INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Structure and texture compression

Jean-François Aujol — Basarab Matei

N° 5076

January 2004

THÈME 3

Rapport de recherche

ISSN 0249-6399 ISRN INRIA/RR--5076--FR+ENG

Structure and texture compression

Jean-François Aujol [*] [†] , Basarab Matei [‡] [§]

Thème 3 — Interaction homme-machine,
images, données, connaissances
Projet Ariana

Rapport de recherche n° 5076 — January 2004 — 29 pages

Abstract: In this paper, we tackle the problem of image compression. During the last past years, many algorithms have been proposed to take advantage of the geometry of the image. We intend here to propose a new compression algorithm which would take into account the structures in the image, and which would be powerful even when the original image has some textured areas. To this end, we first split our image into two components, a first one containing the structures of the image, and a second one the oscillating patterns. We then perform the compression of each component separately. Our final compressed image is the sum of these two compressed components. This new compression algorithm outperforms the standard biorthogonal wavelets compession.

Key-words: Bounded variation, image decomposition, image compression, wavelets, edge adapted technique, structure, texture.

[*] également membre du laboratoire J.A Dieudonné, Université de Nice-Sophia-Antipolis
[†] jfaujol@sophia.inria.fr
[‡] Laboratoire J.L. Lions , UMR CNRS 7598, Université Paris VI, Boîte courrier 187, 4 place Jussieu, 75013 Paris, France
[§] matei@ann.jussieu.fr

Unité de recherche INRIA Sophia Antipolis
2004, route des Lucioles, BP 93, 06902 Sophia Antipolis Cedex (France)
Téléphone : +33 4 92 38 77 77 — Télécopie : +33 4 92 38 77 65

Compression des structures et des textures d'une image

Résumé : Dans ce papier, nous nous intéressons au problème de la compression d'image. Les ondelettes se sont révélées être un outil particulièremment efficace [13]. Récemment, de nombreux algorithmes [6, 10, 12, 14, 9, 8, 2] ont été proposés pour amméliorer la compression par ondelettes en essayant de prendre en compte les strucutres présentes dans l'image. De telles méthodes se révèlents très efficaces pour les images géométriques.

Nous construisons un algorithme de compression d'images qui prend en compte la géométrie de l'image tout en étant capable d'être performant sur des images contenant à la fois des structures et des textures. Pour cela, nous utilisons un algorithme de décomposition d'image récemment introduit dans [4, 3]. Cet algorithme permet de séparer une image en deux composantes, une première composante contenant l'information géométrique de l'image, et une deuxième contenant les éléments oscillants de l'image.

L'idée de notre méthode de compression est la suivante. Nous commençons par décomposer l'image à compresser en sa partie géométrique et sa partie oscillante. Nous effectuons ensuite la compression de la partie géométrique à l'aide de l'algorithme introduit dans [14, 9, 8, 2], ce dernier étant particulièrement bien adapté pour la compression des structures d'une image. Pour la partie oscillante de l'image, nous utilisons l'algorithme classique de compression par ondelettes biorthogonales. Notre nouvel algorithme de compression s'avère plus performant que la méthode classique par ondelettes biorthogonales.

Mots-clés : Fonctions à variations bornées, décomposition d'image, compression d'images, ondelettes, structure, texture.

Contents

1 Introduction

Image compression is a very active field of research. The aim is to find a sparse representation for an image, so that one needs to store few coefficients to retrieve the image.

Wavelets have proved to be a powerful tool in this area [13]. They indeed provide sparse representations for images (especially for texture images). But (see [13] for instance), edges lead to many significant wavelet coefficients. That is why there have been a lot of works these last past years to find sparse representations of images that would better take into account the geometry of the images. Some algorithms [6, 10, 12, 14, 9, 8, 2] have thus been proposed to improve wavelets compression by taking advantage of the structures in the image. Such methods have proved to be efficient for geometrical images.

We want to propose a framework which would extend these methods to images containing both structure and texture information. Stucture compression and texture compression are not done with the same algorithms. Therefore, we first want to split the image to compress into a geometrical part and a texture part, and then to carry out the compression of both components separately. To achieve our goal, we are going to use a decomposition algorithm recently introduced in [18, 4, 3]. Such an approach has been successfully introduced in [5] for image inpainting.

The plan of the paper is as follows. We first present the decomposition model of [4, 3]: we recall its link with Meyer's model [15], and show some numerical examples. We then introduce the compression algorithm of [14]: it is well adapted for geometrical images. For textured images, we use the classical 7-9 biorthogonal wavelets [13]. This way, we are in position to describe our new compression algorithm. We show some numerical results to illustrate the relevance of our method. We end the paper with a short discussion.

2 A decomposition algorithm

In [15], Meyer has discussed the classical Rudin-Osher-Fatemi model [17]. He has introduced a new model to split a given image f into a sum $u+v$ of a bounded variation component and a component containing the oscillating part of the image. This model has first been successfully implemented by Vese and Osher [18]. A different approach has been proposed in [4, 3]. We will use this one, where the decomposition is computed by minimizing a convex functional which depends on the two variables u and v, alternatively in each variable. Each minimization is based on a projection algorithm to minimize the total variation [7]. See also [16] for another interesting approach.

2.1 Modelling

Let Ω be a bounded open set of $\mathbb{R}^2$ with Lipschitz boudary. The space used to model the geometrical component u of an image f is the space BV of functions with bounded variation. We recall here its definition [7]:

Definition: $BV(\Omega)$ is the subspace of functions $u \in L^1(\Omega)$ such that the following quantity is finite:

$$\begin{aligned} J(u) \;=\; \sup \Big\{ \int_\Omega u(x) \operatorname{div}(\xi(x))dx, \\ \xi \in C_c^1(\Omega;\mathbb{R}^2), \|\xi\|_{L^\infty(\Omega)} \leq 1 \Big\} \end{aligned} \qquad (1)$$

$BV(\Omega)$ endowed with the norm $\|u\|_{BV(\Omega)} = \|u\|_{L^1(\Omega)} + J(u)$ is a Banach space. If $u \in BV(\Omega)$, the distributional derivative Du is a bounded Radon measure and (1) corresponds to the total variation $|Du|(\Omega)$.

In [15], Meyer has proposed a new decomposition model:

$$\inf_{(u,v)\in BV(\Omega)\times G(\Omega)/f=u+v} \left(J(u) + \alpha \|v\|_G \right) \qquad (2)$$

The Banach space $G(\Omega)$ contains signals with large oscillations, and thus in particular textures and noise. The G-norm replaces the classical L^2 norm of the Rudin-Osher-Fatemi model [17]. We give here the definition of $G(\Omega)$:

Definition: $G(\Omega)$ is the Banach space composed of the distributions f which can be written

$$f = \partial_1 g_1 + \partial_2 g_2 = \operatorname{div}(g) \qquad (3)$$

with g_1 and g_2 in $L^\infty(\Omega)$. On G, the following norm is defined:

$$\begin{aligned}\|v\|_G &= \inf \Big\{ \|g\|_{L^\infty(\Omega)} = \operatorname*{ess\,sup}_{x\in\Omega} |g(x)| \; / v = \mathrm{div}\,(g), \\ & g = (g_1, g_2),\;\; g_1 \in L^\infty(\Omega), g_2 \in L^\infty(\Omega), \\ & |g(x)| = \sqrt{|g_1|^2 + |g_2|^2}(x) \Big\}\end{aligned} \tag{4}$$

A function belonging to G may have large oscillations and nevertheless have a small norm.

2.2 Functional

In [4, 3], the authors have introduced the following functional:

$$\inf_{(u,v)\in BV(\Omega)\times G_\mu(\Omega)} \left(J(u) + \frac{1}{2\lambda}\|f-u-v\|^2_{L^2(\Omega)} \right) \tag{5}$$

where

$$G_\mu(\Omega) = \{v \in G(\Omega)/\|v\|_G \le \mu\} \tag{6}$$

The parameter λ controls the L^2-norm of the residual $f-u-v$. The smaller it is, the smaller the L^2-norm of the residual gets. And μ controls the G-norm of the oscillating part v. It is shown in [4, 3] that solving (5) is a way to solve (2).

The minimum of (5) is computed by minimizing alternatively in each variable u and v. Each minimization is based on a projection algorithm to minimize the total variation [7].

We denote by P_K the orthogonal projection on a set K. When $K = G_\mu$ for some $\mu > 0$, this projection is computed thanks to Chambolle's algorithm [7].

Algorithm:

1. Initialization:

$$u_0 = v_0 = 0 \tag{7}$$

2. Iterations:

$$v_{n+1} = P_{G_\mu}(f - u_n) \tag{8}$$

$$u_{n+1} = f - v_{n+1} - P_{G_\lambda}(f - v_{n+1}) \tag{9}$$

3. Stopping test: we stop if

$$\max(|u_{n+1} - u_n|, |v_{n+1} - v_n|) \leq \epsilon \tag{10}$$

Remark: It is shown in [4, 3] that such an algorithm converges to the solution of problem (5). It is also shown that, if one has chosen a value $\alpha > 0$ in problem (2), then, when the parameter λ decreases to 0, there exists $\mu > 0$ such that problem (5) and problem (2) have the same solutions. Algorithm (7)-(10) is therefore a good way to solve Meyer's problem (2) numerically.

2.3 Numerical experiments

In the results that we present in this subsection, we have decided to add the small residual $f - u - v$ to the geometrical component. This way, we have an exact decomposition, which is a good point since we intend to carry out compression on this decomposition. Figures 1 and 2 are the images which we want to decompose. The numerical results are displaid on Figures 3 and 4.

3 Nonlinear multiscales representations for geometrical images

While standard 2-d wavelets provide sparse representations of smooth textured images, they are not able to exploit the fact that the discontinuities (edges) delineating different textured regions lie along smooth contours, leading to poor algorithmic performance results. This has motivated new directions of research towards compact representations of geometry : curvelets [6], contourlets [10] and bandlets [12]. Edge adapted (EA) multiscale representations introduced and studied in [14, 2] are another possible track for such improvements. This transform has as guideline the discrete 1-d framework of Ami Harten [11].

Figure 1: Barbara image

Figure 2: Phare image

BV component u

Oscillatory component v

Figure 3: Decomposition of the Barbara image ($\lambda = 0.1$ and $\mu = 60$).

BV component u

Oscillatory component v

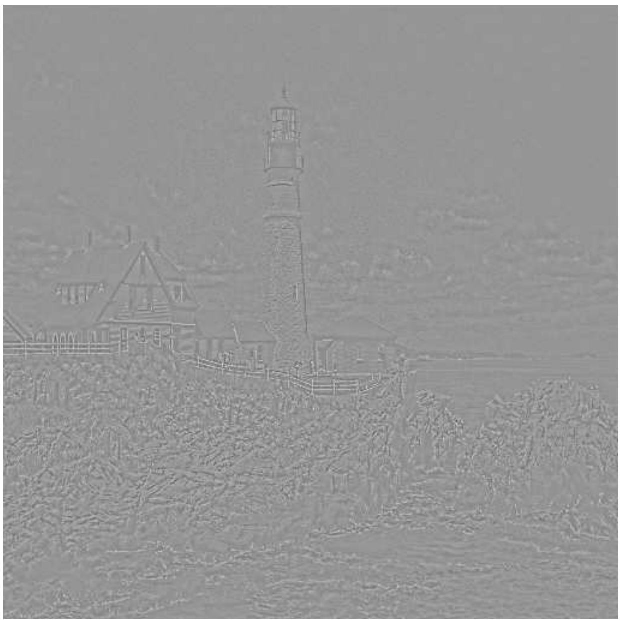

Figure 4: Decomposition of the phare image ($\lambda = 0.1$ and $\mu = 12$).

3.1 Harten's framework

The ideas of A. Harten can be described as follows: we start from a finite set $v^J = (v^J_k)$ of discrete data at the resolution level J. For all j, we define a decimation operator D^{j-1}_j, which extracts from (v^j_k) the discrete data (v^{j-1}_k) at the next coarser level, and a prediction operator P^j_{j-1}, which yields an approximation $(\hat{v}^j_k)$ of (v^j_k) from (v^{j-1}_k). The decimation is always a linear operator, and the prediction is allowed to be a nonlinear operator, but they satisfy the following consistency condition $D^{j-1}_j P^j_{j-1} = I$. Consequently, we represent v^j in terms of (v^{j-1}, e^{j-1}), where e^{j-1} is the prediction error which belongs to the null space of D^{j-1}_j (due to the consistency relation). We represent the error e^j in terms of a basis of the null space of D^{j-1}_j, resulting in the detail vector d^{j-1}. Therefore we can represent v^j by (v^{j-1}, d^{j-1}). By iterating this procedure from the finest level J to the coarsest level $j = 0$ we obtain the multiscale representation of v^J into $(v^0, d^0, ..., d^{J-1})$.

3.2 Edge Adapted multiresolution representations

3.2.1 1-d formalism

For the sake of clarity, we first consider the 1-d case. We consider hierarchical discretizations by cell-averages $v^j_k := 2^j \int_{c^j_k} v(x)dx$, where $c^j_k := [k2^{-j}, (k+1)2^{-j})$. This choice fixes the decimation as $v^{j-1}_k := \frac{1}{2}(v^j_{2k} + v^j_{2k+1})$. Using $e^{j-1}_{2k} + e^{j-1}_{2k+1} = 0$, we define the details by $d^{j-1}_k = e^j_{2k}$.

We now want to define a good prediction operator which makes the approximation error and then the detail coefficents as small as possible, resulting in a sparse representation. The lazzy choice for a such an operator is $\hat{v}^j_{2k} = \hat{v}^j_{2k+1} = v^{j-1}_k$, which corresponds to the decomposition in the Haar system with a low order of accuracy. The order of approximation can be raised by using a higher order polynomial reconstruction, by defining p^{j-1}_k as the unique quadratic polynomial which fits the averages $(v^{j-1}_{k-1}, v^{j-1}_k, v^{j-1}_{k+1})$, and by defining $\hat{v}^j_{2k}$ and $\hat{v}^j_{2k+1}$ as the averages of p^{j-1}_k on the corresponding intervals. The resulting multiscale transform is equivalent to the biorthogonal wavelets transform (BW), raising the approximation order in smooth regions and also generating oscillations near singularities.

ENO mechanism: we can improve the method by using a nonlinear prediction operator, such as *essentially non-oscillatory* (ENO) reconstruction, as proposed in [11]. The goal of such an operator is to improve the approximation order near the singularities. The ENO reconstruction chooses among the polynomials $(p_{k-1}^{j-1}, p_k^{j-1}, p_{k+1}^{j-1})$ the least oscillatory one with respect to some numerical criterion based only on the values at the coarser level.

Subcell resolution technique: we can go further with these ideas by using the subcell resolution technique in the cells containing singularities. These cells are detected as those for which the supports of the adjacent polynomials (selected by the ENO mechanism) do not intersect (see figure 5). For such cells,

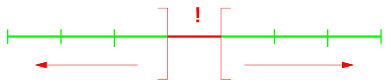

Figure 5: **Singular cell**

we then use a piecewise polynomial reconstruction $p_{k-2}^{j-1}\mathbf{1}_{]-\infty,y]} + p_{k+2}^{j-1}\mathbf{1}_{[y,\infty[}$, where y is obtained by consistency with the average v_k^{j-1}.

Example: in Figure 6 we show the reconstruction of a piecewise smooth function from a resolution level of 3. This illustrates the accuracy of our

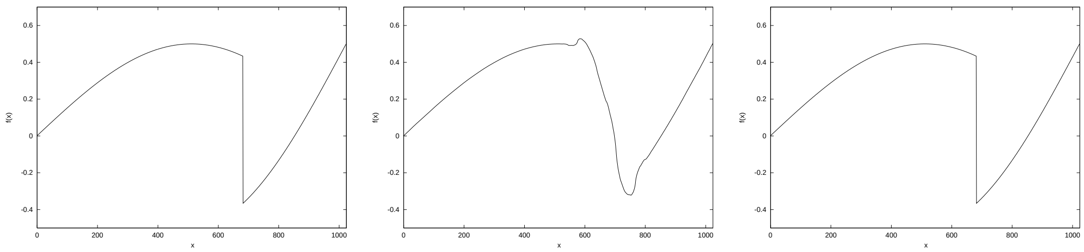

Figure 6: **(a) Original function, (b) BW and (c) ENO-SR**

nonlinear reconstruction method (ENO-SR). BW prediction is accurate for smooth functions and exact for poynomial functions, while ENO-SR prediction is accurate for piecewise smooth functions and exact for piecewise poynomial functions.

3.2.2 2-d formalism

In 2-d (in the case of digital images), we consider hierarchical discretizations by cell averages $v_k^j := 2^{2j} \int_{C_{j,k}} v(x,y)dxdy$, where $C_{j,k} := [k_x 2^{-j}, (k_x+1)2^{-j}] \times [k_y 2^{-j}, (k_y+1)2^{-j}] = I_{j,k_x} \times I_{j,k_y}$, with $k = (k_x, k_y) \in \{0, \cdots, 2^j - 1\}^2$. The finest level J corresponds to the pixel discretization. This choice of discretization fixes the decimation operator as: $v_k^{j-1} := \frac{1}{4}(v^j_{2k_x,2k_y} + v^j_{2k_x,2k_y+1} + v^j_{2k_x+1,2k_y} + v^j_{2k_x+1,2k_y+1})$.

As in the 1-d case, the goal is now to define a good prediction operator which makes the approximation error and then the detail coefficents as small as possible. This will result in a sparse representation for a given class of functions which model the signal. The lazzy choice for a such an operator is $\hat{v}_n^j = v_k^{j-1}$ if $C_{j,n} \subset C_{j-1,k}$, which corresponds to the decomposition in the Haar system with a low order of accuracy. The order of approximation can be raised by using higher order polynomial reconstruction, by defining p_k^{j-1} as the unique biquadratic polynomial which fits the averages $v_{k+l}^{j-1}, l \in \{-1, 0, 1\}^2$, and by defining the averages on the finer level as the averages of p_k^{j-1} on the corresponding cells. The resulting multiscale transform is equivalent to the biorthogonal wavelets transform, raising the approximation order in smooth regions and also generating oscillations near singularities.

Edge Adapted technique: The goal of nonlinear prediction operator such as ENO (see [11, 1, 14]) is to improve the approximation order near singularities. The ENO reconstruction chooses between the polynomials $p_{k+l}^{j-1}, l \in \{1, 0, 1\}^2$, the least oscillatory one with respect to some numerical criterion based only on the values at the coarser level.

We can go further with these ideas by using the edge adapted technique in the cells containing singularities. The EA prediction operates in three steps.

- The first step detects the pixels which might contain an edge by applying the following mechanism : a cell is labeled as singular if a discontinuity is detected in the horizontal or vertical direction using the 1-d ENO criterion.

- In a second step, in order to eliminate some false alarms, we require that the surrounding configuration is compatible with the model of a straight edge. as illustrated on Figure 7.

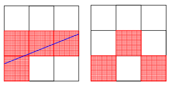

Figure 7: **Admissible and non-admissible configurations**

- In order to be accurate for piecewise smooth functions separated by smooth edges, the EA operator is exact for piecewise polynomial functions separated by straight edges (see [14]) as illustrated on Figure 8.

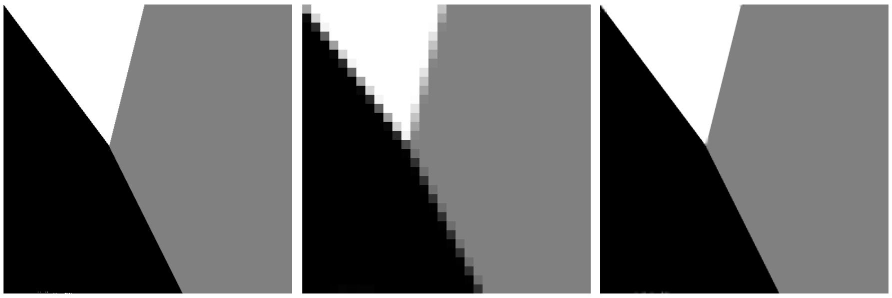

Figure 8: **Original image, coarse average and reconstruction**

That is why, in a third step, in the nonsingular cells, we use a biquadratic polynomial having a 3×3 support which avoids the singular cells. In the singular cells we first estimate the parameters of the edge and we then define a piecewise quadratic polynomial by using a pair of 3×3 configurations from both sides of the edge as illustrated on Figure 9.

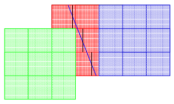

Figure 9: **Edge reconstruction**

For more details on EA prediction we refer the reader to [14] and [9] where this procedure has been introduced and where some properties have been investigated, namely the exactness of the prediction operator with respect to simplified model and also the approximation order.

4 A new compression algorithm

4.1 Presentation

The algorithm presented in Section 3 (EA) is an efficient compression algorithm for geometrical images. Its performance comes from the fact that it takes into account the geometry of the image. Such an algorithm is thus particularily well suited for the compression of the geometrical component of an image.

The idea of our algorithm (UVEA) is the following. We first decompose our original image f into two components u and v with the algorithm presented in Section 2 , u being the geometrical component, and v the oscillatory part. We then use the compression algorithm of Section 3 (EA) on u, and a classical compression algorithm on v. For v, we have decided to use the 7-9 biorthogonal wavelets (BW) (many numerical studies have shown that it gives the best distorsion rate performance for wavelet image transform codes [13]).

4.2 Numerical results and comments

Figure 1 is the Barbara image which we want to compress. When testing our algorithm, we have chosen to use images such as this one: images containing both geometrical and texture information. Figures 10 and 11 are the results we get with our algorithm. In the first case, have kept 10% of the coefficients of both the geometrical part and the textured part that we have obtained thanks to the algorithm of Section 2, and in the second case (Figure 11) we have kept 7% of the coefficients of the geometrical part and 13% for the textured part. We see that such a choice improve the results (see Table 1).

Figure 12 is the result we get with the EA algorithm. We have kept 20% of the coefficients. Since it has been developed for geometrical image, it performs not very well with images containing textures. Figure 13 is the result we get

Figure 10: Compressed image with our method (UVEA) (10% of the coefficients for both the u and the v component)

Figure 11: Compressed image with our method (UVEA) (7% of the coefficients for the u component and 13% for the v component)

Figure 12: Compressed image with EA (20% of the coefficients)

Figure 13: Compressed image with biorthogonal wavelets (BW) (20% of the coefficients)

Table 1: L^2-errors (Barbara image)

Algorithm	UVEA (10% , 10%)	UVEA (7% , 13%)	EA	BW
L^2-error	16.57	14.97	19.94	15.75
PSNR	23.48	24.79	22.13	24.23

UVEA BW

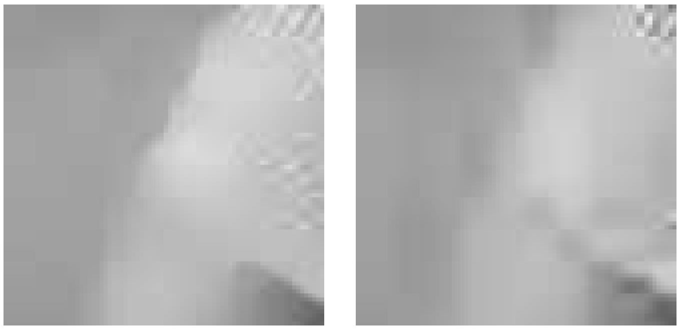

Figure 14: Zoom on Barbara's shoulder

with the standard BW algorithm (we have also kept 20% of the coefficients so that we can compare the results).

Table 1 shows the L^2-error. The best L^2-error is perfomed by our algorithm. Moreover, the geometrical information of the image is much more preserved with our algorithm than with BW (in both cases of Figures 10 and 11): this can be checked on the leg of the table for instance (see also Figure 14 which is a zoom of Barbara's shoulder). And some textures which disappear with the BW algorithm are preserved with our algorithm (see Figure 15 which is a zoom of Barbara's knee). Therefore, our algorithm gives a better result than the BW algorithm.

We have conducted experiments on several images, and this has confirmed these results. For instance, Figure 16 shows the result we get with our algorithm with the phare image of Figure 2. We have kept 10% of the coefficients

UVEA BW

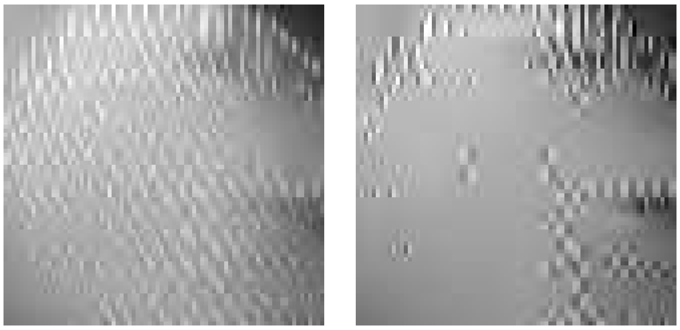

Figure 15: Zoom on Barbara's knee

Table 2: L^2-errors (phare image)

Algorithm	UVEA	BW
L^2-error	9.14	9.96
PSNR	28.88	28.19

of both the geometrical part and the textured part. And Figure 17 is the result we get with the BW algorithm (with 20% of the coefficients). The same conclusions hold for the phare image as for the Barbara image (see Table 2).

Strategy: We have numerically studied the best strategy to compress a given image. Depending on the image, it may be efficient to use more coefficients to code the geometric part than for the textured part (or the contrary). Figure 18 and 19 both illustrate it. Figure 18 (resp. Figure 19) deals with Figure 1 (resp. Figure 2). It shows how the log of the L^2 errors evolve with respect to the percentage p of coefficients kept for the structure part (p goes from 1 to 19). The percentage of coefficients kept for the texture part is $20\% - p$, i.e. we have kept 20% of the coefficients for the whole image. One easily sees that the best choice for the Barbara image is $p = 6$, and for the Phare image $p = 10$.

Figure 16: Compressed image with our algorithm (10% of the coefficients for both the u and the v component)

Figure 17: Compressed image with BW (20% of the coefficients)

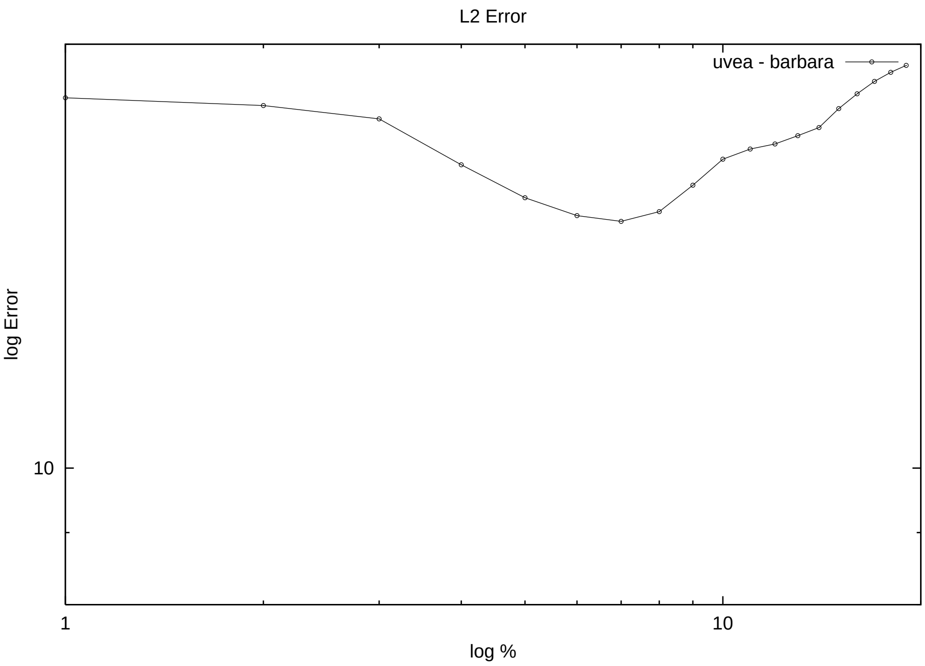

Figure 18: Best strategy for the Barbara image (20% of the coefficients)

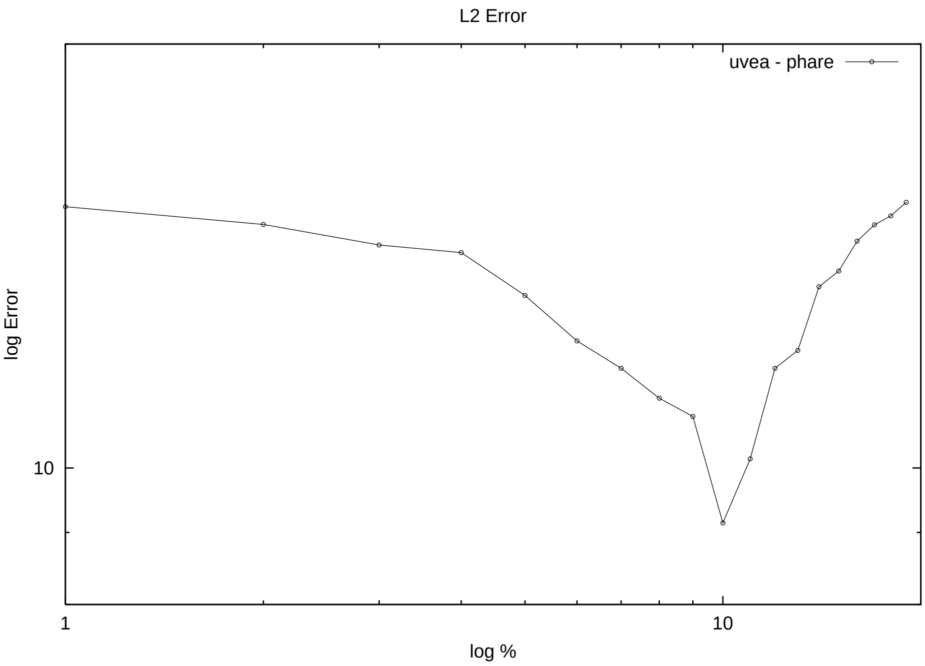

Figure 19: Best strategy for the Phare image (20% of the coefficients)

5 Conclusion and future prospects

We have presented a new compression algorithm for images containing both structure and texture information. We first split the image to compress into two components, one containing the structures, and another one the textures. We then carry out the compression of these two components with adapted algorithms: the EA algorithm of [14] for the geometrical part, anf the 7-9 biorthogonal wavelets for the texture part.

In our numerical tests, it outperforms the standard biorthogonal wavelets (BW) compression algorithm with respect to both the L^2-norm and the visual aspect of the image: the edges are sharper, and more textures are preserved. This is therefore a very promising way to improve compression rates in image processing.

We now plan to modify the decomposition algorithm. Indeed, as shown by the numerical study of the strategy, it has not been specifically developed to carry out compression. We should add another term in functional (5) so that the best strategy to compress an image would be to take the same number of coefficients for both the structure and the texture component. We also plan to improve the prediction operator of the EA algorithm, by taking into account that it only deals here with pure geometrical images. And we intend to compare the EA algorithm with other compression algorithms for geometrical images [6, 10, 12].

References

[1] S. Amat, F. Arandiga, A. Cohen, R. Donat, G. Garcia, and M. Von Oehsen. Data compression with eno schemes: a case study, November 2002. Appl.Comput.Harmon.Anal,11,273-288.

[2] F. Arandiga, A. Cohen, M. Doblas, R. Donat, and B. Matei. Sparse representations of images by edge adapted nonlinear multiscale transforms, September 2003. IEEE ICIP Conference Barcelona.

[3] J.F. Aujol, G. Aubert, L. Blanc-Féraud, and A. Chambolle. Decomposing an image: Application to SAR images. In *Scale-Space '03*, volume 1682 of *Lecture Notes in Computer Science*, 2003.

[4] J.F. Aujol, G. Aubert, L. Blanc-Féraud, and A. Chambolle. Decomposing an image: Application to textured images and SAR images, 2003. INRIA Research Report 4704, to appear in JMIV.

[5] M. Bertalmio, L. Vese, G. Sapiro, and S. Osher. Simultaneous structure and texture image inpainting. *IEEE Transactions on Image Processing*, 12(8):882–889, 2003.

[6] E. Candès and D. Donoho. A surprisingly effective nonadaptive representation for objects with edges, November 1999. Curves and Surfaces, L. L. Schumaker et al. (eds), Vanderbilt University Press, Nashville, TN.

[7] A. Chambolle. An algorithm for total variation minimization and applications, 2003. To appear in JMIV.

[8] A. Cohen and B. Matei. Compact representations of images by edge adapted multiscale transforms, September 2001. IEEE ICIP Conference Thessalonikki.

[9] A. Cohen and B. Matei. Nonlinear subdivisions schemes: applications to image processing, July 2002. Tutorial on multiresolution in geometric modelling,A.Iske, E.Quack and M.Floater eds., Springer.

[10] M. Do and M. Vetterli. The finite ridgelet transform for image representation., November 2001. Technical Report DSC/2001/019, Communication Systems Department, EPFL (Switzerland).

[11] A. Harten. Discrete multiresolution analysis and generalized wavelets, December 1993. Journal of Applied Numerical Mathematics,12:153-193.

[12] E. Lepennec. Bandelettes et représentations géométriques des images, December 2002. Ph. D. Thesis, Ecole Polytecnique, Paris.

[13] S.G. Mallat. *A Wavelet Tour of Signal Processing*. Academic Press, 1998.

[14] B. Matei. Methodes multi-résolution non-linéaires. applications au traitement d'image, November 2002. Ph. D. Thesis, Université Pierre et Marie Curie, Paris.

[15] Yves Meyer. Oscillating patterns in image processing and in some nonlinear evolution equations, March 2001. The Fifteenth Dean Jacquelines B. Lewis Memorial Lectures.

[16] S.J. Osher, A. Sole, and L.A. Vese. Image decomposition and restoration using total variation minimization and the H^{-1} norm. *Multiscale Modeling and Simulation: A SIAM Interdisciplinary Journal*, 1(3):349–370, 2003.

[17] L. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60:259–268, 1992.

[18] L.A. Vese and S.J. Osher. Modeling textures with total variation minimization and oscillating patterns in image processing. *Journal of Scientific Computing*, 19:553–572, 2003.

Unité de recherche INRIA Sophia Antipolis
2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

Unité de recherche INRIA Futurs : Parc Club Orsay Université - ZAC des Vignes
4, rue Jacques Monod - 91893 ORSAY Cedex (France)
Unité de recherche INRIA Lorraine : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex (France)
Unité de recherche INRIA Rennes : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)
Unité de recherche INRIA Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier (France)
Unité de recherche INRIA Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex (France)

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399