A Hybrid Image Compression Scheme Combining Block-based Fractal Coding and DCT

Yao Zhao   Baozong Yuan

Institute of Information Science,Northern Jiaotong University, Beijing 100044, P.R.China
Email: yzhao@center.njtu.edu.cn

Abstract

A novel hybrid image compression scheme based on block-based fractal coding and discrete cosine transform(DCT) is proposed in this paper. An original image is first compressed by DCT domain fractal transform instead of spatial domain fractal transform, then the difference image between the original image and its fractal approximation is quantized and encoded by a Huffman code. The detailed steps to implement the scheme and some experimental results are presented. A comparison with other fractal-based image coding methods is also made in the paper. Experimental results show that when compressing the standard image "Lena" with this technique, for a compression ratio of 12.4, its PSNR can reach 31.7 dB.

Key words: Image Processing, Image Coding, Fractals, DCT.

1. Introduction

Digital images have been used in many different fields. The large volume of data required to describe such images makes transmission and storage slow and impractical. The information contained must be compressed by extracting only the visible elements. Image compression theory has been researched for many years. A lot of compression methods have been proposed, such as predictive coding, DCT, VQ and so on; Meanwhile, in order to achieve high compression and high fidelity, researchers are investigating and developing new compression methods. Fractal-based image coding is one of them.

The first automatic fractal image compression scheme that can compress any monochrome image was proposed by Jacquin in 1990[1]. It modeled the original image as a fractal picture and made use of the blockwise self-similarity of the original image to search for the parameters of fractal transform. The method can greatly reduce the redundancy, so it can provide high compression. Since 1990, this theory has attracted a degree of attention and a number of papers related to the subject have been published[2][3][4][5][6][7]. However, experiments show that even though the method can achieve high compression, reconstructed images are of medium quality. In order to improve the decoded image quality, the size of divided blocks has to be decreased which results in smaller compression ratio. Jacquin proposed a two-level partition technique which can prevent the compression ratio from falling a lot while improving the decoded image quality. In the best experimental result presented in[1], the bit rate is 0.68 bpp and the PSNR is 27.7 dB for the 256×256 "Lena" image..

In order to improve the performance of a fractal-based compression scheme, a hybrid compression scheme combining fractals and DCT is proposed. In the scheme, all fractal transforms are carried out in DCT domain. Before searching for the most similar block, all blocks are transformed by DCT and quantized with a quantization table considering the properties of human visual system. So the matching process is completed without considering those DCT coefficients which are small or unnecessary to the human eyes. In order to improve the quality further, the error image between the original image and its fractal approximation is quantized and encoded by Huffman code.

2. The Coding Scheme

The basic theory of fractal-based image coding can be found in[1]. Here, we simply introduce our scheme.

2.1 Compression Procedure

The compression procedure diagram is as follows:

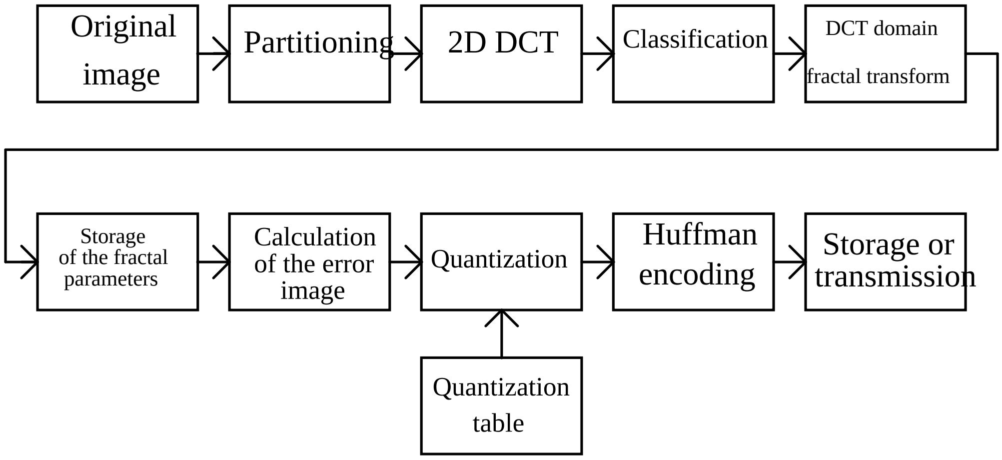

Fig. 1. The compression procedure

Our original image is of size $256\times256\times8$. Before compressing, we subtract 128 from all pixels of the image.

The original image is first partitioned into two types of blocks whose sizes are 8×8 and 16×16. The smaller are called range blocks which are non overlapping and the larger are called domain blocks which may overlap. The domain blocks are constructed in the following way: Place a 16×16 window on the left top of the original image, then slide the window across the image by every 8 pixels vertically or horizontally. Each time, the pixels in the window form a domain block.

For every range block, transform it by DCT. The 8×8 DCT formula used is:

$$F_R(u,v)=\frac{1}{16}c(u)c(v)\sum_{x=0}^{7}\sum_{y=0}^{7} f_R(x,y)\cos\frac{(2x+1)u\pi}{16}\cos\frac{(2y+1)v\pi}{16} \qquad (1)$$

where $c(u),c(v)=\begin{cases}1/\sqrt{2} & u,v=0\\ 1 & \text{otherwise}\end{cases}$

$f_R(x,y)$ denotes the spatial domain range block and $F_R(u,v)$ is its DCT domain range block.

In order to achieve high compression, we will classify the range blocks into two types according to their complexity and process them differently. In the DCT domain, AC coefficients indicate the complexity of the block. So in our scheme, three AC coefficients are used to decide which class a block belongs to. The three coefficients are shown in Fig. 2.

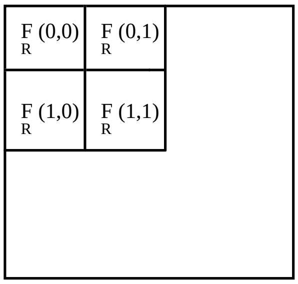

Fig. 2. The three AC coefficients used to classify the range blocks

The decision formula is equation (2).

$$\text{If } |F_R(0,1)| + |F_R(1,0)| + |F_R(1,1)| = \begin{cases} < T & F_R(u,v) \text{ is a low activity range block} \\ \geq T & F_R(u,v) \text{ is a high activity range block} \end{cases} \quad (2)$$

Where $|\cdot|$ denotes the absolute value of its variable. T is a threshold. In our scheme, *T=15.0*.

For a low activity range block, AC coefficients are small and therefore can be omitted without affecting the fidelity much. In the spatial domain, it is a smooth block. So we approximate it by storing only its DC coefficient. From (1), we get

$$F_R(0,0) = \frac{1}{32}\sum_{x=0}^{7}\sum_{y=0}^{7} f_R(x,y) \quad (3)$$

For $-128 \leq f_R(x,y) \leq 127$, so $-256 \leq F_R(0,0) \leq 255$, that is, it needs 9 bits to store $F_R(0,0)$.

For a high activity range block, its fractal coding procedure is to search for suitable transformations τ, φ and domain block $F_D(u,v)$ to satisfy or approximately satisfy the following equation.

$$F_R(u,v) = \tau \circ \phi(F_D(u,v)) \quad (4)$$

$F_R(u,v)$ is a high activity range block to be coded. $F_D(u,v)$ is a DCT domain block whose size is 16×16. The 16×16 DCT formula used here is:

$$F_D(u,v) = \frac{1}{64}c(u)c(v)\sum_{x=0}^{15}\sum_{y=0}^{15} f_D(x,y)\cos\frac{(2x+1)u\pi}{32}\cos\frac{(2y+1)v\pi}{32} \quad (5)$$

Where $c(u), c(v) = \begin{cases} 1/\sqrt{2} & \text{u, v} = 0 \\ 1 & \text{otherwise} \end{cases}$

φ is a contractivity operator which maps a 16×16 block onto a 8×8 block. It is the necessary condition which ensures the convergence of decoding procedure. In DCT domain, it takes the low frequency part of the domain block shown as below:

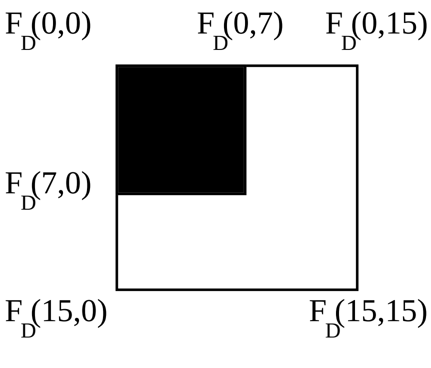

Fig. 3. The contractivity operator and its result. The black part is its mapping result.

τ are the compound transformations composed of an isometry, a scaling and a luminance shift of the form:

$$\tau \circ \varphi(F_D(u,v)) = \iota_n(\alpha \cdot (\varphi(F_D(u,v))) \tag{6}$$

Where α is a scaling factor which takes values in the set {0.2, 0.3, ... , 0.9}. ι_n is one of the eight isometries which include reflection, rotation[1]. But in DCT domain the transformation result changes. The changes are shown in table 1.

	Spatial Domain	DCT Domain
1	identity	$\iota_0(F(u,v)) = F(u.v)$
2	reflection about mid-vertical axis	$\iota_1(F(u,v)) = (-1)^V F(u,v)$
3	reflection about mid-horizontal axis	$\iota_2(F(u,v)) = (-1)^u F(u,v)$
4	reflection about first diagonal	$\iota_3(F(u,v)) = F(v,u)$
5	reflection about second diagonal	$\iota_4(F(u,v)) = (-1)^{u+v} F(v,u)$
6	rotation through +90°	$\iota_5(F(u,v)) = (-1)^u F(v,u)$
7	rotation through +180°	$\iota_6(F(u,v)) = (-1)^{u+v} F(u,v)$
8	rotation through -90°	$\iota_7(F(u,v)) = (-1)^v F(v,u)$

Table 1. The changes of the isometries from spatial domain to DCT domain.

The derivation process of the isometries' changes from spatial domain to DCT domain is similar to that presented by R.N.Bracewell,etc.[8]. As an example, we derive the second change.

The question can be rewritten as:

If a 8×8 block *f(x,y)* has 2D DCT *F(u,v)*, then *g(x,y)=f(x,7-y)* has 2D DCT $G(u,v) = (-1)^v F(u,v)$.

The derivation is as follows:

$$\begin{aligned} G(u,v) &= \frac{1}{16} c(u)c(v) \sum_{x=0}^{7}\sum_{y=0}^{7} g(x,y) \cos\frac{(2x+1)u\pi}{16} \cos\frac{(2y+1)v\pi}{16} \\ &= \frac{1}{16} c(u)c(v) \sum_{x=0}^{7}\sum_{y=0}^{7} f(x,7-y) \cos\frac{(2x+1)u\pi}{16} \cos\frac{(2y+1)v\pi}{16} \end{aligned}$$

Let $x_1 = x,\ y_1 = 7 - y$

$$\begin{aligned} G(u,v) &= \frac{1}{16} c(u)c(v) \sum_{x=0}^{7}\sum_{y=0}^{7} f(x_1,y_1) \cos\frac{(2x_1+1)u\pi}{16} \cos\frac{(2(7-y_1)+1)v\pi}{16} \\ &= \frac{1}{16} c(u)c(v) \sum_{x=0}^{7}\sum_{y=0}^{7} f(x_1,y_1) \cos\frac{(2x_1+1)u\pi}{16} \cos(v\pi - \frac{(2y_1+1)v\pi}{16}) \\ &= (-1)^v \cdot \frac{1}{16} c(u)c(v) \sum_{x=0}^{7}\sum_{y=0}^{7} f(x_1,y_1) \cos\frac{(2x_1+1)u\pi}{16} \cos\frac{(2y_1+1)v\pi}{16} \\ &= (-1)^v \cdot F(u,v) \end{aligned}$$

This finishes the proof.

The coding procedure of the high activity range block consists in searching for the most similar domain block among all the domain blocks to minimize the distortion. The distortion used is:

$$d = \sum_{\substack{u,v=0 \\ u,v\neq 0 \text{ at the same time}}}^{7} \left[F_R(u,v) - \tau \circ \varphi(F_D(u,v))\right]^2 \quad (7)$$

From (7), it is obvious that the DC coefficients $F_R(0,0), F_D(0,0)$ do not affect the matching procedure, but they indicate the luminance difference of the range block and the domain block, so the luminance offset Δg is a parameter needed to store.

$$\Delta g = F_D(0,0) - F_R(0,0) \quad (8)$$

When the most similar domain block is found, we must store $F_D(u,v)$, τ, φ, that is, we must store the co-ordinates of $F_D(u,v)$,.α, ι_n and Δg. The bit allocation is:

1. The co-ordinates of $F_D(u,v)$ 5+5=10 bits
2. α 3 bits
3. ι_n 3 bits
4. Δg 10 bits

Next, we calculate the deference image between the range block and its fractal approximation.

$$D(u,v) = F_R(u,v) - \tau \circ \varphi(F_D(u,v)) \quad (9)$$

For the difference image *D(u,v)*, quantize it first.

$$D_Q(u,v) = INTEG\left[\frac{D(u,v)}{Q(u,v)}\right] \quad (10)$$

$D_Q(u,v)$ is the quantized difference image coefficient. $Q(u,v)$ is a quantization table designed considering human visual properties. It is shown in Fig. 4.

24	17	15	24	36	60	77	92
18	18	21	29	39	87	90	83
21	20	24	36	60	86	104	84
21	26	33	44	77	131	120	93
27	33	56	84	102	164	155	116
74	96	117	131	155	182	180	152
108	138	143	147	168	150	155	149

Fig. 4. The quantization table

INTEG is a function which takes the integer nearest to its variable.

After quantization, most part of $D_Q(u,v)$ is zero. It is easy to design Huffman code according to the coefficients' probabilities. Finally, we store the Huffman code.

When every range block is encoded, the compression procedure is finished.

2.2 Decompression Procedure

The decoding scheme is shown in the following diagram:

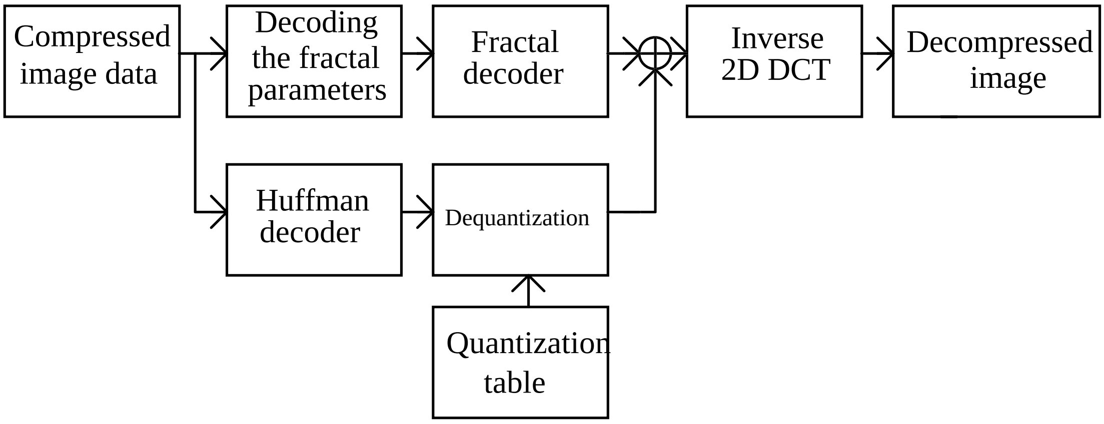

Fig. 5. The decompression procedure

The decompression procedure is relatively simple. First, we decode the fractal parameters and transform an arbitrary image repeatedly by fractals. It needs about 8 iterations to converge. This procedure is similar to that proposed by A. E. Jacquin[1]. The difference with Jacquin's method is that our method is carried out in DCT domain. This procedure produces a DCT domain fractal approximation of the original image; On the other hand, we decode the Huffman code and dequantize it, thus we get the difference image. At last, we transform the addition of the fractal approximation and the difference image by inverse 2D DCT and we get the decompressed image.

The 16×16 and 8×8 inverse 2D DCT formulae used in the decoding procedure are:

$$f_R(x,y) = \sum_{u=0}^{7}\sum_{v=0}^{7} c(u)c(v)F_R(u,v)\cos\frac{(2x+1)u\pi}{16}\cos\frac{(2y+1)v\pi}{16} \quad (11)$$

$$f_D(x,y) = \sum_{u=0}^{15}\sum_{v=0}^{15} c(u)c(v)F_D(u,v)\cos\frac{(2x+1)u\pi}{32}\cos\frac{(2y+1)v\pi}{32} \quad (12)$$

where $c(u),c(v) = \begin{cases} 1/\sqrt{2} & \text{u, v = 0} \\ 1 & \text{otherwise} \end{cases}$

3. Experimental Result

We have compressed a 256×256×8 standard test image "Lena" using the method proposed in the paper. Comparatively, we also compressed the image with the spatial domain fractal coding scheme proposed in [1] and the DCT domain fractal coding scheme[7]. We choose compression ratio (CR) and peak-to-peak signal-to-noise ratio (PSNR) as criterion of comparison. Their definitions are:

$$CR = \frac{B_{orig}}{B_{comp}} \tag{13}$$

$$PSNR = 10\log_{10}\left[255^2 \Big/ \frac{1}{k}\sum_{i=1}^{k}(x_i - c_i)^2 \right] \tag{14}$$

Where B_{orig} denotes the bit number needed to represent the original image, B_{comp} denotes the bit number needed to represent the compressed image. x_i is the pixel of the original image and c_i is the pixel of the decompressed image. *K* denotes the total number of pixels. Table 3 summarizes the experimental results:

Compression Method	CR	PSNR(dB)
The hybrid method presented in the paper	12.4	31.7
Spatial domain fractals presented in[1]	17.4	24.9
DCT domain fractals presented in[7]	18.5	26.1

Table 3. The performance of three fractal-based image coding methods

The reconstructed images of the three methods are shown in Fig. 6.

Fig. 6. (a) The original image
(b) Reconstructed by the method presented in the paper
(c) Reconstructed by the spatial domain fractals
(d) Reconstructed by DCT domain fractals.

Reference

[1] A.E.Jacquin, "A novel fractal block-coding technique for digital images", *Proc., ICASSP*, 1990, pp 2225-2228

[2] J M Beaumont, "Image data compressing using fractal techniques", *BT Technol .J*, Vol. 9, No. 4, 1991, pp 93-109

[3] Roger Dettmer, "Fractal-based image compression", *IEE Review*, Sept.,1992, pp 323-327

[4] E.W.Jacobs,Y.Fisher and R.D.Boss, "Image compression: A study of the transform method", *Signal Processing*, No. 29,1992, pp 251-263

[5] D. M. Monro and F.Dudbridge, "Fractal block coding of images", *Electronics Letters*, Vol. 28, No. 11, 1992, pp 1053-1055

[6] D.M.Monro, "Class of fractal transforms", *Electronics Letters*, Vol. 29, No. 4, 1993, pp 362-363

[7] Y. Zhao and B. Yuan, "A DCT domain fractals for image coding", *Proc. of Inter. Conference on Communication Technology*, 1994.

[8] R.N.Bracewell, K.-Y,Chang,A.K. and Y.-H.Wang, "Affine theory for two-dimensional fourier transform", *Electronics Letters*, Vol. 29, No. 3, 1993, pp 304