in *Proc. of Visual Communications and Image Processing*, 1:132-140, Boston, November 93

Sequence Coding based on the Fractal Theory of Iterated Tranformations Systems

Emmanuel Reusens
Signal Processing Laboratory
Swiss Federal Institute of Technology
CH-1015 Lausanne, Switzerland

Abstract

A new scheme for sequence coding based on the iterated functions systems theory is presented. The method relies on a 3D approach in which the sequence is adaptively partitioned. Each partition block can be coded either by using the spatial self similarities or by exploiting temporal redundancies. The proposed system shows very good performances when compared to other existing methods.

1 Introduction

The concept of *fractal* has been introduced by Mandelbrot [1] in the 1960's as an alternative to the classical Euclidean geometry mainly for describing shapes generated by Nature. Since then, this theory attracts the interest of many researchers in fields ranging from biology to physics through computer imaging and image processing. Iterated contractive functions systems (IFS) have shown their capability to generate fractal structures. The efficiency of such systems for representing fractals is explained by the self-similar appearance of these objects. Fractals can display very intricate structures but they are characterized by statistical or deterministic similarities across scales.

From an image processing point of view, it turns out that some form of self-similarity does exist in many natural images. Such an observation induces naturally to use the theory of fractal for image coding purposes. Barnsley in [2] gives general principles on the use of iterated transformations in the context of image coding. Impressive compression ratios announced by Barnsley however stand only for some particular images (as trees or ferns) possessing a high degree of self similarity, which is obviously not the case for natural images. A lack of details about this scheme and its results induced a feeling of scepticism until Jacquin gave a more detailed version of fractal image coding based on *partitioned* iterated contractive transformations systems [3].

Several schemes for still image compression have been developed over the past few years [4] [5] [6] while no techniques for sequence coding have been presented. This paper proposes to fill up this gap. The scheme described in this paper relies on a tridimensional approach based on an adaptive partitioning of the sequence support. This partitioning enables one to vary in an optimal way the size of the blocks according to their statistical properties both temporal and spatial. The method allows to exploit both the spatial self-similarities and the temporal redundancies. The system shows very good performances when compared with other traditional methods.

This paper is organized as follows. Section 2 briefly reviews the theoretical background of iterated contractive functions systems. A complete description of the sequence encoding procedure is given in section 3. In section 4, the adaptive partitioning of the sequence support is presented. Section 5 presents the simulation results while section 6 gives some interpretations explaining the quality of the system performances. Section 6 concludes this paper by citing some future improvements currently under investigation.

2 Theoretical Foundations

A transformation τ on a metric space $(\mathcal{M},d)$ is called contractive if there is a constant $0 \leq s < 1$ such that

$$d(\tau(\mu),\tau(\nu)) \leq s d(\mu,\nu) \quad \forall \mu,\nu \in \mathcal{M}. \tag{1}$$

The factor s is called the *contractivity factor* of the transformation τ. A fundamental property of contractive transformations in a complete metric space is that iterative application to *any* initial subset of $\mathcal{M}$ assures convergence to a unique fractal attractor.

In the context of image coding, we are facing the *inverse* problem i.e. finding the contractive transformation whose fixed point is close to a given set. A clue to solving this problem is given by the *Collage* theorem [2]:

Let $(\mathcal{M},\ d)$ denotes a complete metric space where d represents a given distance and let μ_{orig} be the set to approximate. If the transformation τ from $\mathcal{M}$ to $\mathcal{M}$ satisfies the following requirements:

- $\exists s < 1$ such that $\forall \mu, \nu \in \mathcal{M}, d(\tau(\mu), \tau(\nu)) \leq s d(\mu, \nu)$ (contractivity requirement) (2)
- $d(\mu_{orig}, \tau(\mu_{orig})) \leq \varepsilon$ (3)

then for any set $\mu_0 \in \mathcal{M}$ and for any positive integer n :

$$d(\mu_{orig}, \tau^n(\mu_0)) \leq \frac{\varepsilon}{1-s} + s^n d(\mu_{orig}, \mu_0). \tag{4}$$

This theorem tells us that we need to find a contractive transformation under which the given set is an approximate fixed point. Then, according to inequality (4), it can be seen that applying the transformation iteratively to any initial set μ_0 causes convergence to an attractor close to the target set. Therefore the transformation τ fully describes the fractal approximation of the original set. For further details, the reader is referred to [2].

3 Fractal based Sequence Coding

The capability of iterated functions systems to generate very intricate structures suggests their use for image coding purposes. In the context of sequence coding, $(\mathcal{M}, d)$ denotes the complete metric space of digital sequences where d represents the traditional mean-square-error:

$$\forall \mu, \nu \in \mathcal{M},\ d(\mu, \nu) = \frac{1}{KLM} \sum_{i=1}^{K} \sum_{j=1}^{L} \sum_{k=1}^{M} (\mu_{i,j,k} - \nu_{i,j,k})^2 \tag{5}$$

The self-similarity encountered in natural images differs from the one present in fractal objects. Instead of having an image formed by copies of its whole self, one has an image formed by copies of properly transformed *parts* of itself. Therefore, as described by Jacquin [3] for the two-dimensional case, the class of contractive transformations considered for sequence coding purposes is defined blockwise. If $\{R_i, 0 \leq i < N\}$ denotes a nonoverlapping partition of the sequence support into N tridimensional *range cells*, the transformation τ is defined by:

$$\forall \mu \in \mathcal{M}, \tau(\mu) = \sum_{0 \leq i < N} \tau(\mu)_{|R_i} = \sum_{0 \leq i < N} \tau_i(\mu_{|D_i}) \qquad \text{with } \tau_i : D_i \rightarrow R_i \tag{6}$$

where $\mu_{|R_i}$ denotes the restriction of the sequence μ to the cell R_i. D_i is called a *domain cell* and denotes a block belonging to the sequence but not necessarily to the partition.

In that context, the *Collage* theorem states that the set of contractive transformations $\{\tau_i, 0 \leq i < N\}$ under which the original sequence is an approximate fixed point possesses an attractor close to this sequence. Therefore the encoding process implies to find, for each of the N *range cells* R_i of the partition, the transformation τ_i leading to the best approximation of R_i with respect to the distortion measure adopted[1]. This operation is performed by scanning all possible blocks D_j and by evaluating the remaining parameters of the transformation τ_j in order to minimize $d(\tau_j(D_j), R_i)$. The transformation τ_i leading to the best approximation of R_i is then selected and encoded. The transformation parameters are the only data encoded and no grey-level information is needed to recover the attractor.

Scanning of the domain blocks is performed within a search area surrounding the range block. This limitation is imposed by computation time requirements and induces a compact representation of the address of D_i within the transformation τ_i. This search can be led either among blocks belonging to the same set of frames as the block R_i or

[1] It must be noted that the specification of the position of the domain cell D_i is included in the description of τ_i.

among blocks belonging to a different set. In the first case, we intend to exploit the spatial self-similarities while in the second case we expect to reduce temporal redundancies. Those alternatives must be sharply distinguished since they lead to two very different processings. Subsection 3.1 will treat the first case while the second alternative will be presented in subsection 3.2.

3.1 Spatial self-similarities

Using spatial self-similarities means exploiting information from one scale to represent information at a lower scale. For this reason, the *spatial* size of the domain blocks is chosen to be twice the size of the range blocks. This factor two proceeds from a trade-off between the quality of the approximation $\tau(\mu_{orig})$ (which experimentally decreases with an increase of the shrinking factor) and the contractivity factor s (which decreases with an increase of the shrinking factor) which will condition the distance between the original sequence μ_{orig} and the final attractor like:

$$\lim_{n\to\infty} d(\mu_{orig}, \tau^n(\mu_0)) \leq \frac{1}{1-s} d(\mu_{orig}, \tau(\mu_{orig})) \tag{7}$$

For sake of clarity, τ_i can be written as the composition of two transformations $\mathcal{S}_i$ and $\mathcal{T}_i$:

$$\tau_i = \mathcal{T}_i \circ \mathcal{S}_i, \tag{8}$$

where $\mathcal{S}_i$ represents the *geometric part* of the transformation, while $\mathcal{T}_i$ denotes a *massic transformation.*

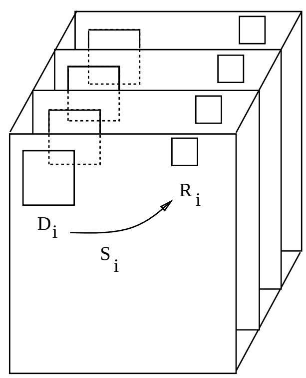

Figure 1: Geometric transformation $\mathcal{S}_i$

The geometric transformation corresponds to a mapping in position and in size from the domain cell D_i to the range cell R_i as illustrated in figure 1. Shrinking is performed by a four by four pixels averaging process. The massic part of the transformation consists in a modification of the pixels values inside the block. This last transformation allows to change the grey-level information in order to find a good approximation of the block R_i. Once again a compromise needs to be reached between the complexity of the transformation and the compactness of its representation. A study of that trade-off for the two-dimensional case can be found in [6]. In this paper, we will use one of the simplest transformations. We will consider only eight possible shuffles of pixels (4 rotations and 4 reflections), a contrast scaling α and a brightness shift Δg. The massic part of the transformation τ_i can therefore be expressed by:

$$\mathcal{T}_i \begin{pmatrix} x \\ y \\ t \\ z \end{pmatrix} = \begin{pmatrix} a_i & b_i & 0 & 0 \\ c_i & d_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & \alpha \end{pmatrix} \begin{pmatrix} x \\ y \\ t \\ z \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 0 \\ \Delta g \end{pmatrix} \tag{9}$$

where the parameters $a_i, b_i, c_i,$ and d_i are constrained to correspond to one of the eight isometries and $|\alpha|$ to be strictly lower than 1 due to the contractivity requirement. Indeed, a simple calculation shows that the contractivity factor s of the massic transformation is α^2.

The search for the best transformation is performed by scanning all the possible domain blocks D_j in the same set of frames as R_i. The geometric transformation $\mathcal{S}_j$ is then applied in order to map D_j onto R_i. Once $\mathcal{S}_j(D_j)$ is computed, the parameters of the massic transformation $\mathcal{T}_j$ are evaluated in order to minimize:

$$d(\mathcal{T}_j \circ \mathcal{S}_j(D_j), R_i) \tag{10}$$

The global transformation $\mathcal{T}_i \circ \mathcal{S}_i$ leading to the best approximation of R_i is then selected and encoded.

It must be noted that using self-similarities for approximating *3D* blocks exploits also, in a certain sense, temporal redundancies.

3.2 Temporal redundancies

The difficulty to use the contractive functions systems theory for reducing temporal redundancies proceeds from the contractivity requirement. The temporal redundancies consist in frame to frame similarities but on the *same* scale. It does mean that the transformations τ_i possess a contractivity factor s close to 1. According to equation (7), it results in a dramatic increase of the distance bound between the attractor and the original sequence since 1/(1-s) $\rightarrow \infty$.

In order to overcome this problem, we are forced to impose a temporal direction in which the search will be performed. The transformations τ_i will be reduced to a geometric translation. The exploitation of temporal redundancies will therefore be performed by using:

$$\tau_i \begin{pmatrix} x \\ y \\ t \\ z \end{pmatrix} = \begin{pmatrix} x \\ y \\ t \\ z \end{pmatrix} + \begin{pmatrix} \Delta x \\ \Delta y \\ \Delta t \\ 0 \end{pmatrix} \tag{11}$$

where Δt is constrained to be strictly negative.

In this scheme, Δt is set to -1 and the determination of the parameters Δx and Δy is performed by an exhaustive search of the block D_i yielding the best matching of R_i. The exploitation of temporal redundancies during the encoding process is therefore reduced to a 3D block matching.

Even if the transformation τ_i of equation (11) is not contractive, imposing a temporal direction search for D_i ensures convergence during the decoding process. Since Δt is constrained to be strictly negative, regions of the last frame will be coded with self-similarities ensuring therefore the recovering of the fractal approximation of this frame (according to the IFS theory). All regions coded by exploiting temporal redundancies will be defined directly or indirectly from that frame what will allow to retrieve these regions. The last frame can therefore be considered as an intraframe or expressed differently as a reference frame.

4 Adaptive partitioning of the Sequence Support

The motivation for using an adaptive partitioning in the context of fractal sequence coding is twofold. First, flexibility in terms of bitrate is obtained. Any budget constraint or picture quality can be reached by controlling the criterion parameter. Then, such a partition is image-dependent and allows to vary in an optimal way the size of the blocks according to their statistical properties, both temporal and spatial.

The partitioning is produced recursively from top to bottom. The root of the tree represents the entire sequence and the leaves symbolize the partition blocks. The decision to split a block is based on a maximum mean-square-error allowance of its approximation by $\tau_i(D_i)$. The split can either be performed along the temporal axis or along the spatial axes. Temporal split will produce two subblocks whose temporal size is half the one of the block of the previous hierarchical level but whose spatial sizes remain the same. The spatial split will generate four subblocks whose spatial sizes are divided by two but whose temporal size matches the size of the block at the previous level. This procedure is illustrated in figure 2.

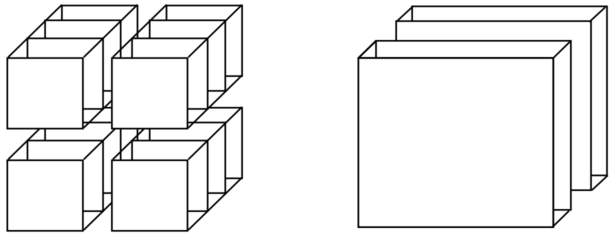

Figure 2: Spatial Split versus Temporal Split

The choice between the split alternatives rests on a try-and-decide procedure. The global mean-square-error in approximating the four subblocks resulting from the spatial split is compared to the mean-square-error of the approximation of the two subblocks generated by the temporal split. The best option is then selected. In a formal way, this gives:

$$split = \begin{cases} Spatial & if \quad \sum_{i=1}^{4} d(\tau_i(D_i^s), R_i^s) < \sum_{j=1}^{2} d(\tau_j(D_j^t), R_j^t) \\ Temporal & otherwise \end{cases}$$

where the superscript s (resp. t) denotes blocks resulting from a spatial (resp. temporal) split. The top-down approach does not lead to the optimal partition of the sequence support but only to a *suboptimal* solution. The optimum path does not indeed correspond necessarily to the path containing the *locally* optimal nodes.

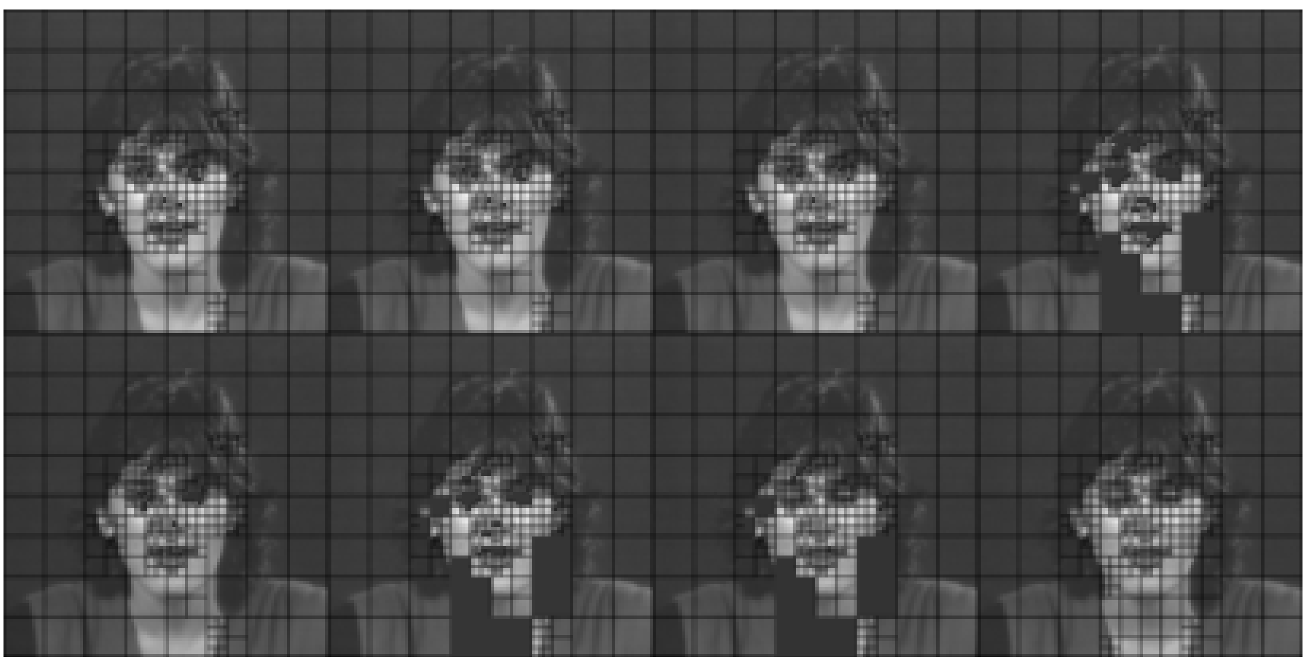

Figure 3: 3D partition of the sequence support. Black regions point at locations of temporal split

This partitioning is attractive for its simplicity and for the low cost of its representation. Describing the partition requires only two bits per node. One to differentiate internal nodes from the leaves and one to specify the split direction. The result is that the partition code constitutes a negligible part of the final bitstream while improving drastically the performance of the system.

5 Simulation Results

Simulations have been run on 8 frames of the Miss America sequence (256x256, 8 bit/pel, 25 frames/sec). The encoding data (i.e. the transform parameters) must be mapped into a finite number of codewords in order to generate the bitstream representing the compressed data. The quantization stage is inserted in the optimization of the transform parameters. In order to take into account the non uniform probability distribution of the transform parameters after quantization, an arithmetic coder of order zero [7] has been used. The addresses of the domain blocks are represented relatively to the addresses of the corresponding range blocks. The brightness shift parameters Δg are coded using an entropy coder conditioned on the corresponding contrast scaling factor α i.e. by using $P(\Delta g|\alpha)$. This operation is motivated by the high correlation between those two parameters as confirmed by the histogram shown in figure 4.

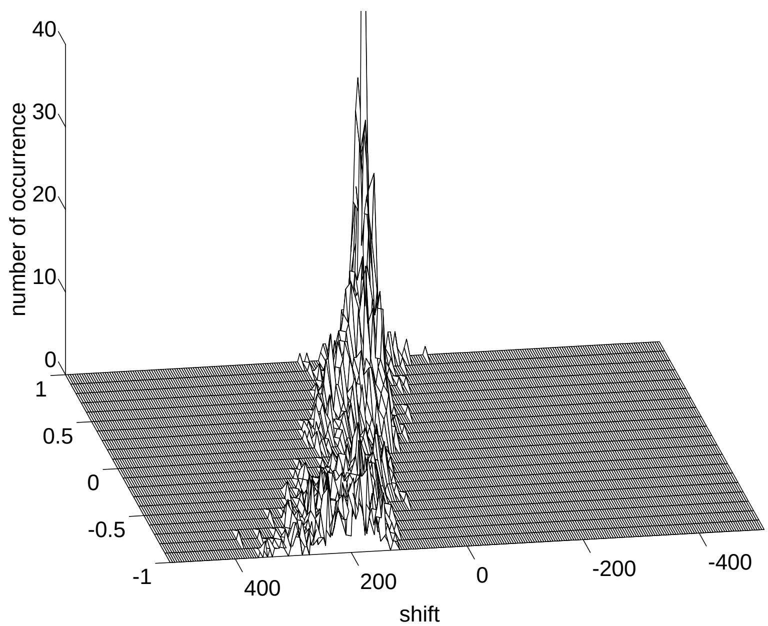

Figure 4: Histogram of $(\alpha, \Delta g)$

	Δt	Δx	Δy	*Isometry*	α	Δg	*Total (bytes)*
Spatial Self-similarities (516 blocks)	-	260	262	171	296	466	1455
Temporal Redundancies (131 blocks)	69	19	37	-	-	-	125
Partition Representation	-	-	-	-	-	-	173

Table 1: Distribution of the bitstream. Simulations correspond to compression of Miss America sequence with a average reconstructed PSNR of 31.44 dB

The eight frames of the sequence have been coded with an average peak signal to noise ratio (PSNR) of 31.44 dB at .0267 bit/pel which corresponds to a compression ratio of 300. For this simulation the maximum mean-square-allowance has been set to 100. Table 1 shows the distribution of the bitstream. As expected, it turns out that the partition code constitutes a negligible part of the final bitstream (about 10%) compared with the efficiency of the adaptive partitioning. It should be remembered that, while simulations have been performed on eight frames, the compressions achieved generally increase substantially with sequence length.

Decoding was performed in 15 iterations starting from 8 identical frames of cameraman. The convergence curve, giving the behavior of the PSNR versus the number of iterations, is shown in figure 5. We must note that the number of iterations can be highly reduced by starting from a decoded group of pictures belonging to the same scene. Indeed, according to equation (4) the convergence will be accelerated since $d(\mu_{orig}, \mu_0)$ decreases.

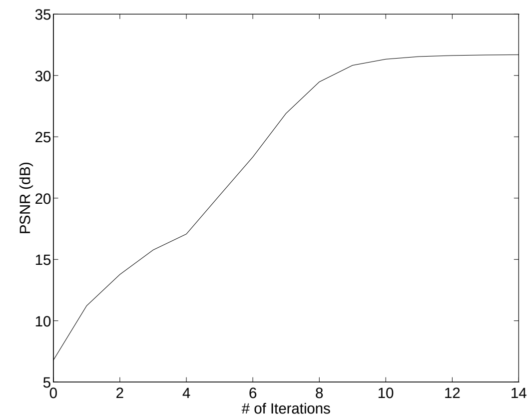

Figure 5: Convergence curve for the decoding process

6 Discussion

In this section, we will briefly analyze the reasons for the success of the scheme described above. This short analysis is realized in comparison with traditional motion compensation techniques using block matching.

- No intraframe needs to be coded independently to the other frames. Even if the last frame can not be coded by using temporal information (for convergence requirement as stated above), regions of this frame will be coded within 3D blocks using therefore the temporal correlation.

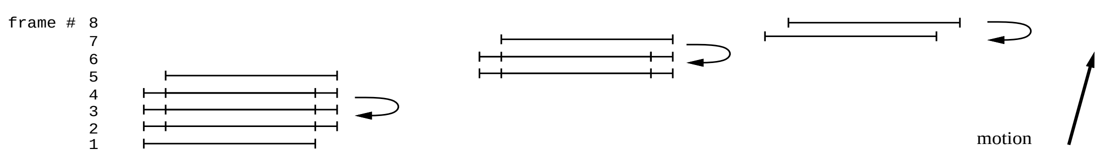

Figure 6: Reduction of spatial redundancies by 3D block matching

- The adaptive partitioning allows to vary in an optimal way the size of the blocks according to their statistical properties both temporal and spatial. In the context of the top-down partitioning, a path in the tree will end when encountering one of these categories of 3D regions:
  - Regions which are both nearly uniform and non-moving. These areas will be approximated by using the spatial self-similarities. Regions very large both temporally and spatially can be very cheaply coded this way.
  - Regions which are both intricate and non-moving. These regions will be coded using temporal redundancies.
  - Regions with uniform translational motion. These ones will be coded by using temporal redundancies.

Using 3D blocks in the context of exploiting temporal redundancies can be seen as multiple compensations of the *vector field* after motion compensation by a traditional block matching algorithm. This comparison is illustrated in figure 6. Let us assume we deal with a region following a uniform translational motion. Regions of the frames 1, 2, 3 and 4 will be approximated by regions belonging to frames 2, 3, 4 and 5; The corresponding regions of the frames 5 and 6 will be represented by regions belonging to frames 6 and 7 etc.. This assertion is confirmed by the way the sequence support is partitioned as shown figure 6.

7 Conclusions and Future Work

In this paper, a new sequence coding scheme has been presented. The system is based on a tridimensional adaptive partitioning of the sequence support. The 3D partition blocks can be coded either by using spatial self-similarities or by exploiting temporal redundancies. Simulations show very good results.

The system presented in this paper is a basic scheme on which several improvements are susceptible to be added. In the last part of this section, some possible improvements which are currently under investigation will be briefly presented.

During the encoding process described above, the choices between different alternatives were based on a comparison in terms of mean-square-error. However a rate-distortion based criterion would be more valid in the context of sequence coding. Such a criterion is used in [8] for determining the best wavelet packet basis with optimal associated quantizers. It consists in comparing the improvement in terms of mean square error with respect to the increasing request in bitrate:

$$\Delta D / \Delta R > \lambda \tag{12}$$

where D denotes the distortion and R the amount of bits required. The parameter λ can be viewed as a quality factor. The reconstructed image quality increases with a decrease in λ. This criterion can be used at three different points of the algorithm:

- When deciding to split a block, the improvement in terms of quality should be related to the increase in bit requirement. The decision to split a parent node into child nodes would be based on formula (12) where

$$\Delta D = D^p - \sum_{i=1}^{n} D_i^c \text{ and } \Delta R = \sum_{i=1}^{n} R_i^c - R^p \tag{13}$$

  where the superscript p (resp. c) stands for the parent node (resp. the child nodes), and n equals 2 or 4 according to the split.

- The number of blocks produced by a spatial split is twice the number of blocks generated by a temporal split. This means that that the amount of parameters to code is potentially doubled. Therefore, in the context of a rate-distortion criterion, the decision to split either temporally or spatially would rely on formula (12) where

$$\Delta D = \sum_{j=1}^{2} D_j^t - \sum_{i=1}^{4} D_i^s \text{ and } \Delta R = \sum_{i=1}^{4} R_i^s - \sum_{j=1}^{2} R^t j \tag{14}$$

  where the superscript t (resp. s) stands for blocks resulting from a temporal (resp. spatial) split.

- As can be seen from table 1, the average number of bits per block when using self-similarities is higher than when exploiting temporal redundancies. Consequently, the decision to use the spatial self-similarities or the temporal redundancies would rather rest on formula (12) where

$$\Delta D = D^{tr} - D^{ss} \text{ and } \Delta R = R^{ss} - R^{tr} \tag{15}$$

  where the superscript tr (resp. ss) denotes the use of temporal redundancies (resp. spatial self-similarities). The use of a rate-distortion criterion in this context is expected to lead to very significant improvements.

As in the fractal-based image coding schemes, the memoryless blockwise partitioning of the sequence support induces blockiness artifacts. These distortions are mainly located near sharp edges under the form of a staircase effect and artifacts taking the form of sharp transitions between adjacent blocks will be visible in smoothly varying areas. In order to overcome these artifacts, a method based on an overlapped partitioning has been built in the context of still image coding and will be presented in a further publication. This technique, which yields an actual improvement in terms of visual quality, can easily be extended to the 3D scheme.

Acknowledgment

The author is grateful to Mohsine Karrakchou and Jean-Marc Vesin for the proof-reading of the manuscript.

References

[1] Mandelbrot B. *The Fractal Geometry of Nature*. Freeman and co, San Francisco, 1982.

[2] M.F. Barnsley. *Fractals Everywhere*. Academic Press Inc., San Diego, 1988.

[3] A.E. Jacquin. Image coding based on a fractal theory of iterated contractive image transformations. *IEEE Transactions on Image Processing*, 1:18–30, January 1992.

[4] Y. Fisher. Fractal image compression. *SIGGRAPH'92 course notes*, 1992.

[5] D. Monro and F. Dudbridge. Fractal block coding of images. *Electronics Letters*, 11:1053–1055, November 1992.

[6] E.W. Jacobs, Fisher Y., and Boss R.D. Image compression: A study of iterated transform method. *Signal Processing*, 29:251–263, December 1992.

[7] M. Nelson. *The data compression book*. M & T Publishing Inc., Redwood City, CA, 1991.

[8] K. Ramchandran and M. Vetterli. Best wavelet packet bases using rate-distortion criteria. *IEEE Int. Symposium on circuits and systems*, 2:971–974, May 1992.