ISSN 1392 – 124X INFORMATION TECHNOLOGY AND CONTROL, 2006, Vol.35, No.1

A MODIFIED APPROACH TO FRACTAL ENCODING OF BINARY IMAGES

Tomas Žumbakis, Jonas Valantinas

Department of Applied Mathematics, Kaunas University of Technology
Studentų St. 50-325c, LT-51368 Kaunas, Lithuania

Abstract. In the paper, a new modified version of the fractal image encoding technique, adapted to process binary (black and white) images, is presented. A few invariant image parameters (image smoothness estimates, in particular) are introduced and employed to state the necessary image similarity condition. The latter condition plays a key role in the search process for optimal pairings (range block – domain block), i.e., it enables to narrow greatly the search region (domain pool) for each range block.

Experimental analysis results show that implementation of the new modified fractal image encoding technique accelerates image compression times considerably. Exceptionally good results (compression times and quality of restored images) are obtained for binary silhouette images.

Keywords: digital images, image smoothness estimates, fractal image compression.

1. Introduction

Digital image encoding (compression) is usually characterized as a process leading to the reduction of information needed to preserve graphical data (images). There is a great number of image compression techniques (methods), which can be divided into two groups – lossless methods and lossy methods. Data compression techniques such as RLE and LZW fall into the first group. Their specific property – no information is lost at the end of the decompression stage, i.e., restored images coincide completely with the original ones. Unfortunately, the data compression ratios are comparatively low (2-3 times), [1]. Image compression techniques (JPEG, BTC, fractal coding procedures, etc.), falling into the second group, are characterized by much higher compression ratios (15-25 times), achieved, mainly, at the expense of quality of the restored images, [2-6].

The JPEG Standard is one of the most widely used image compression methods. Qualitative characteristics of the method are very good at lower image compression ratios (up to 20 times). For higher ratios, undesirable "block" artifacts appear in the restored images. Worth emphasizing, the JPEG is absolutely inapplicable to the compression of binary (black and white) images, [2]. Fractal image compression techniques distinguish themselves by sufficiently high image compression effect and comparatively small amounts of lost data (at the decompression stage). The main deficiency of the fractal approach – asymmetry ("speed") problem, associated with prolonged image encoding stage and fast image decoding stage. The essence of the fractal approach – detection of similar fragments (blocks) in the image under processing. The best image estimates (restored images) are associated with optimal pairings, i.e., with a thorough analysis of all possible image blocks. The latter circumstance, however, leads to heavy image encoding time expenditures, what makes the real time applications of the approach very problematic.

In the paper, a new version of the fractal image coding technique, oriented to process binary (black and white) images, is presented and analyzed. The approach is based, mainly, on the application of image smoothness estimates and other invariant image parameters in the search process for optimal pairings (range block – domain block).

Theoretical and experimental analysis results showed that the developed version – binary image coding technique – was rather efficient in reducing fractal image compression times, i.e., in solving the earlier mentioned "speed" problem. Moreover, the proposed technique gave exceptionally good results (image compression times and quality of image estimates) for binary silhouette images.

2. Invariant smoothness parameters for binary images – definition, determination

Consider a finite metric space of two-dimensional digital images $(S^2(n), \delta)$, where $S^2(n) = \{[X(m)] \mid m \in I^2\}$, $I = \{0, 1, ..., N-1\}$, $N = 2^n$, $n \in$ N;

$X(m) \in \{0,1,...,2^p - 1\}$, $p \geq 1$; so, parameter p specifies the number of bits attached to encode pixel values in the image (for binary (black and white) images, $p = 1$). The distance (metrics) δ between any two elements (images) of $S^2(n)$ ($[X_1(m)]$ and $[X_2(m)]$) is defined to be

$$\delta = \delta(X_1, X_2) = \left(\frac{1}{N^2} \sum_{m \in I^2} \left(X_2(m) - X_1(m) \right)^2 \right)^{1/2} \quad (1)$$

This formula is applied every time when it is necessary to compare the quality of a restored image against that of the initial one.

In general, the smoothness level of a digital image $[X(m)] \in S^2(n)$, is understood to be the rate of "decay" of spectral coefficients in the discrete spectrum (DCT, Walsh, etc.) of $[X(m)]$, [7].

For binary (black and white) images, smoothness parameters can be defined in a simpler way. Below, we present one criterion for the determination of smoothness level of a binary image. Let $[X(m)] \in S^2(n)$; $X(m) = X(m_1, m_2) \in \{0,1\}$, for all $m_1, m_2 = 0,1,...,N-1$; $n = \log_2 N$. The quantity

$$\alpha = \alpha_x = 1 - \left(\sum_{m_1=0}^{N-1} \sum_{m_2=0}^{N-2} X(m_1, m_2) \oplus X(m_1, m_2+1) + \right.$$
$$\left. + \sum_{m_2=0}^{N-1} \sum_{m_1=0}^{N-2} X(m_1, m_2) \oplus X(m_1+1, m_2) \right) \frac{1}{2N(N-1)}, \quad (2)$$

characterizing the total number of changes (in pixel values) along the image coordinates, is assumed to be the smoothness level (parameter, class) of the binary image $[X(m)]$; here "$\oplus$" signifies the logical operation "EXCLUSIVE-OR".

Image smoothness parameter α_x possesses two highly important (from the standpoint of practical applications) properties:

- Invariance with respect to the isometric transformations (rotation, reflection, inversion), acting upon the image. This property follows directly from the above definition (expression (2));
- Continuity, grasped in the way that small changes in the image call forth small changes in the smoothness parameter value α_x. Indeed, a simple analysis shows that inversion of $k (1 \leq k < N^2)$ bits in the image leads to the increment $\Delta\alpha_x$ (positive or negative) of the smoothness parameter value, which (in absolute value) doesn't exceed $2k / N(N-1)$.

We are to show that the latter property (continuity) can be successfully applied to the detection of similar fragments (blocks) in a binary image, i.e., to finding of optimal pairings (range block-domain block) in block based fractal image coding procedures.

First of all, let us note that, in general, similarity of any two digital images (blocks) is determined using the earlier introduced metrics δ (expression (1)), i.e., two images $[U(m)]$ and $[V(m)]$ are assumed to be similar if and only if $\delta = \delta(U,V) \leq \delta_0$ (δ_0 being a priori prescribed small positive number); otherwise ($\delta > \delta_0$), images $[U(m)]$ and $[V(m)]$ are said to be dissimilar.

Now, on the basis of the continuity property of the parameter α_x, we can state the necessary image similarity condition

$$(\delta(U,V) \leq \delta_0) \Rightarrow (|\alpha_U - \alpha_V| \leq \mu_0); \quad (3)$$

here α_u and α_v are smoothness parameter values for images $[U(m)]$ and $[V(m)]$, respectively. For practical applications, dependence between μ_0 and δ_0 should be established experimentally.

In other words, two images can be characterized as being similar if they fall into the same class of smoothness. The search region (pool) for the best pairings (range block - domain block) can be reduced considerably – it suffices to compare smoothness parameter values (only one operation!) of image blocks (fragments) under analysis.

For the establishment of similarity between two binary images, the use also can be made of quantity, characterizing the relative number of "ones" (or "zeros") in the image:

$$\nu = \nu_x = \left| 1 - \frac{2}{N^2} \sum_{m_2=0}^{N-1} \sum_{m_1=0}^{N-1} X(m_1, m_2) \right|. \quad (4)$$

It is evident that the introduced quantity (monochrome disbalance parameter) ν_x continuously depends on the changes in the image $[X(m)]$. Obviously, ν_x is also invariant with respect to the transformations (rotation, reflection, inversion), acting upon the image.

Despite the fact that ν cannot be used in the role of image smoothness parameter value, it gives one more chance to formulate anew the necessary image similarity condition. Really, for two binary images $[U(m)]$ and $[V(m)]$, we have:

$$(\delta(U,V) \leq \delta_0) \Rightarrow (|\nu_U - \nu_V| \leq \mu_0) \quad (5)$$

So, two images cannot be similar if their monochrome disbalance parameter values differ more than somewhat. The relationship between μ_0 and δ_0 should be established experimentally.

The necessary image similarity condition (expression (3), or (5)) together with the invariance property of earlier introduced image parameters (α_x and ν_x; expressions (2) and (4), respectively) can be successfully employed to accelerate compression times in fractal block based (binary) image coding procedures.

In its simplest form, the block based fractal image encoding idea (Jacquin's approach, [4]) can be described this way – the image to be processed $[X(m)] \in S^2(n)$ is partitioned at two scales (one twice the other) – into the so-called range blocks $[U(m)] \in S_1^2(3) \subset S^2(3)$ and domain blocks $[V(m)] \in S_1^2(4) \subset S^2(4)$. The former (range) blocks are non-overlapping and contain every pixel. The latter ones (domain blocks) may overlap and not necessarily contain every pixel. The essence of the approach is the pairing of each range block $[U(m)]$ to a domain block $[V(m)]$ such that $\delta = \delta(U, V)$ is minimal. Special transformations (rotation, reflection, etc.) are applied to the image (block) $[V(m)]$ to enlarge the detection probability of similar blocks (best pairings) in the image $[X(m)]$. Without any doubt, the computation required is enormous.

To accelerate the pairing process, the search region should be limited, i.e., the domain pools for the range blocks should be decreased considerably. There are many ways to solve this task. A comprehensive review of various approaches (spatial constraints, block classification, etc.) is given in [8].

We here present a new development (approach, technique) in overcoming the "speed" problem, associated with image encoding and decoding steps. The new fractal image coding technique is oriented to process two-dimensional binary (black and white) images.

3. A modified approach to fractal coding of binary images

The fractal image coding procedure, presented in [6], is fully applicable to binary (black and white) images. To say more, for black and white images, additional image compression time gains are achievable. They have links with slight modifications of the fractal image coding strategy, described in [6]. The main change – only non-monochrome range blocks of the binary image are processed at the encoding stage, whereas monochrome range blocks escape analysis, i.e., they are matched by the value 0 (black block) or 1 (white block) at once.

The modified fractal image coding scheme, oriented to process black and white images, is presented in Figure 1.

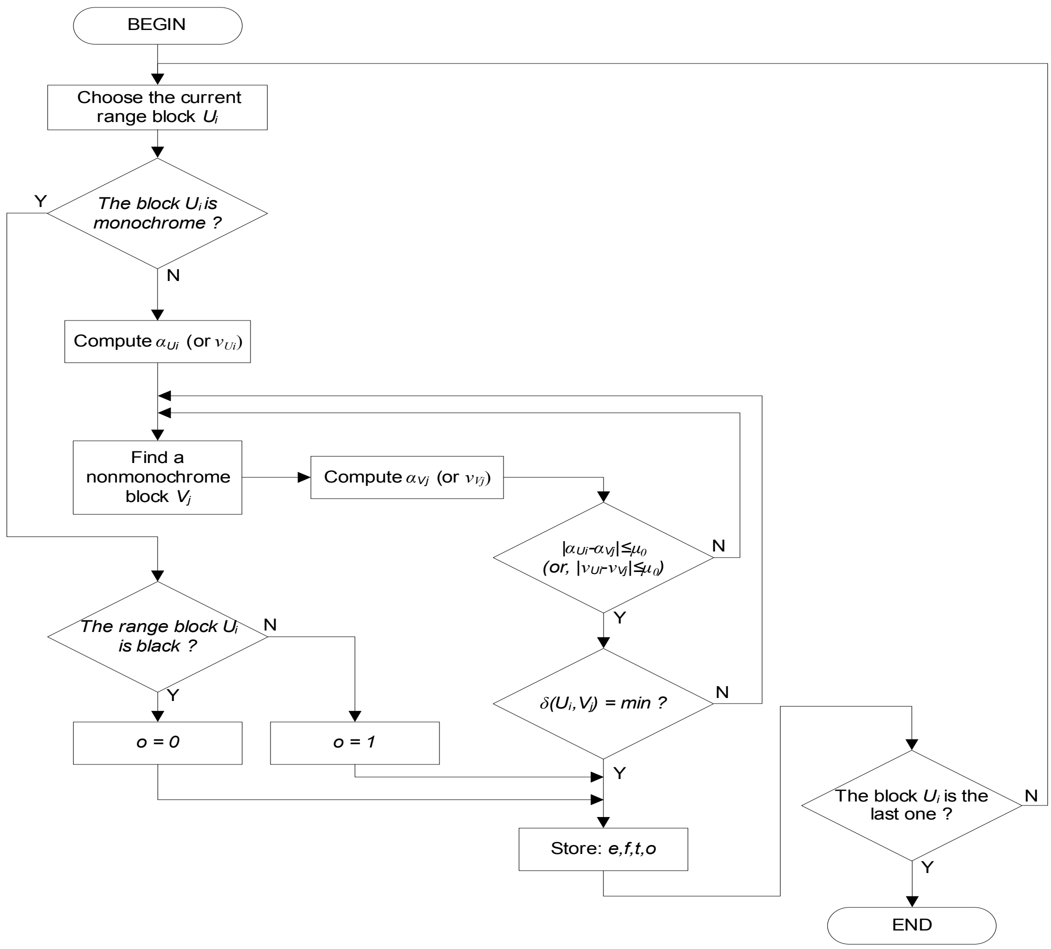

Figure 1. The fractal image coding procedure adapted to process binary images

As it can be seen, range blocks of three distinct types are singled out at the encoding stage, namely: absolutely black, absolutely white and non-monochrome (mixed). Suppose, the size of the image under processing $[X(m)]$ is 256x256, and the size of a range block U is 8x8. If the structure of the stored information (reflecting best pairings "range block U – domain block V " is chosen to be $<e, f, t, o>$, where e and f specify coordinates of the domain block V for a current range block U ($8x2 = 16$ bits needed), t specifies the type of the transformation (3 bits) and parameter o characterizes the type of the range block U (2 bits needed), then the total number of bits needed to encode the given image equals 21×(256/8)× (256/8) = 21504. So, the image compression ratio equals 65536/21504 ≈ 3.05. The result, evidently, doesn't meet any requirements. Is it possible to enhance the image compression effect?

The answer is positive if the binary image under processing is of silhouette type. In this case, the binary image contains quite a number of monochrome (absolutely black or absolutely white) blocks. So, the earlier mentioned structure for data storing becomes surplus and needs to be optimized. No doubt, it suffices to store the parameter values, specifying types of successive range blocks, and 3-tuples $<e, f, t>$ only for non-monochrome (mixed) range blocks. In the latter case, the structure (format) of the stored image data takes the form shown in Figure 2.

o_0	o_1	o_2	o_3	…………….	e_0	f_0	t_0	…	e_i	f_i	t_i	…	e_{N-1}	f_{N-1}	t_{N-1}

The number of 3-tuples is much less than the number of range blocks in the image

Figure 2. The structure of the file for storing of encoded data

For instance, let us take a silhouette (black and white) image 256×256 (Figure 4, b). Since it contains 899 monochrome blocks and 125 mixed blocks, application of the fractal coding procedure (with the range blocks of size 8x8 and the domain blocks of size 16×16) leads to the following result – 4423 bits for stored information (Figure 2), and the image compression ratio 14.82, i.e., nearly five times better than in the previous case.

Amounts of stored information highly depend both on the internal structure of the image under processing and on the size of the range (domain) blocks explored. Thus, it is very difficult to define uniquely "schemes" for storing of encoded data. In a particular case, the number of bits for separate parameters, entering the earlier discussed format (Figure 2), can be determined this way: $e = \log_2(N_X - V_S + 1)$, $f = \log_2(N_Y - V_S + 1)$, where: N_X – the width of the image under processing, N_Y – the height of the image, V_S – the size of the range block; the number of bits, attached to encode parameters o and t, remains to be fixed, i.e., 2 and 3, respectively.

4. Experimental results

To corroborate theoretical analysis results obtained, a few many-sided experiments were carried out. The main goal in these experiments was to find out which one of the proposed three fractal image coding techniques (modifications) gave the best results in the sense of image compression ratio, as well as the quality of restored images.

Quite a number of binary images, falling into different smoothness classes, were processed, namely: *butterfly.bmp*, *girl.bmp*, *car.bmp*, *debate.bmp*, *face.bmp* and *dragon.bmp* (Figures 3, 4).

(a) (b) (c)

Figure 3. Binary images: (a) *butterfly.bmp*, α=0.536; (b) *girl.bmp*, α=0.577; (c) *car.bmp*, α= 0.685

(a) (b) (c)

Figure 4. Binary silhouette images: (a) *debate.bmp*, α= 0.980; (b) *face.bmp*, α=0.990; (c) *dragon.bmp*, α=0.991

Table 1. Results of application of newly developed fractal image encoding techniques

	Method 1, $\lvert\alpha_U - \alpha_V\rvert \le 0.013$		*Method* 2, $\lvert\nu_U - \nu_V\rvert \le 1$		*Method* 3, $\lvert\alpha_U - \alpha_V\rvert \le 0.013$	
Image	Wrong pixels (%)	Time (s)	Wrong pixels (%)	Time (s)	Wrong pixels (%)	Time (s)
butterfly.bmp	55.096	12.313	40.108	7.687	32.373	68.875
girl.bmp	27.136	13.703	27.136	2.797	25.129	233.031
car.bmp	56.109	13.734	25.399	9.422	21.541	26.625
debate.bmp	2.709	9.250	0.839	1.720	0.656	2.766
face.bmp	0.861	3.172	0.385	0.593	0.375	1.703
dragon.bmp	0.853	3.547	0.468	0.610	0.264	1.781

All the selected images (Figures 3, 4) were processed using three approaches: *Method* 1 – detection of best pairings (range block-domain block) in the image is based on the use of image smoothness parameter α (expression (2)); *Method* 2 – detection of best pairings is based on the use of image monochrome disbalance parameter ν (expression (4)); *Method* 3 – detection of best pairings is based on the methodology applied to the encoding of gray-level images, [6].

Results of application of these approaches (with range blocks U and domain blocks V of size 8x8 and 16x16, respectively) are presented in Table 1.

As it can be seen (Table 1), for non-silhouette images (*butterfly.bmp*, *girl.bmp*, *car.bmp*) the results obtained are terribly bad (the number of wrong pixels is very high), whereas, for silhouette images, all three methods lead to quite reasonable results (the quality of restored images is good enough, time expenditures are tolerable). For instance, application of the full search (Table 2) and *Method* 2 gave approximately the same result – the difference in the numbers of wrong pixels doesn't exceed 0.81% (the human eye is not sensitive to changes at such scale!). But, image coding times differ more than 160 times.

Table 2. Full search (no limitation on domain pools)

Image	Wrong pixels (%)	Time (s)
debate.bmp	0.037	285.189
face.bmp	0.006	282.164
dragon.bmp	0.006	287.523

The serious conclusion can be made – the fractal image coding technique, in general, is inapplicable to binary images, but it is efficient enough in the case of binary silhouette images.

Next, two binary images - *butterfly.bmp* (Fig. 3, a; α=0.536) and *dragon.bmp* (Fig. 4, c; α=0.991) were analyzed. Trials were made to find out what is the influence of the threshold value μ_0 on the overall performance (percentage of wrong pixels and compression time expenditures) of the proposed fractal image coding techniques. Range blocks were chosen to be 4x4 and 8x8, respectively. The results obtained (Table 3) once again confirmed that application of the fractal approach to encoding of binary non-silhouette images is deplorable.

For binary silhouette images, the best results were fixed using *Method* 2 (Table 3). Similar analysis of many other binary images showed that the threshold values μ_0, in many cases, ensured detection of optimal pairings and comparatively high overall performance of the proposed approach.

Finally, we did try to lighten the influence of the scale of notation, used at the image decoding stage, on the image processing times. Two cases were analyzed, namely:

- Intermediate image decoding results (obtained after each fractal iteration) were round up to integral values (0 or 1);
- Intermediate image decoding results are presented as real numbers from the interval [0; 1], and only at the end of the decoding stage the round up process is put into action.

Some interesting results are presented in Table 4 (Method 2; $\nu \le 3$; blocks U and V of size 4×4 and 8×8, respectively).

Table 3. The influence of the threshold size on the binary image coding results

Image	*Method* 1			*Method* 2			*Method* 3		
	Wrong pixels (%)	Time (s)	μ_0	Wrong pixels (%)	Time (s)	μ_0	Wrong pixels (%)	Time (s)	μ_0
butterfly.bmp	50.577	22.625	0.001	26.230	13.688	=0	20.181	140.515	0.001
	47.849	22.625	0.01	22.633	36.062	≤1	17.488	155.875	0.01
	44.879	57.203	0.05	19.786	58.375	≤2	15.704	182.125	0.05
	44.206	91.641	0.1	16.147	105.109	≤4	14.661	211.016	0.1
	42.418	307.235	0.4	14.465	244.735	≤16	12.280	362.297	0.4
dragon.bmp	0.664	1.265	0.001	0.018	0.438	=0	0.240	0.609	0.001
	0.664	1.250	0.01	0.009	0.703	≤1	0.075	0.750	0.01
	0.664	2.172	0.05	0.006	0.984	≤2	0.017	0.922	0.05
	0.664	7.766	0.1	0.006	1.453	≤4	0.014	1.109	0.1
	0.664	25.265	0.4	0.006	2.781	≤16	0.006	2.188	0.4

Table 4. Representation of the intermediate results at the image decoding stage

Image	Integers		Real numbers	
	Wrong pixels (%)	Time (s)	Wrong pixels (%)	Time (s)
debate.bmp	2.254	5.562	0.037	5.203
face.bmp	0.084	1.375	0.006	1.296
dragon.bmp	1.813	1.312	0.008	1.266

One can easily ascertain that the quality of restored images is much better when all the intermediate data (at the iterative image decompression stage) are represented as real numbers.

5. Conclusion

In the paper, a new version of the fractal image coding procedure, adapted to process binary images, is presented. At the image encoding stage, the limited search regions (domain pools) for particular range blocks are formed using some newly developed criteria, namely: (1) the total number of changes (in pixel values) along the image coordinate axes – binary image smoothness parameter (expression (2); Section 2); (2) the relative number of “ones” and “zeros” in the image - image monochrome disbalance parameter (expression (4); Section 2); (3) the rate of “decay” of spectral coefficients in the discrete spectrum of the image under processing. All these criteria have been successfully employed to formulate the necessary image (block) similarity condition, which played a key role in narrowing (limiting) domain pools for the image range blocks.

Experimental analysis results showed that all three criteria “worked” sufficiently well when applied to binary silhouette images. Exceptionally good performance was fixed in the case of the second criterion - best pairings (range block – domain block) were searched among the image blocks containing comparatively the same number of “ones” (“zeros”). In parallels, a new format (structure) for storing of encoded binary image data was presented.

Finally, an interesting detail has been found out – higher quality of restored binary images is obtained when (at the iterative image decompression stage) intermediate computational results are presented as real numbers (not integers), with the round up procedure at the very end.

References

[1] **D. Hankerson, G.A. Harris, P.D. Johnson, Jr.** Introduction to Information Theory and Data Compression (Second Edition). *CHAPMAN&Hall/ CRC Press Company*, 2003.

[2] **G.K. Wallace.** The JPEG Still Picture Compression Standard. *Communications of the ACM*, *Vol.*34, *No.*4, 1991, 30-44.

[3] **P. Fränti, O. Nevalainen, T. Kaukoranta**. Compression of Digital Images by Block Truncation Coding: A survey. *The Computer Journal, Vol.*37, *No.*4, 1994, 308-332.

[4] **A. Jacquin**. Fractal Image Coding: A review. *Proceedings of the IEEE*, *Vol.*81, *No.*10, 1993, 1451-1465.

[5] **M.F. Barnsley, L.P. Hurd.** Fractal Image Compression. *AK Peters, Wellesley, Mass*, 1993.

[6] **T. Žumbakis, J. Valantinas.** The Use of Image Smoothness Estimates in Speeding Up Fractal Image Compression. *Lecture Notes in Computer Science* 3540 (*Proceedings of the* 14th *Scandinavian Conference SCIA* 2005, *Joensuu, Finland*). ISSN 0302-9743. *Springer-Verlag Berlin Heidelberg*, 2005, 1167-1176.

[7] **J. Valantinas, T. Žumbakis.** Definition, evaluation and task-oriented application of image smoothness estimates, *Information Technology and Control*, *No.*1 (14), *Technologija, Kaunas*, 2004, 15 – 24.

[8] **W. Brendt, G. de Jager**. A Review of the Fractal Image Coding Literature. *IEEE Transactions on Image Processing*, *Vol.*8, *No.*12, 1999, 1716-1729.

Received December 2005.

DOI: 10.5755/j01.itc.35.1.12040