A Modified Fractal Transformation to Improve the Quality of Fractal Coded Images

José F. L. de Oliveira
E-mail: jleite@lpc.ufrj.br

Gelson V. Mendonça
Universidade Federal do Rio de Janeiro
Caixa Postal 68504, CEP: 21945-970
Rio de Janeiro, RJ, Brazil
E-mail: gelson@lps.ufrj.br

Roosevelt J. Dias
E-mail: roose@coe.ufrj.br

Abstract

The Discrete Fractal Transformation has recently emerged as a powerful technique for coding images. The scheme works by dividing an image into blocks and making use of a contraction mapping by which the domain blocks are mapped into range blocks. During this process it is usual to filter the domain blocks. In this work, we show how a change in the filtering procedure can improve the image quality of fractal coding methods based on the mentioned transformation. After a presentation of the discrete fractal transformation, we explain the change and we apply it to the quadtree-based fractal scheme obtaining improvements in PSNR that can reach 1 dB when compared with the results obtained by traditional filtering.

1. Introduction

Fractal coding consists of mapping the domain blocks into range blocks in which an image is previously divided as described in [1], [2] and [3]. Usually these blocks have the same shape but their sizes are different and to compensate this difference a kind of lowpass filtering, followed by decimation, is performed. The mapping is made by means of a contractive Discrete Fractal Transformation and the coding scheme makes the assumption that images are piecewise self-similar [1]. The fractal coding schemes based on this transformation exhibit, in general, long compression time to obtain good quality images when compared with well known algorithms as those based on the Discrete Cosine Transform. Several works [1] were developed to reduce the compression time without decreasing significantly the image quality. In this paper, we show how a modification in the filtering procedure of the domain blocks can enhance the image quality, which is measured by the PSNR (*Peak Signal to Noise Ratio*).

2. The discrete fractal transformation

Fractal compression algorithms are derived from the Iterated Function System (IFS) theory developed by Barnsley [4] and Hutchinson [5]. However, images produced by these systems are self-similar and, in general, real world images do not have this characteristic. Thus, the notation used in the original IFS theory is not adequate for application in digital image compression. A more efficient notation is provided by the Discrete Fractal Transformation. We begin giving definitions of an image and an associated metric space, which are used to define a Discrete Fractal Transformation as proposed by Fisher in [1]. Then its application to fractal image compression is briefly explained.

Definition 1 Let X and Y be positive integers. Define the set $\mathbf{S} = \mathbf{X} \times \mathbf{Y}$ where $\mathbf{X} = \{0,1,2,\ldots, X-1\}$ and $\mathbf{Y} = \{0,1,2,\ldots,Y-1\}$. Let f be a function from $\mathbf{S}$ to $\boldsymbol{R}$, where $\boldsymbol{R}$ is the set of real numbers. An *image* $\mathbf{L} \subset (\mathbf{S} \times \boldsymbol{R})$ is the graph of the function $f{:}\mathbf{S} \to \boldsymbol{R}$, that is, $\mathbf{L} = \{((x,y), f[x,y]) \in \mathbf{S} \times \boldsymbol{R}$. $\mathbf{S}$ is called the *support* of the image $\mathbf{L}$. The notation $f[\cdot]$ is used to indicate that f is a discrete two-dimensional function.

Definition 2 Let $\mathbf{P}$ be the set whose points are images with the same support $\mathbf{S}$. Define a metric for the space $\mathbf{P}$ by $d_{\mathbf{P}}(f,g) = \max\{|f[x,y] - g[x,y]| : (x,y) \in \mathbf{S}\}$, for all $f, g \in \mathbf{P}$. Then $(\mathbf{P}, d_{\mathbf{P}})$ is the *metric space of images*.

A metric space is a set (space) together with a real-valued function, which measures the distance between pairs of points on this space.

0-8186-8821-1/98 $10.00

Definition 3 Let $\mathbf{D}_1, \mathbf{D}_2, \ldots, \mathbf{D}_N$ and $\mathbf{M}_1, \mathbf{M}_2, \ldots, \mathbf{M}_N$ be non-empty subsets of $\mathbf{S}$ such that $\mathbf{S} = \bigcup_{i=1}^{N} \mathbf{M}_i$ and $\mathbf{M}_i \cap \mathbf{M}_j = \emptyset$ for $i \neq j$. Let $(\mathbf{P}, d_\mathbf{P})$ be the metric space of images. Let $f \in \mathbf{P}$ and define the W operator by

$$(Wf)[x,y] = a_i[x,y]\, f[u_i[x,y]] + b_i[x,y] ,$$

where a_i is a function from $\mathbf{M}_i$ to the real interval $[-\alpha_{\max}, \alpha_{\max}]$, u_i is a function from $\mathbf{M}_i$ to $\mathbf{D}_i$, b_i is a function from $\mathbf{M}_i$ to $\boldsymbol{R}$ and, $(x,y) \in \mathbf{M}_i$ for $i = 1, 2, \ldots, N$. $\alpha_{\max}$ is a non-negative real constant. The function $W{:}\mathbf{P} \to \mathbf{P}$ is called a *discrete fractal transformation.*

It can be shown that W has a unique fixed point $f_W \in \mathbf{P}$ such that $Wf_W = f_W$ and $\lim_{m\to\infty} W^{\circ m} f = f_W$ for all $f \in \mathbf{P}$. This comes from the fact that $(\mathbf{P}, d_\mathbf{P})$ is a complete metric space and W is a contraction map in $(\mathbf{P}, d_\mathbf{P})$ [4].

Definition 4 We say that a map $\theta{:}\mathbf{E} \to \mathbf{E}$ in the metric space $(\mathbf{E}, d)$ is *contractive* if, for $m = 1$, there is $s \in [0,1)$ such that

$$d(\theta^{\circ m}(x), \theta^{\circ m}(y)) \leq s d(x,y)$$

for all $x, y \in \mathbf{E}$. When this occurs for $m > 1$ we say that the map is *eventually contractive*.

This is the key of fractal image compression. If we can find a W such that f_W is close to a given image f, where closeness is measured by the $d_\mathbf{P}$ metric (in practice, it is used the Euclidean metric or the rms metric [1]), then we can take W as a fractal representation of f. Compression is achieved if W can be stored compactly.

Practical implementations of fractal compression algorithms adopt for the scale function $a_i{:}\mathbf{M}_i \to [-\alpha_{\max}, \alpha_{\max}]$ and for the offset function $b_i{:}\mathbf{M}_i \to \boldsymbol{R}$ constant values for $i = 1, 2, \ldots, N$. That is, $a_i[x,y] = \alpha_i$ and $b_i[x,y] = \beta_i$ for each $(x,y) \in \mathbf{M}_i$, where α_i and β_i are constants. The function $u_i{:}\mathbf{M}_i \to \mathbf{D}_i$ allows that rotated versions of the domain blocks $f[\mathbf{D}_i]$ be compared with the range blocks $f[\mathbf{M}_i]$ during the coding process. In the quadtree partition scheme [1] the support of the sub-images $f[\mathbf{M}_i]$ and $f[\mathbf{D}_i]$ have square shape. Using matrix notation we can write $f[\mathbf{M}_i]$ and $f[\mathbf{D}_i]$ as

$$\begin{bmatrix} \mu_{00}^{(i)} & \cdots & \mu_{0(M-1)}^{(i)} \\ \vdots & \ddots & \vdots \\ \mu_{(M-1)0}^{(i)} & \cdots & \mu_{(M-1)(M-1)}^{(i)} \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} \delta_{00}^{(i)} & \cdots & \delta_{0(D-1)}^{(i)} \\ \vdots & \ddots & \vdots \\ \delta_{(D-1)0}^{(i)} & \cdots & \delta_{(D-1)(D-1)}^{(i)} \end{bmatrix}$$

respectively, where $D = 2M$ is a common choice. Thus, we need to downsample the domain blocks before compare them with the range blocks. However, if we only downsample the domain blocks using $\overline{\delta}_{kl}^{(i)} = \delta_{(2k)(2l)}^{(i)}$, where $k, l = 0, 1, \ldots, M-1$, to approximate the $\mu^{(i)}$ values, we may obtain aliasing. Filtering the domain blocks before downsampling can reduce the aliasing. Making $\overline{\delta}_{kl}^{(i)} = \frac{1}{4}\sum_{p=0}^{1}\sum_{q=0}^{1}\delta_{(2k+p)(2l+q)}^{(i)}$ is equivalent to a lowpass filtering followed by downsampling by four and this reduces the mixing of low and high frequencies. In fact, the use of the averaging filter results in better encoded images, that is, we have an improvement in the PSNR.

3. Modified discrete fractal transformation

Since the arithmetic average corresponds to a type of lowpass filtering, high frequency attenuated versions of the regions $f[\mathbf{D}_i]$ are being mapped into regions $f[\mathbf{M}_i]$ that are not filtered. Then reducing the lowpass effect of the average in the domain blocks we expect to have a better reproduction of the range blocks. However, this effect can not be totally eliminated because we need to map blocks with $2M \times 2M$ points into blocks of $M \times M$ points. A way to address this problem is to multiply the average value above mentioned by factors greater than one. Hence, we can adopt

$$\overline{\delta}_{kl}^{(i)} = \lambda_i \frac{1}{4}\sum_{p=0}^{1}\sum_{q=0}^{1}\delta_{(2k+p)(2l+q)}^{(i)} ,$$

where $\lambda_i > 1$. This modification in the filtering process is equivalent to have $\alpha_{\max} > 1$. This gives us some explanation for the fact observed by Fisher [1] that allowing $\alpha_{\max} > 1$ produces better encoded images.

When the domain blocks are sufficiently band limited there is no aliasing and they have a greater potential to approximate a given range block. In this case, only a stretch by two in frequency, caused by the downsampling process, and a scaling in amplitude, caused by the filtering process, affect their original spectrums. Scaling these filtered and downsampled domain blocks by factors less than one, we reduce the amplitude of their spectrums. This also reduces the diversity of domain blocks that may cover a given range block. Allowing scale factors greater than one, we expand the universe of domain blocks with spectrum close to those of the range blocks. If the resulting Discrete Fractal Transformation is eventually contractive the sequence $\{W^{\circ m} f : m = 0, 1, 2, \ldots\}$ converges and we obtain better encoded images. Then we can formulate the following question: Is there any special value for $\alpha_{\max}$, greater than one, that can be used to code a wide set of images such that the fractal transformation be eventually contractive? The answer may be yes as we shall see in the results section.

4. Results

We present here some coding results to verify the observations made in the last section. We want to determine if we can find an "optimal" α_{max} that can be used for a wide set of images. We use a quadtree-based fractal scheme with two partition levels to code all images. The range blocks can have sizes of 4x4 and 8x8 pixels. The domain pool is obtained from a regular grid with a step equals to the size of the domain blocks. The α_i values are quantised with 5 bits and the β_i values are quantised with 7 bits. We are using random images for starting the decoding process to observe if we have eventually contractive fractal transformations. We divide a range block of size 8x8 into four of size 4x4 if we have an rms error greater than 8.

We code a set of seven distinct images of diverse sizes (see Table 1) using a range of α_{max} values that vary from 0.6 to 2.4. Averaging the results of all these coded images we construct the graph in figure 1. From this figure it is easy verified that a possible optimum value for α_{max} is located between 1.6 and 1.8. For $\alpha_{max} > 1.8$, we begin to have regions in the image that do not converge and the PSNR reduces drastically. In figure 2, we plot the average bit rate of all test images versus α_{max} to illustrate the compression rate achieved.

Table 1. Images used to obtain the graphs in figures 1 and 2.

Image	Baboon, Lena	Eltoro, Lena	Barbara, Boats, Zelda
Size	256x256	512x512	720x576

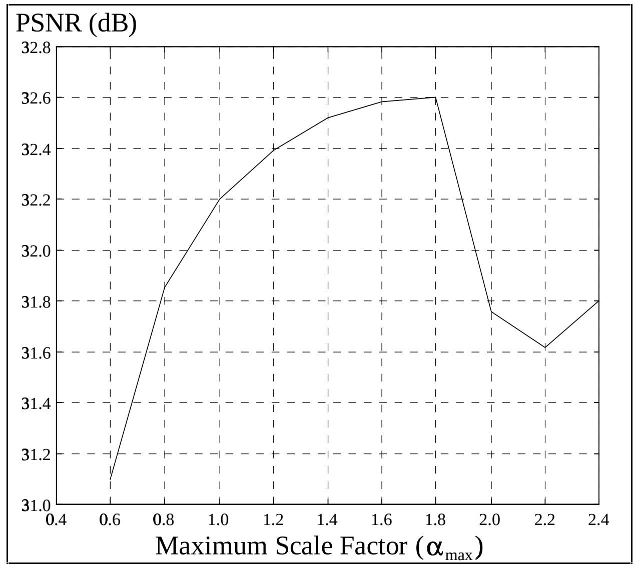

Figure 1: Average PSNR as a function of α_{max}.

In figures 3 to 4, we plot PSNR versus bit rate using $\alpha_{max} = 1$ and $\alpha_{max} = 1.7$ for Barbara and Boats images. In figures 5 to 7, we present three other graphs for images that are not used to obtain the graphs in figures 1 and 2. In all images, we can observe an improvement in PSNR when we use $\alpha_{max} = 1.7$. Moreover, none of these images presented instability in convergence for this value of α_{max}. That is, the fractal transformation remained eventually contractive. It seems also that the edges are better reproduced. We made experiments with several images obtaining similar behaviour. This reinforces our claim that α_{max} in the range between 1.6 and 1.8 produces better results while maintaining convergence of the fractal transform.

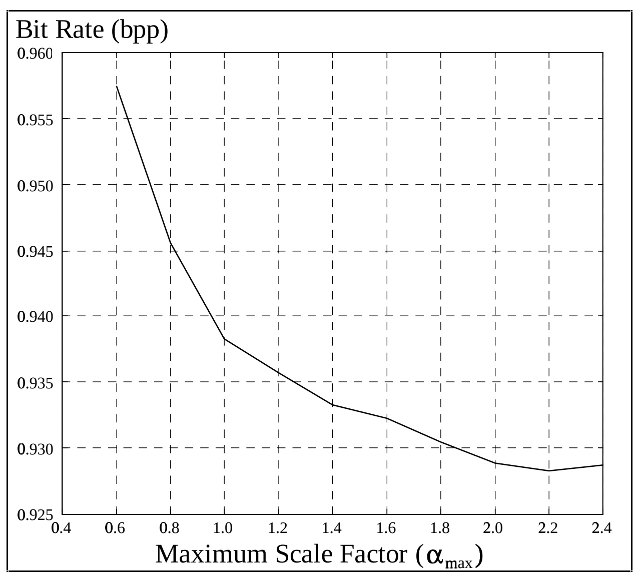

Figure 2: Average bit rate for the coded images as a function of α_{max}.

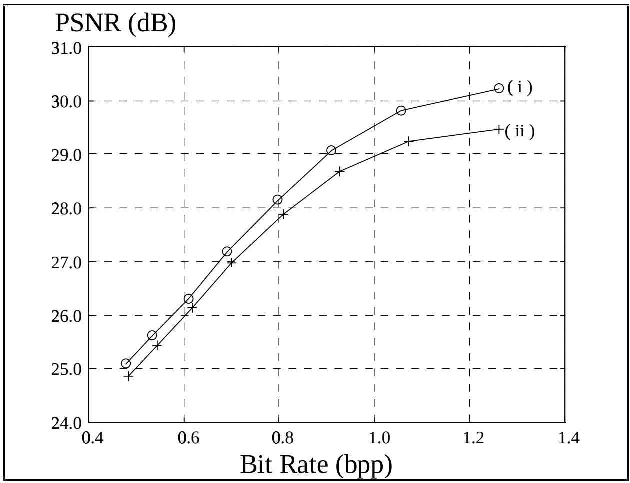

Figure 3. Results for Barbara 720x576. In (i) $\alpha_{max} = 1.7$ and in (ii) $\alpha_{max} = 1.0$.

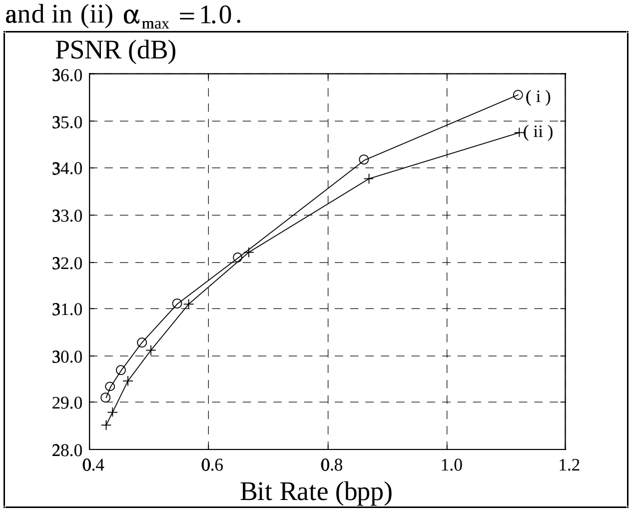

Figure 4. Results for Boats 720x576. In (i) $\alpha_{max} = 1.7$ and in (ii) $\alpha_{max} = 1.0$.

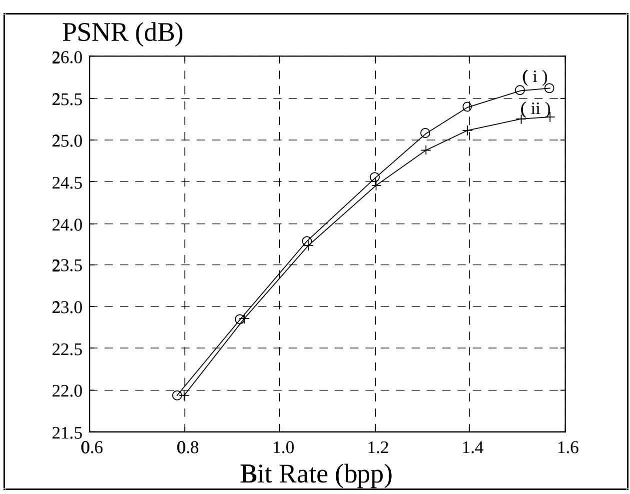

Figure 5. Results for Frog 256x256. In (i) $\alpha_{max} = 1.7$ and in (ii) $\alpha_{max} = 1.0$.

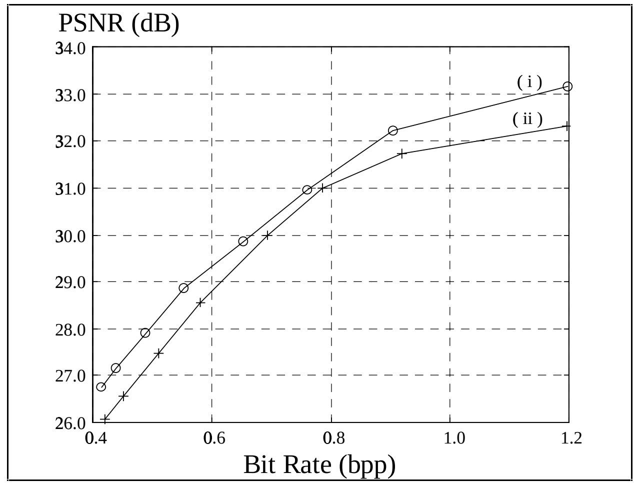

Figure 6. Results for Peppers 256x256. In (i) $\alpha_{max} = 1.7$ and in (ii) $\alpha_{max} = 1.0$.

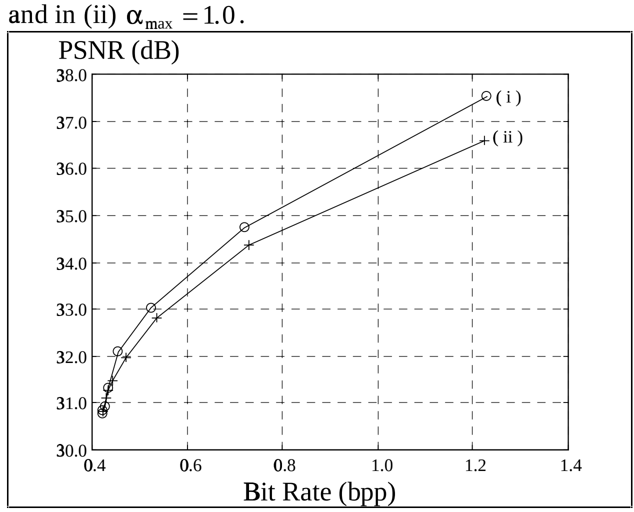

Figure 7. Results for Girl 720x576. In (i) $\alpha_{max} = 1.7$ and in (ii) $\alpha_{max} = 1.0$.

5. Conclusions

In this paper, we explain how a change the filtering procedure adopted in the decimation of the domain blocks can improve the image quality of fractal coding methods based on the Discrete Fractal Transformation. The presented results show that the proposed modification obtains improvements in PSNR that can reach 1 dB. We also observe that maximum scale factors less than 1. 6 reduce the coding performance and maximum scale factors greater than 1 .8 may cause instability in the decoding process.

6. References

[1] Y. Fisher (editor), *Fractal Image Compression*, New York, Springer-Verlag, 1994.

[2] E. W. Jacobs, Y. Fisher, R. D. Boss, "Image Compression: Study of the Iterated Transform Method," *Signal Processing*, n. 29, pp. 251-263, 1992.

[3] Y. Fisher, E. W. Jacobs, R. D. Boss, *Fractal Image Compression Using Iterated Transforms*, In: Technical Report 1408, Naval Ocean Systems Center, San Diego, CA, 1991.

[4] M. F. Barnsley, *Fractals Everywhere*, 2 ed. Cambridge, MA, Academic Press, 1993.

[5] J. Hutchinson, "Fractals and Self-Similarity," *Indiana University Mathematics Journal*, v. 30, n. 5, pp. 713-747, 1981.