A Fast Fractal Image Coding Based on Kick-Out and Zero Contrast Conditions

Cheung-Ming Lai, Kin-Man Lam, *Member, IEEE*, and Wan-Chi Siu, *Senior Member, IEEE*

***Abstract*—A fast algorithm for fractal image coding based on a single kick-out condition and the zero contrast prediction is proposed in this paper. The single kick-out condition can avoid a large number of range-domain block matches when finding the best matched domain block. An efficient method for zero contrast prediction is also proposed, which can determine whether the contrast factor for a domain block is zero or not, and compute the corresponding difference between the range block and the transformed domain block efficiently and exactly. The proposed algorithm can achieve the same reconstructed image quality as the exhaustive search, and can greatly reduce the required computation or runtime. In addition, this algorithm does not need any pre-processing step or additional memory for its implementation, and can combine with other fast fractal algorithms to further improve the speed. Experimental results show that the runtime is reduced by about 50% of that of the exhaustive search method. When combined with the DCT Inner Product algorithm, the required runtime for the algorithm can be further reduced by about 50%. The proposed algorithm was also compared to two other fast fractal algorithms. Experimental results also show that our algorithm achieves a better efficiency and requires a much smaller amount of memory for implementation.**

***Index Terms*—Fractal image coding, full search, image compression, zero contrast.**

I. INTRODUCTION

A SIGNIFICANT amount of research work has recently been done on fractal image compression. Fractal image coding can provide a highly reconstructed image quality with a high compression ratio (CR), is independent of resolution, and has a fast decoding process. Fractal theory was first presented by Barnsley, and is based on a mathematical theory called Iterated Function Systems (IFS). Jacquin [4] proposed the first practical fractal image compression scheme, which relies on the assumption that image redundancy could be efficiently exploited through self-transformability on a block-wise basis. Fractal image compression is based on the representation of an image by a set of iterated contractive transformations for which the reconstructed image is an approximate fixed point and close to the original image.

The problem with fractal coding is the high computational complexity in its encoding process. Most of the encoding time is spent on finding the best matched domain block from a large domain pool to represent an input range block with respect to contrast and intensity offset, as well as the isometry transformations. By minimizing the difference between the range block and the transformed domain block, the corresponding iterated contractive transformation τ_i can be found for the range block. The exhaustive search algorithm can obtain the optimal result by searching exhaustively all the blocks within the domain pool, but this process requires a high computational cost, which limits its practical application. To solve this problem, extensive research on fast fractal image encoding algorithms [1], [3], [5]–[10], [12],[13], [15] has been carried out. In particular, Bani-Eqbal [13] proposed a tree search method, which devised an incremental procedure to bind the domain block pixels, and then arranged the domain blocks in a tree structure to direct the search. In [12], an adaptive search algorithm based on an adaptive necessary condition to reduce the computational complexity was proposed. However, these techniques can reduce the required computation only at the expense of additional memory and degradation of the reconstructed image quality. So, some efficient algorithms [2], [11], [14] have been developed to alleviate the computation burden, while the image quality can be maintained at the level of that of the full search. However, these algorithms first need to transform the image blocks into the frequency domain. In other words, pre-processing is required.

In this paper, we propose an efficient algorithm based on a single kick-out condition and zero contrast prediction, which can greatly reduce the required computation as compared to the exhaustive search, while maintaining the same reconstructed image quality. Our proposed kick-out condition can determine efficiently whether a domain block is a good representation of a range block, and so excessive computation can be avoided in the early stage. With zero contrast prediction, the computation involved is further reduced. In addition, the algorithm does not need any pre-processing and extra memory for its implementation. Our proposed approach can also be combined with other fast encoding methods, such as the DCT Inner Product [2] method, to further speed up the encoding time. Experimental results show that the runtime can be reduced by about 75% when our algorithm is combined with the DCT Inner Product algorithm [2].

This paper is organized as follows. We briefly describe the fundamentals of the fractal coding algorithm and review the DCT Inner Product approache in Section II. The DCT approach can be combined with our proposed scheme to further reduce the computational complexity. Section III presents our new fractal image compression algorithm based on a kick-out condition and

---

Manuscript received April 3, 2002; revised April 30, 2003. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Fernando M. B. Pereira.

C.-M. Lai and W.-C. Siu are with the Centre for Multimedia Signal Processing, Department of Electronic and Information Engineering, The Hong Kong Polytechnic University, Hong Kong.

K.-M. Lam is with the Centre for Multimedia Signal Processing, Department of Electronic and Information Engineering, The Hong Kong Polytechnic University, Hong Kong (e-mail: enkmlam@polyu.edu.hk).

Digital Object Identifier 10.1109/TIP.2003.817246

zero contrast prediction. In Section IV, we compare the performance of our proposed fast algorithm with the full search, the adaptive search scheme [12], tree search scheme [13], DCT Inner Product algorithm [2], and our algorithm combined with the DCT algorithm. Finally, conclusions are given in Section V.

II. REVIEW OF FRACTAL IMAGE COMPRESSION

In the fractal image compression scheme, an image, f_{orig}, of size $r \times r$ is partitioned into two basic block units: the range blocks and the domain blocks. The range blocks $\Re$ are a set of nonoverlapping image blocks of size $k = n \times n$, which are denoted as $\{R_i\}_{i=1}^{N_R} = \{r_{i1}, r_{i2}, \ldots, r_{ik}\}_{i=1}^{N_R}$. The number of range blocks is $N_R = (r/n) \times (r/n)$, and image f_{orig} is a union of $\{R_i\}_{i=1}^{N_R}$

$$f_{orig} = \bigcup_{i=1}^{N_R} R_i. \tag{1}$$

Overlapping image blocks of f_{orig} in a domain pool with size larger than that of the range blocks are called domain blocks. These domain blocks can be obtained by sliding a window of size $l = m \times m$, where $m > n$, throughout the image to construct the domain pool. The size of a domain block is usually four times that of a range block, i.e., $l = 2n \times 2n$. To encode a range block R, each of the blocks in the domain pool is scaled to the size of the range block, and is then compared to R with respect to intensity offset and contrast parameters, as well as the isometry transformations. The set of contracted domain blocks is denoted as $\{D_i\}_{i=1}^{N_D} = \{d_{i1}, d_{i2} \ldots, d_{il}\}_{i=1}^{N_D}$, where N_D is the number of domain blocks in the domain pool. The corresponding parameters for the affine transformation τ are determined by minimizing the following equation:

$$E(R, D_i) = \|R - (s.D_i + oI)\|^2 \quad \text{where } o, s\epsilon\Re \tag{2}$$

where D_i is the contracted domain block under isometry transformation, I denotes a unity vector of dimension k, and s and o are the contrast and offset parameters, respectively. For a given range block and the corresponding domain block, these two parameters are given as follows:

$$s = \frac{\langle R, D\rangle - \frac{1}{k}\langle R, I\rangle \langle D, I\rangle}{\|D\|^2 - \frac{1}{k}\langle D, I\rangle^2} \text{ and } o = \frac{1}{k}\left(\langle R, I\rangle - s\langle D, I\rangle\right). \tag{3}$$

The $\|\,\|$ is the two-norm and $\langle\cdot,\cdot\rangle$ is inner product. The contrast factor should be $-1 < s < 1$ to ensure the contractivity of the transformation. The domain block which results in the smallest difference from (2) is then chosen as the best matched block, and the corresponding parameters for the transformations $\{\tau_i | i = 1, 2, \ldots, N_R\}$ are encoded and stored. At the decoding phase, the transformation parameters are recursively applied to an arbitrary initial image, which will then converge to the fractal image after fewer than 10 iterations.

An efficient fractal image encoding algorithm was proposed by Truong [2], which can produce the same image quality as an exhaustive search. In this method, the image blocks are demeaned, and the error function (2) between a range block and a transformed domain block can be simplified as follows:

$$E(R, D) = \|R - \bar{r}I\|^2 - \frac{\left\langle R - \bar{r}I, D - \bar{d}I\right\rangle^2}{\left\|D - \bar{d}I\right\|^2} \tag{4}$$

where $\bar{r}$ and $\bar{d}$ are the means of the range block and domain block, respectively. The most computational part of (4) is the inner product $\left\langle R - \bar{r}I, D - \bar{d}I\right\rangle$. To determine the best matched domain block for a range block, the isometry transformation consisting of four orientations and four reflections of each domain block must be considered. In other words, the error function has to be computed eight times; once for each of the transformed domain blocks. Most of the computational cost of this method comes from the overhead for calculating the inner product of the range block and the transformed domain block in computing the error function. In order to reduce computation involved, [2] proposed using Discrete Cosine Transform (DCT) to convert the image block to the frequency domain, which can reduce the number of computations of the inner product from eight to two. The other inner products can be obtained by a proper arrangement of these two inner products.

III. PROPOSED ALGORITHM

The principle of our proposed algorithm is to bypass those domain blocks that satisfy a kick-out condition, so they will no longer be considered and no further computations will be needed. In our algorithm, we first convert the full search (2) from two parameters, i.e., the contrast s and the offset o, to a function which only contains the contrast s. Based on this formulation, we can successively eliminate the search space in the domain pool and thus decrease the computation required to compare a range block and a transformed domain block.

A. Kick-Out Condition

From (4), the error function can be further simplified as follows:

$$\begin{aligned} E(R, D) &= \left[\|R\|^2 - \frac{1}{k}\langle R, I\rangle^2\right] - \frac{\left[\langle R, D\rangle - \frac{1}{k}\langle R, I\rangle\langle D, I\rangle\right]^2}{\left[\|D\|^2 - \frac{1}{k}\langle D, I\rangle^2\right]} \\ &= \left[\|R\|^2 - \frac{1}{k}\langle R, I\rangle^2\right] - \frac{\left[\langle R, D\rangle - \frac{1}{k}\langle R, I\rangle\langle D, I\rangle\right]^2}{\left[\|D\|^2 - \frac{1}{k}\langle D, I\rangle^2\right]^2} \\ &\quad \times \left[\|D\|^2 - \frac{1}{k}\langle D, I\rangle^2\right]. \end{aligned} \tag{5}$$

As $[s = \langle R, D\rangle - (1/k)\langle R, I\rangle\langle D, I\rangle]/[\|D\|^2 - (1/k)\langle D, I\rangle^2]$, the error function $E(R, D)$ can be written as follows:

$$E(R, D) = A - s^2 B \tag{6}$$

where $A = \|R\|^2 - (1/k)\langle R, I\rangle^2$ and $B = \|D\|^2 - (1/k)\langle D, I\rangle^2$.

The coefficient s is limited to the range (−1,1) to ensure convergence in the decoding process. If A is greater than B, the maximum error occurs when $s = 0$ while minimum error occurs when $s = 1$. Therefore, we have:

If $A - B \geq 0$, then

1) The maximum error occurs when $s = 0$,

$$E_{\max} = A - s^2 B = A \tag{7}$$

2) The minimum error occurs when $s = \pm 1$,

$$E_{\min} = A - s^2 B = A - B. \tag{8}$$

This means that, in finding the best matched domain block, the search is performed only if the minimum error $E_{\min}$ for the domain block under consideration is less than the current minimum error $d_{\min}$. Thus, we propose the kick-out condition as follows:

$$E_{\min} = A - B \geq d_{\min}. \tag{9}$$

Based on (9), we propose a fast search algorithm which can reject dissimilar domain blocks efficiently for a given range block. In our algorithm, we select the domain blocks from left to right and top to bottom. The matching errors of the first domain block D_1 with each of the eight isometry operations are calculated. The one with the minimum error is considered to be the initial best matched domain block. The current minimum distance $d_{\min}$ is then set to this minimum distortion, say $E(R, D_1)$, and the search proceeds in a raster scan order. To determine whether the next candidate domain block D_2 is closer to R than the current best match D_1, we compute $E_{\min}(R, D_2)$ and compare it to $d_{\min}$. If $E_{\min}(R, D_2)$ is larger than or equal to $d_{\min}$, it also means that the condition $E(R, D_2) \geq d_{\min}$ is always guaranteed. The domain block D_2 is therefore rejected. Otherwise, the actual distortion $E(R, D_2)$ is calculated for the domain block D_2 with the eight isometry operations, and is compared to $d_{\min}$. If all the matching errors between the transformed domain blocks of D_2 and R are larger than the current minimum error $d_{\min}$, D_2 is rejected for the same reason mentioned above. Otherwise, $d_{\min}$ is replaced by $E(R, D_2)$ and the current best matched domain block is set to D_2 with the corresponding isometry operation. This process is repeated for all the domain blocks D_i in the domain pool to find the best matched one for an input range block. Based on this kick-out condition, the required computation for searching the best matched domain block will be greatly reduced.

B. Fast Error Calculation Using Zero Contrast Prediction

In the implementation, the contrast factor s is encoded using 5 bits. Therefore, any value of the contrast s falling within ($-0.031\,25$, $0.031\,25$) will be set to zero after quantization. With (6), as $|s| < 1$, this means that the zero contrast condition will happen only when $A < B$

$$E(R, D) = A - s^2 B \geq 0 \quad \text{or} \sqrt{\frac{A}{B}} \geq |s|. \tag{10}$$

The contrast factor s is quantized to 0 if the absolute value of s is less than 0.031 25, i.e.,

$$0.031\,25 > \sqrt{\frac{A}{B}} \geq |s|. \tag{11}$$

TABLE I
COMPARISON OF THE CODING RESULTS USING DOMAIN GRID OF ONE-AND-THREE LEVEL QUADTREE PARTITIONING (16 × 16, 8 × 8 AND 4 × 4)

Test Images		Algorithms	
		Full Search	Proposed Algorithm (Case 1 & 2)
Lena	Time (s)	5940	3242
	PSNR (dB)	34.91	34.91
	CR	15.50	15.50
Boat	Time(s)	8463	3666
	PSNR (dB)	34.91	34.91
	CR	9.71	9.71
Goldhill	Time(s)	8976	5755
	PSNR (dB)	33.36	33.36
	CR	9.11	9.11

TABLE II
COMPARISON OF COMPUTATIONAL COMPLEXITY FOR THE FULL SEARCH AND OUR PROPOSED ALGORITHM

	Lena	Boat	Goldhill
Full Search	100%	100%	100%
Kick-out	50%	54%	45%
Zero Contrast	3%	3%	2%

When s is set to zero, the corresponding error is given as follows:

$$E(R, D) = A. \tag{12}$$

In this case, the range-domain block matching error can be obtained without performing any calculation, and their error can be represented by the constant value A.

C. Combining Other Approaches

Our proposed algorithm can combine with other fast fractal algorithms to further improve their speed. One example is the DCT Inner Product [2] approach, which allows the computation of two inner products only for the eight isometry operations of a domain block. However, the whole domain pool still has to be considered in order to obtain the best matched domain block. This DCT approach can combine with our algorithm to further improve its speed. In encoding an image, the single kick-out condition (9) will be checked to reject those dissimilar domain blocks. Zero contrast prediction (11) is then used to determine whether the contrast factor is zero or not, and the corresponding error function can be computed without performing the range-domain block matching. Therefore, the required runtime for the algorithm can be further reduced.

IV. EXPERIMENTAL RESULTS

In the experiments, we first compared the performance of our proposed schemes with the full search scheme. A three-level quadtree partition scheme with range block sizes of 4 × 4, 8 × 8 and 16 × 16 pixels, and a search grid of one are used. Three popular 512 × 512 images, Lena, Boat and Goldhill, are used to evaluate the performance of the proposed algorithm and other algorithms. The computer used is a Pentium III 500 MHz. The

TABLE III
COMPARISON OF THE CODING RESULTS USING DOMAIN GRID OF ONE AND 8 × 8 RANGE BLOCK

Test Images		Algorithms			
		Baseline	DCT	DCT + Proposed Algorithm case (1)	DCT + Proposed Algorithm cases (1) & (2)
Lena	Time (s)	5340	2473	1540	1342
	PSNR (dB)	31.14	31.14	31.14	31.14
Boat	Time (s)	5340	2473	1571	1376
	PSNR (dB)	29.21	29.21	29.21	29.21
Goldhill	Time (s)	5340	2473	1625	1501
	PSNR (dB)	29.68	29.68	29.68	29.68

TABLE IV
ADAPTIVE SEARCH ALGORITHM [12]: THE ENCODING RESULTS USING THE TEST IMAGE LENA OF SIZE 256 × 256 WITH RANGE BLOCK SIZE 4 × 4 AND DOMAIN GRID 4 × 4. THE CPU IS PII233 MHz

T_0	T_1	Encoding Time (sec)	PSNR(dB)	Speedup	PSNR Drop(dB)
Full Search		65	31.01	1	0
0	*std* (*R*)/2	26	31.00	2.5	-0.01
0	*std* (*R*)/4	20	30.79	3.25	-0.22
0	*std* (*R*)/8	13	30.28	5	-0.73
3	*std* (*R*)/8	7	30.14	9.28	-0.87
3	*std* (*R*)/16	5	29.36	13	-1.75
3	*std* (*R*)/32	4	28.15	16.25	-2.86

TABLE V
OUR PROPOSED ALGORITHM: THE ENCODING RESULTS USING THE TEST IMAGE LENA OF SIZE 256 × 256 WITH RANGE BLOCK SIZE 4 × 4 AND DOMAIN GRID 4 × 4. THE CPU IS PIII 500 MHz

	Encoding Time (sec)	PSNR(dB)	Speedup	PSNR Drop(dB)
Full Search	38	31.70	1	0
$A\text{-}B<d_{\min}$	20	31.70	1.9	0
$0<A\text{-}B<d_{\min}$	2.5	30.53	15.2	-1.17
$0<A\text{-}B<0.4{*}d_{\min}$	1.2	29.93	31.7	-1.77

runtimes (in seconds) for our proposed algorithm and the full search are listed in Table I. We measured the runtimes of our algorithm based on (i) the single kick-out condition (i.e., case 1), (ii) the zero contrast prediction (i.e., case 2), and (iii) the combination of both conditions (i.e., cases 1 and 2). Experimental results show that, considering both conditions, our proposed algorithm can reduce the required computation by about 50% as compared to the exhaustive search method. In other words, a large number of domain blocks are rejected for performing the range-domain block matching by the kick-out condition, and a number of the error functions are obtained based on the zero contrast prediction.

The required computation of our algorithm is a function of the number of range-domain block matching removed by the kick-out condition, and the number of error functions computed by zero contrast prediction. However, these numbers are image-dependent. Table II tabulates the percentages of the kick-out condition and zero contrast prediction that were applied in encoding the three images. Experimental results show that about 50% of the domain blocks are rejected by the kick-out condition, while 6% of the remaining range-domain block matching can use zero contrast prediction to compute the corresponding error functions. In (6), the kick-out condition has not considered the effect of quantizing the luminance offset, so we cannot guarantee that the optimal domain block will be obtained. Therefore, in order to obtain the best domain block for representing a range block, we set a tolerance of 10% more of the current minimum error $d_{\min}$ when considering whether a domain block to be rejected or not. With this setting, we found that the reconstructed image quality will be equal to that of the exhaustive search.

The performance of our algorithm combined with the DCT Inner Product approach was also investigated. The size of the range blocks is set to 8×8 only. We combined the DCT approach with the single kick-out condition and zero contrast prediction. These combined algorithms were compared with the baseline method and the DCT Inner Product method in terms of the encoding time and PSNR. The experimental results are tabulated in Table III, which shows that the runtime of the combined algorithm is about 25% of the baseline approach and 50% of the DCT approach. Furthermore, the PSNR based on our algorithm is the same as that of the baseline method.

Our algorithm was also compared to the adaptive search algorithm [12] and the tree search algorithm [13]. In [12], 2 bits and 6 bits were used to represent the contrast scaling factor and the range mean. In our algorithm, the corresponding numbers of bits used for these two factors are 5 bits and 7 bits, respectively. Experimental results show that the adaptive search scheme suffers from a significant drop in PSNR at a high speedup, although its compression ratio is slightly higher due to a smaller number of bits being used to represent the two factors. In our proposed algorithm, only those domain blocks satisfying the condition $A - B < d_{\min}$ will be considered as candidate blocks. From Table V, our proposed algorithm can achieve a speedup of 1.9 without any loss of image quality when compared to the conventional full search scheme. In addition, the encoding time required by our algorithm can be further reduced by limiting the search range in the domain pool. If these domain blocks satisfying the condition $0 < A - B < d_{\min}$ are considered as candidate blocks, the speedup of our algorithm will be significantly increased. From Tables IV and V, we can see that our algorithm can achieve a speedup of 15.2 with a PSNR drop of 1.17 dB. At

TABLE VI
TREE SEARCH ALGORITHM [13]: THE ENCODING RESULTS USING THE TEST IMAGE LENA OF SIZE 256 × 256 WITH RANGE BLOCK SIZE 4 × 4 AND DOMAIN GRID 2 × 2. THE MACHINE IS SUN SPARCSTATION 10 MODEL 30

Lena	PSNR(dB)	Time(sec)	Speedup	PSNR Drop(dB)
Full Serach	31.61	8750	1	0
Tree β=20	30.29	340	25	-1.32
Tree β=100	29.34	150	58	-2.27

TABLE VII
OUR PROPOSED ALGORITHM: THE ENCODING USING THE TEST IMAGE LENA OF SIZE 256 × 256 WITH RANGE BLOCK SIZE 4 × 4 AND DOMAIN GRID 2 × 2. THE CPU IS PIII 500 MHz

Lena	PSNR(dB)	Time(sec)	Speedup	PSNR Drop(dB)
Full Serach	33.52	586	1	0
$A-B<d_{min}$	33.52	322	1.8	0
$0< A-B<0.6*d_{min}$	32.17	23	25	-1.35
$0< A-B<0.1*d_{min}$	30.88	10	58	-2.64

a comparable speedup of 16.25, the PSNR drop based on [12] is 1.75 dB.

Tables VI and VII illustrate the performances of the tree search algorithm [13] and our algorithm. When the speedup is 25, the corresponding relative drops in PSNR are 4.2% and 4%. At a higher speedup of 58, the relative drops become 7.18% and 7.88%, respectively. Apparently, the tree search algorithm shows a slightly better performance level at a very high speedup. However, the additional memory requirement of our algorithm is small, and it can achieve the same PSNR and compression ratio (CR) as the full search with a speedup of 1.8. According to [13, Table 6], the tree search algorithm has a large memory requirement. To encode an image of size 256 × 256, 16 MB of memory space will be required to store the leave nodes of the tree for range block size of 4 × 4 with eight isometry operations and 2 × 2 domain grid, and 1.5 MB will be required if pixel average is considered. The experimental results shown in Tables VI and VII were obtained without using the quadtree partitioning scheme, which is normally used in fractal image coding. If quadtree partitioning scheme is adopted, the memory requirement of the tree search algorithm will be increased significantly.

V. CONCLUSIONS

In this paper, we propose a single kick-out condition and the use of zero contrast prediction to speed up the encoding process. The efficiency of the kick-out condition depends on how quickly the global minimum error is detected. Once this global error is found, most of the remaining domain blocks will be rejected and range-domain block matching will not be performed. Experimental results show that the runtime of our algorithm is about 50% of the exhaustive search. Our algorithm can also be combined with other fast fractal coding algorithms, such as the DCT Inner Product, to further improve the speed. The combined algorithm can reduce the required computation by about 75% as compared to the baseline approach. In addition, our algorithm was also compared with an adaptive search algorithm and a tree search algorithm.

REFERENCES

[1] Y. Fisher, Ed., *Fractal Image Compression: Theory and Application*. Berlin, Germany: Springer-Verlag, 1995.

[2] T. K. Truong, J. H. Jeng, I. S. Reed, P. C. Lee, and A. Q. Li, "A fast encoding algorithm for fractal image compression using the DCT inner product," *IEEE Trans. Image Processing*, vol. 9, pp. 529–535, Apr. 2000.

[3] D. Saupe, "Accelerating fractal image compression by multi-dimensional nearest neighbor search," in *Proc. Data Compression Conf.*, Mar. 1995, pp. 222–231.

[4] A. E. Jacquin, "Fractal image coding: A review," *Proc. IEEE*, vol. 81, pp. 1451–1465, Oct. 1993.

[5] D. Saupe, R. Hamzaoui, and H. Hartenstein, "Fractal image compression—An introductory overview," in *Fractal Models for Image Synthesis, Compression and Analysis*. New Orleans, LA: ACM, 1996.

[6] C. K. Lee and W. K. Lee, "Fast fractal image block coding based on local variances," *IEEE Trans. Image Processing*, vol. 7, pp. 888–891, June 1998.

[7] M. Polvere and M. Nappi, "Speed-up in fractal image coding comparison of methods," *IEEE Trans. Image Processing*, vol. 9, pp. 1002–1009, June 2000.

[8] D. Saupe, "Accelerating fractal image compression by multi-dimensional nearest neighbor search," *Proc. IEEE Data Compression Conf.*, pp. 222–231, Mar. 1995.

[9] L. Thomas and F. Deravi, "Region-based fractal image compression using heuristic search," *IEEE Trans. Image Processing*, vol. 4, pp. 832–838, June 1995.

[10] J. Cardinal, "Fast fractal compression of greyscale images," *IEEE Trans. Image Processing*, vol. 10, pp. 159–164, Jan. 2001.

[11] D. Saupe and H. Hartenstein, "Lossless acceleration of fractal image compression by fast convolution," *Proc. IEEE Int. Conf. Image Processing*, vol. 1, pp. 185–188, Sept. 1996.

[12] C. S. Tong and M. H. Pi, "Fast fractal image encoding based on adaptive search," *IEEE Trans. Image Processing*, vol. 10, pp. 1269–1277, Sept. 2001.

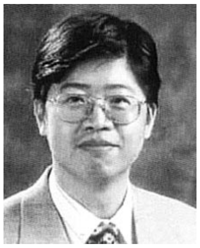
[13] B. Bani-Eqbal, "Enhancing the speed of fractal image compression," *Opt. Eng.*, vol. 34, no. 6, pp. 1705–1710, June 1995.

[14] S. Lee and S. Ra, "An analysis of isometry transforms in frequency domain for the fast fractal coding," *IEEE Signal Processing Lett.*, vol. 6, pp. 100–102, May 1999.

[15] N. Lu, *Fractal Imaging*. New York: Academic, 1997.

Cheung-Ming Lai received the B.Eng. degree in electronic and information engineering from the Hong Kong Polytechnic University in 2000. He is currently pursuing the PhD degree at the Hong Kong Polytechnic University.

His research interests include fractal image compression, subband coding, video coding, and model-based video coding.



Kin-Man Lam (M'96) received his Associateship in electronic engineering with distinction from the Hong Kong Polytechnic University in 1986. He won the S.L. Poa Scholarship for overseas studies and was awarded an M.Sc. degree in communication engineering from the Department of Electrical Engineering, Imperial College of Science, Technology and Medicine, U.K., in 1987. In August 1993, he undertook a Ph.D. degree program in the Department of Electrical Engineering at the University of Sydney, Australia, and won an Australia Postgraduate Award for his studies. He completed his Ph.D. studies in August 1996, and was awarded the IBM Australia Research Student Project Prize.

He joined TechTrend E. & C. Ltd. as an Application Engineer in 1987. From 1990 to 1993, he was a Lecturer at the Department of Electronic Engineering of the Hong Kong Polytechnic University. He joined the Department of Electronic and Information Engineering, The Hong Kong Polytechnic University again as an Assistant Professor in October 1996, and has become an Associate Professor since February 1999. He is a Guest Editor for the Special Issue on Biometric Signal Processing of the *Journal on Applied Signal Processing*. His current research interests include human face recognition, image and video processing, computer vision and architecture, and pattern recognition.

Dr. Lam was a Reviewer, Session Chairman, and member of the Technical Program Committee of the IEEE Symposium on Circuits and Systems (ISCAS'97), the Secretary of the Hong Kong Special Session for the China Fourteenth National Conference on Circuits and Systems, the Secretary of the 2001 International Symposium on Intelligent Multimedia, Video and Speech Processing organized by the Centre for Multimedia Signal Processing, EIE, HKPolyU, a Program Committee Member and Session Chair of the 2002 Conference on Visual Communications and Image Processing, a Principal Member of Technical Programs of the 2002 Seventh International Conference on Control, Automation, Robotics and Vision (ICARCV'02), and the Secretary of the 2003 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'03). Dr Lam was also the Secretary of the IEEE Hong Kong Chapter of Signal Processing between 1992 and 1993 and between 1999 and 2002. He is the Treasurer of the IEEE Hong Kong Chapter of Signal Processing, an International Advisory Committee Member of the 2003 IEEE International Conference on Neural Networks and Signal Processing (ICNNSP 2003), a Program Committee Member of the Workshop on Multimodal User Authentication to be held in Santa Barbara, CA, and the Technical Chair of the 2004 International Symposium on Intelligent Multimedia, Video and Speech Processing (ISIMP 2004) to be held in Hong Kong between Oct. 20–22, 2004.

Wan-Chi Siu (M'77–SM'90) received the Associateship from the Hong Kong Polytechnic University, the M.Phil. degree from the Chinese University of Hong Kong, and the Ph.D. degree from the Imperial College of Science, Technology & Medicine, London, U.K., in 1975, 1977, and 1984, respectively.

He was with the Chinese University of Hong Kong as a Tutor and subsequently as an Engineer between 1975 and 1980. He joined the Hong Kong Polytechnic University as a Lecturer in 1980 and has been Chair Professor since 1992. He was also Associate Dean of the Engineering Faculty (1992–1994), Head of the Department of Electronic and Information Engineering (1994–2000), and Dean of the Engineering Faculty 2000–2002. Since September 1998, he has been Director of Centre for Multimedia Signal Processing. He has published over 220 research papers, over 100 of which appeared in international journals, such as IEEE Transactions and IEE Proceedings. He is editor of the recent book, *Multimedia Information Retrieval and Management* (Berlin, Germany: Springer, 2003). His research interests include digital signal processing, fast computational algorithms, transforms, wavelets, image and video coding, and computational aspects of pattern recognition and neural networks. He is a member of the editorial board of the *Journal of VLSI Signal Processing Systems for Signal, Image, and Video Technology* and the *Journal on Applied Signal Processing* as well as other journals.

Dr. Siu was Guest Editor and Associate Editor of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS II between 1995–1997. He has held the position of general chair or technical program chair of many international conferences. In particular, he was a Technical Program Chair of the IEEE International Symposium on Circuits and Systems (ISCAS'97), and is the General Chair of the 2003 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP'2003). Between 1991 and 1995, he was a member of the Physical Sciences and Engineering Panel of the Research Grants Council (RGC), Hong Kong Government, and, in 1994, he chaired the first Engineering and Information Technology Panel to assess the research quality of 19 Cost Centers (departments) from all universities in Hong Kong.