encoder
plied to the realm of wavelet sub trees. Improved compression and decoded image fidelity is obtained. A range tree is fractally encoded by a bigger domain tree.

A. Compression process
DWT is applied that dis-integrates the image into several sub-bands, mainly approximation sub-band and detail sub-band. Apply Haar transform twice on the image, to obtain wavelet coefficients. The approximation sub-band wavelet coefficients are taken from the transformed image. The image is now segmented into Range blocks, further variance of each range block is calculated.

According to the given formula, where $R$: range block, $S$: size of the range block, $r_{i,j}$: gray level value at position $i, j$ and $\mu_R$ is the mean of that block, $Var(R)=\frac{1}{s^2} \sum_{0 \leq i, j \leq s} (r_{i,j} - \mu_R)^2$.

Minimum variance is coded by mean; otherwise it is coded by contractive affine transform. In FIC, each range block is represented by the following information. First parameter represents whether the transformation is of affine or mean type, second is the position of domain block and it's respective isometric transformation type. Flag 1 is set if the range block is coded by mean else flag is set 0 i.e range block coded by affine transformation. There are totally eight isometric transformations.

B. De-compression process
At the decoder side, the compressed image is the input, along with the coded file. It is read sequentially, and correspondingly flags 0 are replaced by mean pixel, and for flags 0, the subsequent block position is read from coded file. The block is replaced by the corresponding domain block which is taken from domain pool using block position. Inverse DWT operation for all blocks is performed to recover the reconstructed image.

Figure 3: Block diagram for DWT-Fractal encoder
VII. RESULTS AND DISCUSSION

Arriving finally here to affirm our made proposal of amalgamating DCT and DWT respectively with Fractal technique, to mainly reduce it’s large encoding times while maintaining the image quality to it’s best. The above illustrated algorithms for both the hybrid techniques were implemented using MATLAB R2009b. The aforesaid proposal can be verified on the basis of three parameters used:

\[
PSNR = 10 \log \left( \frac{255^2}{MSE} \right) dB
\]

Where \( MSE = \frac{1}{MN} \sum (f'(x,y) - f(x,y))^2 \)

Here MN is the total number of pixels in the image \( f'(x,y) \) is the decompressed image and \( f(x,y) \) is the original image.

SSIM: The similarity measure that compares the local patterns of pixel intensities that have been normalized for luminance and contrast is known as Structural Similarity (SSIM).

\[
SSIM(x,y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{\left(\mu_x^2 + \mu_y^2 + C_1\right)\left(\sigma_x^2 + \sigma_y^2 + C_2\right)}
\]

Here \( \mu_x \) and \( \mu_y \) are the mean value of the luminance in the original and decompressed image respectively. \( \sigma_x \) and \( \sigma_y \) are the standard deviation of the luminance. \( C_1 \) and \( C_2 \) are the contrast values of the original and decompressed images.

Time taken by processor to compress the image. Below are shown graphs of the proposed technique compared with baseline JPEG on PSNR and SSIM basis.

The above graph clearly indicates a step in PSNR values from the most commonly used JPEG scheme.
Above is the graph illustrating the structural similarity index (SSIM). The values again reaffirm our claim of proposed techniques being better than the commonly existing and used schemes for compression.

Depicting now is the very criterion we wanted to reduce, that is the enormous time Fractal compression scheme takes to compress image. The results show that times for encoding are hugely suppressed by using these new algorithms. For 0.50 bit rate following results were observed.

After having done factual analysis by use of various parameters we now finally present the original image and its respective decompressed image produced by using hybrid schemes and alone Fractal algorithm(Figure 7(a),7(b),7(c),7(d))
VIII. CONCLUSION:

By going through the various analysis based on PSNR, SSIM, time complexity and also by the mere visual perception of the decompressed image, we may comfortably arrive at this conclusion that hybrid technique of using DWT & Fractal helps reduce the time complexity and renders best image quality verified on all the parameters listed above. It has best PSNR, SSIM, least time taken to compress image and the best visual quality.

REFERENCES:


IJOAR© 2013
http://www.ijoar.org