Partial iterated function system based fractal image coding

Zhou WANG and Yinglin YU

Institute of Radio and Automation
South China University of Technology
Guangzhou 510641, P. R. China

ABSTRACT

A recent trend in computer graphics and image processing has been to use Iterated Function System (IFS) to generate and describe images. Barnsley et al. presented the conception of fractal image compression and Jacquin was the first to propose a fully automatic gray scale still image coding algorithm. This paper introduces a generalization of basic IFS, leading to a conception of Partial Iterated Function System (PIFS). A PIFS operator is contractive under certain conditions and when it is applied to generate an image, only part of it is actually iteratedly applied. PIFS provides us a flexible way to combine fractal coding with other image coding techniques and many specific algorithms can be derived from it. On the basis of PIFS, we implement a partial fractal block coding (PFBC) algorithm and compare it with basic IFS based fractal block coding algorithm. Experimental results show that coding efficiency is improved and computation time is reduced while image fidelity does not degrade very much.

Keywords: image coding, data compression, fractal, iterated function system (IFS)

1. INTRODUCTION

Since Barnsley et al. presented the conception of fractal image compression based on Iterated Function System (IFS)[1] and Jacquin proposed a fully automatic gray scale still image coding algorithm,[2,3] the interest in fractal image coding has been steadily growing. Jacquin's algorithm has been studied and improved in recent years [e.g., 4,5,6]. We briefly describe the basic IFS based image coding algorithm as follows which is equivalent to those in most previously published work.

A gray scale image can be described as a real valued function $\phi:\Omega \rightarrow I$. Here, $\Omega \subset \mathcal{R}^2$ ($\mathcal{R}$ denotes the set of real number) represents the underlying space and $I=[I_{\min}, I_{\max}] \subset \mathcal{R}$ is a real interval representing the possible intensity values within the image (for example, for 8bits per pixel (bpp) gray scale images as discussed in this paper, it is $[0,255]$). The set of all these functions can be converted into a complete metric space (Y, d) by defining the distance between two images as the l^{∞} metric:

$$\forall \phi_1,\phi_2 \in Y \qquad d(\phi_1,\phi_2) = \sup_{(x,y)\in\Omega} \left|\phi_1(x,y) - \phi_2(x,y)\right| \tag{1}$$

Ω is partitioned into non-overlapping subsets, such that

$$\begin{cases} \Omega = \bigcup_{i=1}^{N} R_i & R_i \subset \Omega \quad (i = 1,2,\ldots N) \\ R_i \cap R_j = \varnothing & (i \neq j) \end{cases} \tag{2}$$

Each R_i corresponds to a geometrical transformation $g_i: D_i \rightarrow R_i$ ($D_i \subset \Omega$, $i=1,2,\ldots N$) and a luminance transformation v_i: $\mathcal{R} \rightarrow \mathcal{R}$ ($i=1,2,\ldots N$). g_i is combined with v_i to form the mapping

$$w_i: \boldsymbol{D}_i \rightarrow \boldsymbol{R}_i \qquad (\text{here } \boldsymbol{R}_i = R_i \times I\,,\ \boldsymbol{D}_i = D_i \times I)$$

$$\forall (x,y,z) \in \boldsymbol{D}_i \qquad w_i(x,y,z) = (g_i(x,y), v_i(z))$$

0-8194-2132-4/96/$6.00

In practice, the geometrical transformation g_i is an affine linear transformation that combines a spatial contraction and a position shift that maps D_i (often block) to the position of R_i (often block). The luminance transformation v_i is set to:

$$v_i: \; z_R = a_i z_g + b_i$$

where z_R and z_g are the intensities of the pixels within R_i and $g_i(D_i)$ respectively. a_i called scale factor and b_i called translation term are chosen to minimize

$$e_i = \int_{(x,y)\in R_i} [\psi_R(x,y) - a_i\psi_g(x,y) - b_i]^2 dxdy$$

where $\psi_R(x,y)$ and $\psi_g(x,y)$ are intensities of the pixels extracted or calculated from the original image ψ. We can obtain a_i and b_i by solving the equations $\frac{\partial e_i}{\partial a_i} = 0$ and $\frac{\partial e_i}{\partial b_i} = 0$. We get

$$\begin{cases} a_i = \dfrac{\overline{\psi_R \psi_g} - \overline{\psi_R} \cdot \overline{\psi_g}}{\overline{\psi_g^2} - (\overline{\psi_g})^2} \\[2ex] b_i = \overline{\psi_R} - a_i \overline{\psi_g} \end{cases}$$

where " $\overline{}$ " denotes the average operator.

The combination of all w_i 's results in a basic Iterated Function System (IFS) operator

$$W: Y \to Y$$

$$W(\phi) = \bigcup_{i=1}^{N} w_i(D_i \cap \phi)$$

$$\forall (x,y) \in \Omega \quad (W\phi)(x,y) = v_i(\phi(g_i^{-1}(x,y))) \quad when \quad (x,y) \in R_i$$

W is a contractive mapping in space *(Y, d)*, provided all v_i 's are contractive. That is

if $\qquad \exists 0 < s < 1, \quad \forall i \in [1, N], \quad \forall z_1, z_2 \in \mathcal{R}, \quad |v_i(z_1) - v_i(z_2)| < s \cdot |z_1 - z_2| \qquad$ (3)

then $\qquad \forall \phi_1, \phi_2 \in Y \qquad d(W(\phi_1), W(\phi_2)) \le s \cdot d(\phi_1, \phi_2)$

where *s* is called the contractive factor of *W*. According to the *Fixed Point Theorem* of complete metric space, *W* possesses the following properties:

(*a*) There exists a unique attractor image $\phi \in Y$, such that

$$W(\phi) = \phi$$

(*b*) To compute the attractor image ϕ:

$$\forall \phi^{(0)} \in Y, \quad \lim_{n\to\infty} W^{\circ n}(\phi^{(0)}) = \phi$$

(*c*) Collage theorem estimate:

$$\forall \psi \in Y, \quad d(\phi, \psi) \le \frac{d(\psi, W(\psi))}{1-s}$$

According to the theorems above, if we want to code an image ψ, then the encoding procedure is to find an IFS operator *W*, such that application of *W* to ψ does not change ψ very much. The operator *W* usually can be represented by fewer bits, thus the image data of ψ is compressed. When decoding, the IFS operator *W* is iteratedly applied to any initial image $\phi^{(0)}$, so that a sequence of images $\phi^{(1)}$, $\phi^{(2)}$, $\cdots$, $\phi^{(n)}$, $\cdots$ is generated and we can obtain a unique attractor image ϕ (reconstructed image) in the end.

2. PARTIAL ITERATED FUNCTION SYSTEM

When applying the basic IFS based algorithm to real world images, we find it is difficult to code some kinds of regions efficiently. For example, a very smooth region can be represented by a uniformly gray region with gray level equal to the mean value of the pixels within the region and usually 6-8 bits are enough. However, to code this region by a fractal mapping, several parameters including horizontal and vertical geometrical shifts, intensity scale factor and translation term should be coded. It is difficult to represent the codes for these parameters by fewer than 8bits. For the above reason, coding these kinds of regions using some other more adequate approaches may be a better choice. Then a problem arouses: The fractal codes for different regions are highly correlative to each other. If some of these codes are replaced by other kinds of codes, can the remaining fractal codes still be effective? We try to solve this problem by introducing a new conception of Partial Iterated Function System (PIFS).

To code only part of the image by fractal transform, we modify the partition of Ω (2) as:

$$\begin{cases} \Omega = G \bigcup H & G \bigcap H = \varnothing \\ G = \bigcup_{i=1}^{M} R_i & R_i \subset G \quad (i = 1,2,\cdots M) \\ R_i \bigcap R_j = \varnothing & (i \neq j) \end{cases}$$

Where G is to be coded by fractal transform and H is to be coded by other approaches. Let $\boldsymbol{G} = G \times I$, $\boldsymbol{H} = H \times I$. We define the PIFS operator as

$$W^P : Y \rightarrow Y$$

$$W^P(\phi) = (\bigcup_{i=1}^{M} w_i(\boldsymbol{D}_i \bigcap \phi)) \bigcup (C(\boldsymbol{H} \bigcap \phi))$$

Where the structures of w_i 's are the same as those having been described for basic IFS operators. C is a mapping applied to H part of the image. Particularly, when $H = \varnothing$, W^P and W are the same. Therefore, PIFS can be viewed as a generalization of basic IFS. If C is a constant function mapping, such that

$$\forall \phi \in Y \qquad C(\boldsymbol{H} \bigcap \phi) = \boldsymbol{H} \bigcap c \tag{4}$$

(where $c \in Y$ is a constant image), then the function form of W^P become

$$\forall (x,y) \in \Omega \qquad (W^P \phi)(x,y) = \begin{cases} v_i(\phi(g_i^{-1}(x,y))) & \text{when} \quad (x,y) \in R_i \subset G \\ c(x,y) & \text{when} \quad (x,y) \in H \end{cases}$$

Theorem1: *In l^∞ metric space (Y,d) (the l^∞ metric is defined as (1)), if there exists a contractive factor s (0<s<1) for a PIFS operator W^P, then W^P is a contractive mapping.*

Proof:

$\forall (x,y) \in \Omega$,

if $(x,y) \in D_i \subset G$, then

$$\begin{aligned} &\left|(W^P \phi_1)(x,y) - (W^P \phi_2)(x,y)\right| \\ &= \left|v_i(\phi_1(w_i^{-1}(x,y))) - v_i(\phi_2(w_i^{-1}(x,y)))\right| \\ &\overset{(3)}{<} s \cdot \left|\phi_1(w_i^{-1}(x,y)) - \phi_2(w_i^{-1}(x,y))\right| \\ &\leq s \cdot \sup_{(x,y) \in \Omega} \left|\phi_1(x,y) - \phi_2(x,y)\right| \end{aligned}$$

if $(x,y) \in H$, then

$$\left|(W^P \phi_1)(x,y) - (W^P \phi_2)(x,y)\right|$$

$$= |c(x,y) - c(x,y)|$$
$$= 0$$
$$\le s \cdot \sup_{(x,y)\in\Omega} |\phi_1(x,y) - \phi_2(x,y)|$$

$\therefore$
$$\sup_{(x,y)\in\Omega} \left|(W^P\phi_1)(x,y) - (W^P\phi_2)(x,y)\right| \le s \cdot \sup_{(x,y)\in\Omega} |\phi_1(x,y) - \phi_2(x,y)|$$

that is
$$d(W^P(\phi_1), W^P(\phi_2)) \le s \cdot d(\phi_1, \phi_2) \ \#$$

Also according to the *Fixed Point Theorem* of complete metric space, W^P still possesses the properties (*a*), (*b*) and (*c*). Similar to having been discussed for basic IFS operator *W*, [4] the condition which is sufficient to ensure contractivity for l^∞ metric may be sufficient to ensure only eventual contractivity for l^2 metric:

$$\forall \phi_1, \phi_2 \in Y \qquad d(\phi_1, \phi_2) = \Big(\sum_{(x,y)\in\Omega} |\phi_1(x,y) - \phi_2(x,y)|^2 \Big)^{1/2}$$

Actually, eventual contractivity is sufficient to ensure (*a*) and (*b*) but not (*c*).

Theorem2: *In l^2 metric space (Y,d), there exists a unique image $\phi \in Y$, such that*
$$W^P(\phi) = \phi$$

Proof:
Existence:

$\because$ there exists a unique image ϕ in l^∞ metric space, such that
$$W^P(\phi) = \phi$$

that is
$$\sup_{(x,y)\in\Omega} \left|(W^P\phi)(x,y) - \phi(x,y)\right| = 0$$

$\therefore$ $\forall (x,y) \in \Omega$
$$\left|(W^P\phi)(x,y) - \phi(x,y)\right| = 0$$

for l^2 metric space,
$$d(W^P(\phi), \phi) = \Big(\sum_{(x,y)\in\Omega} \left|(W^P\phi)(x,y) - \phi(x,y)\right|^2 \Big)^{1/2} = 0$$

$\therefore$
$$W^P(\phi) = \phi$$

That is to say, W^P has the same attractor both in l^∞ and l^2 metric spaces.

Uniqueness:
if there exist different ϕ_1 and ϕ_2, such that
$$W^P(\phi_1) = \phi_1 \qquad \text{and} \qquad W^P(\phi_2) = \phi_2$$

then
$$d(W^P(\phi_1), \phi_1) = \Big(\sum_{(x,y)\in\Omega} \left|(W^P\phi_1)(x,y) - \phi_1(x,y)\right|^2 \Big)^{1/2} = 0$$

$\therefore$ $\forall (x,y) \in \Omega$
$$\left|(W^P\phi_1)(x,y) - \phi_1(x,y)\right| = 0$$
$$\sup_{(x,y)\in\Omega} \left|(W^P\phi_1)(x,y) - \phi_1(x,y)\right| = 0$$

$\therefore$ in l^∞ metric space, $\quad W^P(\phi_1) = \phi_1$

for the same reason, in l^∞ metric space, $\quad W^P(\phi_2) = \phi_2$

This is contradictory to that W^P only has one attractor in l^∞ metric space (property (*a*)). #

Theorem3: *In l^2 metric space (Y,d), to compute the unique attractor image ϕ :*

$$\forall \phi^{(0)} \in Y, \qquad \lim_{n\to\infty} (W^P)^{\circ n}(\phi^{(0)}) = \phi$$

Proof:

For l^∞ metric space, according to property (b)

$$\exists N \qquad \forall n > N \qquad \sup_{(x,y)\in\Omega} \left|(W^P)^{\circ n}(\phi^{(0)})(x,y) - \phi(x,y)\right| < \varepsilon \tag{5}$$

then for l^2 metric space

$$\begin{aligned}
& d((W^P)^{\circ n}(\phi^{(0)}),\phi) \\
&= \Big(\sum_{(x,y)\in\Omega} \left|(W^P)^{\circ n}(\phi^{(0)})(x,y) - \phi(x,y)\right|^2\Big)^{1/2} \\
&\le \Big(\sum_{(x,y)\in\Omega} \big(\sup_{(x,y)\in\Omega} \left|(W^P)^{\circ n}(\phi^{(0)})(x,y) - \phi(x,y)\right|\big)^2\Big)^{1/2} \\
&\overset{(5)}{<} \varepsilon \cdot \Big(\sum_{(x,y)\in\Omega} 1\Big)^{1/2} \\
&= C\cdot\varepsilon \qquad (C \text{ is a constant})
\end{aligned}$$

$$\therefore \qquad \lim_{n\to\infty} (W^P)^{\circ n}(\phi^{(0)}) = \phi \quad \#$$

All the conditions for the above theorems are sufficient but not necessary conditions. In practice, the conditions usually are not strictly fulfilled. Actually, even some of the v_i 's can be non-contractive.[4] However, we still get many "experimentally contractive" operators.

3. IMAGE CODING ALGORITHM

According to the construction and properties of PIFS operator, we develop a general PIFS based image encoding/decoding system which is shown in Fig. 1. To encode an original image, we partition it into two parts, G and H, which are to be coded using fractal coding and other techniques respectively. The codes for G and H, together with the codes for partition information are stored or transmitted. When decoding, the partition information is decoded first. Then is H. In the end, W^P is iteratedly applied to generate the reconstructed image. For each iteration,

$$\begin{aligned}
\phi^{(n)} &= W^P(\phi^{(n-1)}) \\
&= \Big(\bigcup_{i=1}^{M} w_i(\boldsymbol{D}_i \cap \phi^{(n-1)})\Big) \cup \big(C(\boldsymbol{H} \cap \phi^{(n-1)})\big) \\
&\overset{(4)}{=} \Big(\bigcup_{i=1}^{M} w_i(\boldsymbol{D}_i \cap \phi^{(n-1)})\Big) \cup \big(C(\boldsymbol{H} \cap c)\big) \qquad n=1,2,\ldots
\end{aligned}$$

Notice that only the first part of $\phi^{(n)}$ (the codes for G part) is actually iteratedly applied in each iteration. The iteration procedure stops at the nth iteration when we are satisfied with the image $\phi^{(n)}$ which is an approximation of the attractor image ϕ.

The general system offers much flexibility for practical implementation.

- The flexibility of partition standard and algorithm.

• The flexibility of choosing which technique to code H part of the image.

• The flexibility of partition information coding.

The flexibility leads to many specific image encoding/decoding algorithms. An example is a fractal coding and bi-linear interpolation combined algorithm.[6] We implement another algorithm where an adaptive quadtree segmentation algorithm is employed. We describe it as follows.

The whole image ($2^D \times 2^D$) is segmented into 4 subblocks ($2^m \times 2^m$, $m=D-1$). For each subblock, we compute its mean value M and the mean square error (*MSE*) between the original block and the uniformly gray block with gray level equal to M. *MSE* is compared with a threshold T_m. If $MSE \le T_m$, the block is considered as a smooth block and M is quantified and coded. If $MSE > T_m$, we try to approximate this block using fractal mapping. If the *MSE* between the original block and the fractal mapping block is not greater than T_m, the block is considered to be suitable for fractal coding and its fractal codes are stored. Otherwise, the block is partitioned into 4 $2^{m-1} \times 2^{m-1}$ subsubblocks. The processing of each subsubblock is the same as its parent block. This procedure continues until a minimal block size threshold is reached. The partition information is coded using Huffman coding. For each partition stage m, the *MSE* threshold T_m can be adjusted to make a compromise between compression ratio and image fidelity. In our practical algorithm, the whole image is directly partitioned into 64×64 or 32×32 blocks and segmentation stops when block sizes of 4×4 or 2×2 are reached.

We use peak-signal-to-noise (*PSNR*) to determine image fidelity which is defined as:

$$PSNR = 10\log_{10} \frac{(2^k - 1)^2}{\frac{1}{r^2}\sum_{i=1}^{r}\sum_{j=1}^{r}(I_{ij} - I'_{ij})^2}$$

where k is the number of bits per pixel (bpp) of the image and r is the number of the size of the image. I_{ij} and I'_{ij} are the intensities of the pixels at the position (i, j) within the original image and the test image respectively. For example, for the image "Lena" 512×512, 8bpp, k=8 and r=512 (see Fig. 2).

Our algorithm can be viewed as a representative of PIFS based partial fractal block coding (PFBC) algorithms. Table 1 and Fig. 3 show some coding results of basic IFS based basic fractal block coding algorithm (basic FBC) and PFBC algorithm. It appears that by using PFBC technique, the bit rates are reduced by more than half and the coding time is reduced by about 60~75% while the peak signal-to-noise ratios (*PSNR*s) do not degrade very much. Much time is saved because only part of the image is coded using fractal approximation which dominates most of the coding time. Since some regions are coded by more suitable approaches, coding efficiency is improved.

4. CONCLUSIONS AND EXTENSIONS

In this paper, we present a conception of partial iterated function system (PIFS) which is a generalization of basic IFS. Some properties of PIFS such as contractivity are discussed. A general PIFS based image encoding/decoding system is also introduced. Theoretical analysis and experimental results show that PIFS based algorithm has some advantages over basic IFS based algorithm for improvement of coding efficiency and reduction of computation time.

We think PIFS based fractal image coding is very promising because on the basis of PIFS, most other image coding techniques can be easily and flexibly combined with fractal image coding. The improvement may be in the following directions:

• The improvement of fractal image coding and other image coding techniques themselves. Any improvement in either fractal coding or other techniques can also be effective in the combined algorithm.

• Good standards to determine which kinds of regions are suitable for fractal coding and which are not. This is a difficult but very important problem because a good classification of image regions is the premise of taking full advantage of PIFS.

• The improvement of partition information coding. This is related to partition standard and algorithm. Arithmetic coding may be a good choice.

• The fractal and non-fractal coded regions may make use of information from each other, so that both of them can be coded more efficiently.

5. ACKNOWLEDGMENT

This work is supported by National Natural Science Foundation of China, National "Climbing" Project of China and Natural Science Foundation of Guangdong province.

6. REFERENCES

1. M. F. Barnsley, A. D. Sloan, "A better way to compress images", *BYTE Magazine*, vol. 13, no.1, pp.215-223, Jan. 1988.
2. A. E. Jacquin, "A fractal theory of iterated Markov operators with applications to digital image coding", Ph. D. Dissertation, Georgia Institute of Technology, Aug. 1989.
3. A. E. Jacquin, "Image coding based on a fractal theory of iterated contractive image transformations", *IEEE Transaction on Image Processing*, vol. 1, no. 1, pp.18-30, Jan. 1992.
4. E. W. Jacobs, Y. Fisher, R. D. Boss, "Image compression: a study of the iterated transform method", *Signal Processing*, vol. 29, no. 3, pp.251-263, Dec. 1992.
5. Yudong Fang, Yinglin Yu, "A quick fractal image compression coding method", to appear in *Acta Electronica Sinica*, vol. 24, no. 1, pp.29-34, Jan. 1996.
6. Zhou Wang, Yinglin Yu, "A novel fractal image coding approach", to appear in *Journal of China Institute of Communications*

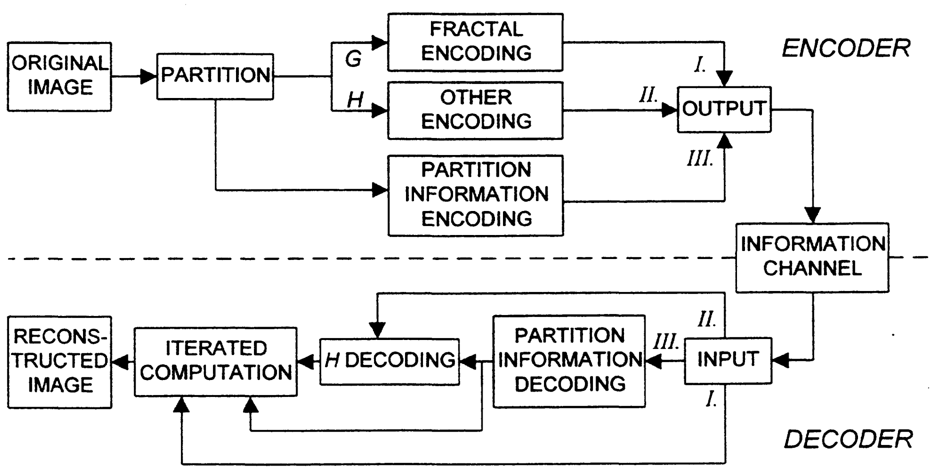

Fig. 1. PIFS based image encoding/decoding system

Table 1. Comparison of coding results by basic FBC and PFBC. Test image: "Lena"(512x512, 8bpp).

Coding Algorithm	Bit Rate(bpp)	PSNR(dB)	Relative Coding Time
basic FBC	0.391	30.1	1
PFBC	0.194	29.8	0.39
	0.178	29.5	0.33
	0.143	28.9	0.26

Fig. 2. Original image "Lena" (512x512, 8bpp)

(a) *(b)*

Fig. 3. Reconstructed image. (a) by basic FBC, 0.391bpp, 30.1dB; (b) by PFBC, 0.178bpp, 29.5dB