New mapping functions for IFS-based coding on high frequency content images.

Guillaume ROBERT[1], Nathalie LAURENT[1], and Jean-Marc CHASSERY[2]

[1] France Telecom - CNET / DIH, rue du Clos Courtel, 35512 Cesson Sevigne Cedex, FRANCE
[2] Equipe Infodis / TIMC, Grenoble, FRANCE

Abstract. In this paper, we present an evolution of IFS-based image compression schemes, well adapted to high frequency contents. We propose to substitute the usual affine mass map which tends to smooth irregular surfaces, by an harmonic based map. This change implies the creation of a new IFS determination algorithm for solving the inverse problem, and a new way to quantize and encode coefficients.

1 INTRODUCTION

It may be surprising to note that IFS in the family of Jacquin's coding schemes [11] suffer from annoying distortions when dealing with high frequency image contents reconstruction [16] [21]. In fact, some of them are able to generate many rich and different textures [1] at a very high compression rate, but under condition they are fractal [14]. Generally, stationary textures do not possess the self-similarity property on which block-based fractal coding methods rest. Nevertheless, some fractal tools are very efficient for common textures manipulation. This is the case of the multifractal theory [22][17] which has recently drawn much attention as a tool for studying the structure of singular measures, both in theory and in application: analyzing, modifying and classifying textures[13]. This would prove that fractal coders are potentially able to accurately compress rich-contented images.

1.1 Related works

Different directions have been explored, so as to improve IFS coding performance over chaotic surfaces.
Among the early ideas, we can quote the creation of a range blocks partition adapted to the content of each block (the richest, the smallest). This solutions, dramatically improves IFS capabilities, because textured regions are split into small blocks that probability of which to be well mapped is increased. Davoine presents in [9] an overview of all different types of partition that have been used to solve the inverse problem : quadtrees, HV, triangular, polygonal, complex regions.

Another way to get better rate-distortion curves consists in mixing IFS with other coding methods. These hybrid schemes, provide satisfactory results. Techniques using this idea are described in [6][3].
Barthel defined in [2], a fractal based transform coding in which an IFS is computed over DCT coefficients of small square blocs. This approach combines the interest of both methods: DCT expresses intra-block redundancies while IFS expresses inter-bloc redundancies.
Novak chooses another kind of IFS-DCT hybridizing, proposing a model-residual algorithm[20]. First the input image is approximated by a parametric model (polynomial basis or DCT), then the residual part of the signal is encoded using an IFS. The advantage of this method is its robustness to low bitrate and noisy channels.
It has also been proven that fractal coders are able to point out likely relationships existing between the different resolutions of a subbands or wavelets image decomposition. Davis[5] and Krupnik[12] propose to express those likely relationships between square range blocks in a subband, and square domain blocks in another subband at a lower resolution. Contrary to DCT hybridization those methods do not create block effects at high compression rates.
Unlike the previous methods, VQ-IFS hybridization is not developed in a transformed space. The IFS is seen as an auto-quantizing process and a vectorial quantizing is applied to the domain blocks set [19] [7] i.e. similar domain blocks are merged.

Compared with other elements of the IFS coding chain, map functions have not been widely studied. Only a few works attempt to optimize maps which are affine functions for most of the time. In the early years of fractal coders, Monro who wanted to bypass a domain-range embedding constraint, proposed [18] the use of polynomial map function. More recently Lutton introduced nonaffine tree-based map functions and resorts to a genetic programming method for solving this much more complex inverse problem developed in [15].

1.2 Main contribution

Considering textures to be the IFS pitfall, and observing the efficiency of DCT methods, we address this IFS drawback by considering harmonic based maps. The main advantage of our approach over others IFS-DCT hybridization is that both methods are not chained but are really mixed and conjointly optimized.

1.3 Paper organization

In section 2 the framework of our method is looked at. Section 3 details the resolution of the inverse problem. First the range blocks partition building

method is exposed, and then our map function family is defined : a one-pixel precise domain block positioning and harmonic functions based map are presented. Finally, section 4 concludes the preliminary results and outlines future works.

2 OVERVIEW

Our work is divided into three parts. The first part is the range blocks partition building. This step consists in the creation of triangular blocks, possibly fitting the image content. Textured parts are covered with small blocks. This part is further detailed in [8] [10].
The second part, is the main evolution we propose for improving IFS capabilities on disturbed regions. Contrary to usual mapping models resting on affine functions, we introduce harmonic functions based model. The mapping error is defined as the L_2 distance between the original range block and its associated decimated domain block whose surface is projected in an harmonic functions basis, in the way of a DCT transform. Only a few coefficients are needed for a good result. Moreover, domain positions are computed so as to reach a one-pixel precise position.

3 GENERAL ALGORITHM

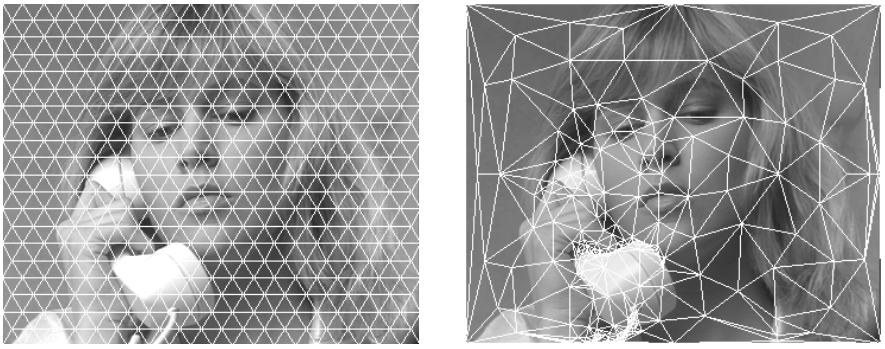
The inverse problem remains unchanged since the early IFS coders. Let us consider a given grey-level image I whose support is split into n_R range blocks $\{\mathcal{R}^i\}$. Each range block has to be associated by α to a unique and often larger domain block $\mathcal{D}^{\alpha(i)}$ among n_D possibly overlapping domain blocks $\{D^j\}$ so that there is a map $\omega^{\alpha(i)}$whose image $\hat{\mathcal{R}}^{\alpha(i)}$ of $\mathcal{D}^{\alpha(i)}$ is close to $\mathcal{R}^i$. The union of the n_R chosen maps is named IFS W. In the case W is eventually contractive over the metric space $(\mathbb{N}^{\mathbb{N}^2}, d)$ of grey-level images I, W has a unique grey-level image $\hat{I}$ as afixed point. Resolving the inverse problem for I, means finding out an IFS W whose fixed point $\hat{I}$ is so that :

$$d(I, \hat{I}) \;=\; 0$$

According to the collage theorem [4], this property can be verified if

$$\forall i \in [0..n_R[\; d(\mathcal{R}^i\;, \hat{\mathcal{R}}^{\alpha(i)}\;) = 0 \qquad (1)$$

The L_2 distance is commonly considered for image compression with losses. Nevertheless, a zero distance is too strong a constraint and only a minimization of this distance is performed.



a. Regular triangle range blocs. **b.** Content adaptative Delaunay range blocs.

Fig. 1. Our IFS algorithm, still presents content adaptative range blocks .

3.1 Range blocks creation

Our algorithm is based on two types of range partition construction. The first one (figure 1a) provides a low cost regular triangles partition building. Its use is advantageous on homogenous content images such as *Radar256* (figure 3). The second mode (figure 1b) allows the creation of an adaptative Delaunay triangulation, whose blocks are larger on flat zones.

3.2 Spatial map

The determination of a map $\omega^{\alpha(i)}$can be broken down into two steps. The first one creates a decimated domain block $\bar{\mathcal{D}}^{\alpha(i)}$ through an affine spatial transform $\omega_S^{\alpha(i)}$[23]. The second one is the mass map $\omega_M^{\alpha(i)}$and creates $\hat{\mathcal{R}}^{\alpha(i)}$ by modifying $\bar{\mathcal{D}}^{\alpha(i)}$ surface.

The decimated domain block $\bar{\mathcal{D}}^{\alpha(i)}$ is a subsample of the domain block $\mathcal{D}^{\alpha(i)}$ associated with the range block $\mathcal{R}^i$. Our spatial transform family $\mathcal{F}$ is a subset of affine transforms, disabling rotations and letting the domain position free in the vicinity of $\mathcal{R}^i$. The domain pool $\mathcal{P}^i$ associated to $\mathcal{R}^i$ is defined as $\mathcal{P}^i = \{D \subset I,\ \exists \omega \in \mathcal{F} \setminus support(\mathcal{R}^i) = support(\omega(D))\}$. On the one hand, one may think the size of $\mathcal{P}^i$ will increase the amount of information for a domain block characterization and thus reduce the coder performance. That is true, but on the other hand this offers high precision maps, which is of a great interest concerning edges, and advantageously balance the increased cost.
The parameters of the spatial transform are the spatial contraction vector $\boldsymbol{\mathcal{C}}^{\alpha(i)} \in \mathbb{R}^2$ and the spatial translation vector $\boldsymbol{\mathcal{T}}^{\alpha(i)} \in \mathbb{N}^2$. To express the $\bar{\mathcal{D}}^{\alpha(i)}$ content, let us introduce subsets $\mathcal{B}^{\alpha(i)q} \subset \mathcal{D}^{\alpha(i)}$ associated to a pixel

$q \in \bar{\mathcal{D}}^{\alpha(i)}$ and defined as follows:

$$\mathcal{B}^{\alpha(i)q} = \{p \in \mathcal{D}^{\alpha(i)} \setminus \begin{pmatrix} (\bar{D}_q^{\alpha(i)})_x \\ (\bar{D}_q^{\alpha(i)})_y \end{pmatrix} = \begin{pmatrix} \frac{1}{\mathcal{C}_x^{\alpha(i)}} & 0 \\ 0 & \frac{1}{\mathcal{C}_y^{\alpha(i)}} \end{pmatrix} \times \begin{pmatrix} p_x \\ p_y \end{pmatrix} + \begin{pmatrix} \mathcal{T}_x^{\alpha(i)} \\ \mathcal{T}_y^{\alpha(i)} \end{pmatrix}\}$$

Thus the value of a q pixel in the decimated block $\bar{\mathcal{D}}^{\alpha(i)}$ is

$$\bar{D}_q^{\alpha(i)} = \frac{\sum_{p \in support(\mathcal{B}^{\alpha(i)q})} \mathcal{B}_p^{\alpha(i)q}}{card(\mathcal{B}^{\alpha(i)q})}$$

3.3 Affine mass map

The mass map is the transformation through which the surface of a decimated domain block $\bar{\mathcal{D}}^{\alpha(i)}$ is moved in order to approximate the surface of its associated range block $\mathcal{R}^i$. The usual constraint on the mass map $\omega_M^{\alpha(i)}$is its minimization of the collage error $E^{\alpha(i)}$ in the sense of the L_2 distance :

$$E^{\alpha(i)} = d(\mathcal{R}^i, \omega_M^{\alpha(i)}(\bar{\mathcal{D}}^{\alpha(i)})) = d(\mathcal{R}^i, \hat{\mathcal{R}}^{\alpha(i)}) \quad (2)$$

When using an affine mass map, the collage error has the expression:

$$E^{\alpha(i)} = \sum_{p \in support(\mathcal{R}^i)} (s \times \bar{\mathcal{D}}_p^{\alpha(i)} + o - \mathcal{R}_p^i)^2 \quad (3)$$

Where $s \in \mathbb{R}$ and $o \in \mathbb{N}$ are named scale and offset of the map. The minimum value for $E^{\alpha(i)}$ according to equation 3 can be computed by resolving the two equationed linear system resulting from $E^{\alpha(i)}$ derivations.

3.4 Cosine mass mapping

Instead of an affine mass map mentioned in 3.3, and in order to thoroughly consider high frequency surfaces, we introduce a cosine mass mapping. This map $\omega_M^{\alpha(i)}$depends on a basis of N_c fixed pulsation cosine functions, whose amplitudes have to be computed.
The expression of the value of pixel p in the approximated block $\hat{\mathcal{R}}^{\alpha(i)}$ is defined as:

$$\begin{aligned} \hat{\mathcal{R}}_p^{\alpha(i)} &= \omega_M^{\alpha(i)}(\bar{\mathcal{D}}_p^{\alpha(i)}) \\ &= b + \sum_{l \in [0..N_c[} \sum_{k \in [0..N_c[} c_{lk}^{\alpha(i)} \times \bar{\mathcal{D}}_p^{\alpha(i)} \times \cos(\theta_l \times p_x) \times \cos(\theta_k \times p_y) \end{aligned} \quad (4)$$

Where $b \in \mathbb{Z}$ is the offset, $\theta_l = \frac{2\pi}{2^l}$ are the fixed pulsations, $(p_x, p_y) \in \mathbb{N}^2$ stands for the relative position of pixel p in $\bar{\mathcal{D}}^{\alpha(i)}$ and $c^{\alpha(i)} \in \mathcal{M}^{N_c}(\mathbb{R})$ is a matrix of cosine amplitudes. One can notice that $\theta_0 = 2\pi$ which means that $c_{00}^{\alpha(i)}$ from equation 4 is equivalent to the scale factor of equation 3.
The map error between the original range block $\mathcal{R}^i$ and the appoximated range block $\hat{\mathcal{R}}^{\alpha(i)}$ is then deduced from equations 2 and 4 :

$$E^{\alpha(i)} = \sum_{p \in support(\mathcal{R}^i)} ((\sum_{k \in [0..N_c[} \sum_{l \in [0..N_c[} c_{kl}^{\alpha(i)} \times \bar{\mathcal{D}}_p^{\alpha(i)} \times \cos(\theta_k \times p_x) \times \cos(\theta_l \times p_y) \ ) - \mathcal{R}_p^i \ )^2 \tag{5}$$

The determination of the $c_{kl}^{\alpha(i)}$ coefficients minimizing $E^{\alpha(i)}$ is performed by resolving the following equations system :

$$\begin{cases} \frac{\partial E^{\alpha(i)}}{\partial c_{00}^{\alpha(i)}} = 0 \\ \vdots \\ \frac{\partial E^{\alpha(i)}}{\partial c_{N_c-1 N_c-1}^{\alpha(i)}} = 0 \\ \frac{\partial E^{\alpha(i)}}{\partial b} = 0 \end{cases} \tag{6}$$

After simplifications, 6 can be written as a linear matrix equation :

$$A \begin{pmatrix} A_{00}^{00} & \cdots & A_{0N_c-1}^{00} & A_{10}^{00} & \cdots & A_{N_c0}^{00} \\ \vdots & & & & & \\ A_{00}^{0N_c} & & & & & \\ A_{00}^{1N_c} & & & & & \\ \vdots & & & & & \\ A_{00}^{N_c0} & \cdots & A_{0N_c-1}^{N_c0} & A_{10}^{N_c0} & \cdots & A_{N_c0}^{N_c0} \end{pmatrix} * X \begin{pmatrix} c_{00} \\ \vdots \\ c_{N_c-1N_c-1} \\ b \end{pmatrix} = B \begin{pmatrix} B_{00} \\ \vdots \\ B_{N_c-1N_c-1} \\ B_{N_c-1N_c} \end{pmatrix} \tag{7}$$

If we introduce the function $\psi_p^{kl} = cos(\theta_k \times p_x) \times cos(\theta_l \times p_y)$, then coefficients of the known matrix $A \in \mathcal{M}^{N_c^2+1}(\mathbb{R}\ )$ are :

$$\begin{cases} A_{mn}^{kl} = \sum_{p \in support(\mathcal{R}^i)} (\bar{D}_p^{\alpha(i)})^2 \times \psi_p^{mn} \times \psi_p^{kl}, & \forall k \in [0..N_c[\ , \ \forall l \in [0..N_c[\ , \\ & \forall m \in [0..N_c[\ , \ \forall n \in [0..N_c[\\ A_{mn}^{N_cN_c} = \sum_{p \in support(\mathcal{R}^i)} \bar{D}_p^{\alpha(i)} \times \psi_p^{mn}, & \forall m \in [0..N_c[\ , \ \forall n \in [0..N_c[\\ A_{N_c0}^{kl} = \sum_{p \in support(\mathcal{R}^i)} \bar{D}_p^{\alpha(i)} \times \psi_p^{kl}, & \forall k \in [0..N_c[\ , \ \forall l \in [0..N_c[\\ A_{N_c0}^{N_cN_c} = card(\mathcal{R}^i\ ) \end{cases}$$

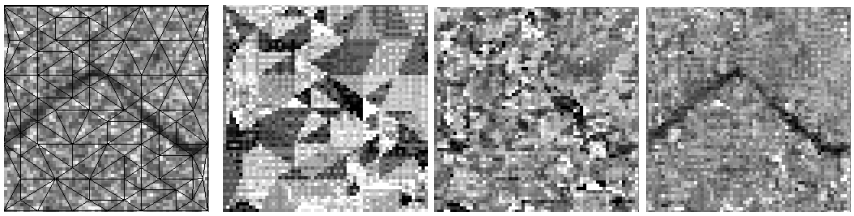
and coefficients of the known vector $B \in \mathbb{R}^{N_c^2+1}$ are:

$$\begin{cases} B_{kl} = \sum\limits_{p \in support(\mathcal{R}^i)} \bar{D}^{\alpha(i)}_{\beta(p)} \times R_p \psi_p^{kl}, \forall k \in [0..N_c[\ ,\ \forall l \in [0..N_c[\\ B_{N_c-1N_c} = \sum\limits_{p \in support(\mathcal{R}^i)} R_p \end{cases}$$

The solution of this system can be performed using an LDL^t factorization.

4 PRELIMINARY RESULTS AND CONCLUSION

In this paper we proposed the use of a one-pixel precise domain block positioning associated with an image content range partition building. These facilities ensure IFS capabilities on contour reconstruction. We also introduced a new kind of mass map, resting on a domain block projection over an harmonic basis in order to enable a better reconstruction of textures.



a. Original + Mesh **b.** Iteration 1 **c.** Iteration 2 **d.** Iteration 9

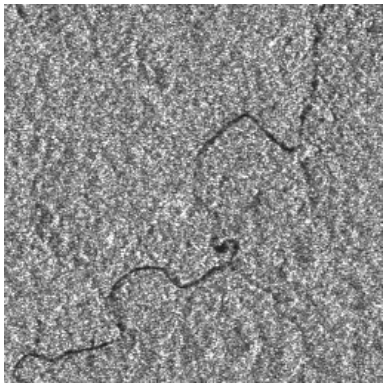
Fig. 2. Iterations of a cosine-IFS at a rate of 0.5bpp.

Figure 2 and 3 illustrates the algorithm behaviour over a textured image. One can notice that high frequency appears at first iteration. Precise details like the river are kept while conserving the general texture appearance with no block effects. A classic affine based IFS (figure 3.c) succeeds in reconstructing the river, but tends to smooth out the texture.

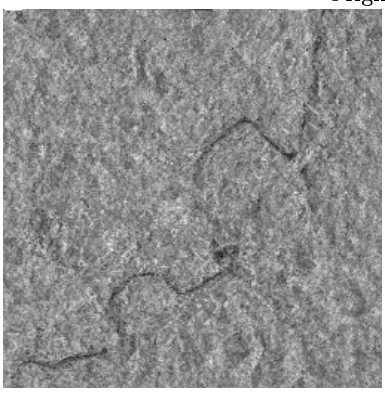
The total CPU time for encoding this image with our algorithm is about $5min$ on a *PII350* with 2 different cosine pulsations (i.e. 5 coefficients to fix). Nevertheless, the use of 3 or more cosine pulsations leads to prohibitive computing times. Future works will solve this problem by an optimized prioritized hierarchical scanning. We also plan to improve cosine amplitudes quantization.

References

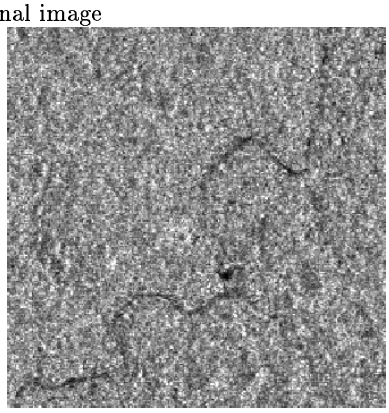
1. M. F. Barnsley, *Fractal everywhere*, Academic Press, New York, 1988.



a. Original image



b. Affine-IFS at 0.31*bpp*



c. Cosine-IFS at 0.30*bpp*

Fig. 3. Comparison of algorithm performance at low bitrate on an highly textured image.

2. K. U. Barthel, J. Schüttemeyer, T. Voyé, and P. Noll, *A new image coding technique unifying fractal and transform coding*, IEEE International Conference on Image Processing (Austin, Texas), November 1994, pp. 112–116.
3. P. Bas, *Hybridation de mthodes en compression d'images*, Master's thesis, INPG, 1997.
4. B. Bielefeld and Y. Fisher, *A convergence model*, pp. 215–228, Springer Verlag, New York, 1995.
5. G. Davis, *Adaptive self-quantization of wavelet subtrees: a wavelet-based theory of fractal image compression*, SPIE conference on mathematical imaging: wavelet applications in signal and image processing (1995).
6. F. Davoine, *Compression d'images par fractales basée sur la triangulation de Delaunay*, Thèse de l'Institut National Polytechnique de Grenoble, France (1995).
7. F. Davoine, M. Antonini, J.-M. Chassery, and M. Barlaud, *Fractal image compression based on Delaunay triangulation and vector quantization*, IEEE Trans-

actions on Image Processing - Special Issue on Vector Quantization (1996).
8. F. Davoine and J.-M. Chassery, *Adaptive delaunay triangulation for attractor image coding*, ICPR (Jerusalem), 1994, pp. 801–803.
9. F. Davoine, G. Robert, and J.M. Chassery, *How to improve pixel-based fractal image coding with adaptative partitions*, Fractals In Engineering (J Lévy Véhel ans E. Lutton and C. Tricot, eds.), Springer-Verlag, 1997, pp. 292–306.
10. H. Le Floch, *Echantillonnage irrgulier et interpolation: application la reprsentation d'images fixes et animes*, Ph.D. thesis, SPM/IRISA, 1997.
11. A. E. Jacquin, *Image coding based on a fractal theory of iterated contractive image transformations*, IEEE Transactions on Image Processing **1** (1992), no. 1, 18–30.
12. H. Krupnik, D. Malah, and E. Karnin, *Fractal representation of images via the discrete wavelet transform*, IEEE 18th Conv. of EE in Israel (Tel-Aviv), March 1995.
13. Y. Lechevallier and J. Levy Vehel, *Arthur: a system for texture analysis*, Proceedings of RFIA (1991), 1185–1193.
14. J. Lévy-Véhel, *Texture analysis using fractal probability functions*, Tech. Report 1707, INRIA - Rocquencourt, France, 1992.
15. E. Lutton, J. Lévy-Véhel, G. Cretin, P. Glevarec, and C. Roll, *Mixed ifs: resolution of the inverse problem using genetic programming*, Complex Systems, vol. 5, 1995, pp. 375–398.
16. B.B. Mandelbrot, *The fractal geometry of nature*, W. H. Freeman and Co., San Francisco, 1982.
17. ———, *A class of multinomial multifractal measures with negative (latent) values for the dimension $f(\alpha)$.*, Fractals' Physical Origin and Proprieties (1989), 3–29.
18. D. M. Monro, *Class of fractal transforms*, Electronics Letters **29** (1993), no. 4, 362–363.
19. M. Muller, R. Hamzaoui, and D. Saupe, *Vq-enhanced fractal image compression*, ICIP, 1996.
20. M. Novak, *Attractor coding of images*, Licentiate Thesis, Department of Electrical Engineering, Linköping University (1993).
21. G.E. Øien, R. Hamzaoui, and D. Saupe, *On the limitations of fractal image texture coding*, Proceedings of ICASSP.
22. R. Riedi, *An improved multifractal formalism and self-similar measures.*, Journal of Approximation Theory (1995), 189.
23. G. Robert and J.M. Chassey, *Compression d'images par fractales base de maillages polygonaux de vorono*, Seizime Colloque GRETSI (Grenoble, France), vol. 2, 1997.