Available online at www.sciencedirect.com
ScienceDirect
Appl. Comput. Harmon. Anal. 23 (2007) 181–197
Applied and Computational Harmonic Analysis
www.elsevier.com/locate/acha

Image compression based on a multi-directional map-dependent algorithm ☆

F. Arandiga [a], J. Baccou [b], M. Doblas [a], J. Liandrat [b,*]

[a] *Universitat de Valencia, Campus de Burjassot, 46100, Burjassot, Spain*
[b] *LATP and Centrale Marseille, Technopôle de Château-Gombert, 13451, Marseille Cedex 20, France*

Received 13 April 2006; revised 21 December 2006; accepted 28 December 2006

Available online 13 January 2007

Communicated by Albert Cohen

Abstract

This work is devoted to the construction of a new multi-directional edge-adapted compression algorithm for images. It is based on a multi-scale transform that is performed in two steps: a detection step producing a map of edges and a prediction/multi-resolution step which takes into account the information given by the map. A short analysis of the multi-scale transform is performed and an estimate of the error associated to the largest coefficients for a piecewise regular function with Lipschitz edges is provided. Comparisons between this map-dependent algorithm and different classical algorithms are given.
© 2007 Elsevier Inc. All rights reserved.

1. Introduction

Compression of images involves sparse representations associated to efficient encoding. In such a framework, classical multi-scale wavelet-based methods turn out to be efficient tools [14]. Indeed, on one hand, they provide fast multi-scale algorithms for the decomposition/reconstruction of the image as well as sparse representations when a threshold is applied to the decomposition. On the other hand, they allow efficient encoding thanks to the tree structure of the dominant wavelet coefficients.

However, these classical approaches do not lead to fully satisfactory results for two main reasons: first, they use a linear multi-scale process and therefore do not adapt themselves to the local properties of the image (edges position) and second, the 2D-tensorial wavelet transform, generally used, is not optimal for the treatment of non-vertical and non-horizontal edges; as a result, the compression rate after a non-linear thresholding of the wavelet coefficients does not behave optimally (see, for instance, [14]).

In order to circumvent these drawbacks, recent works have produced multi-scale techniques incorporating the geometry of the image such as bandelets [15], contourlet [9] or ENO-based algorithms [1,7]. The multi-scale algorithm

☆ Research partially supported by European network "Breaking Complexity" #HPRN-CT-2002-00286 and by MTM 2005-07214.
* Corresponding author.
E-mail addresses: arandiga@uv.es (F. Arandiga), jean.baccou@irsn.fr (J. Baccou), doblas@uv.es (M. Doblas), jliandrat@ec-marseille.fr (J. Liandrat).

1063-5203/$ – see front matter © 2007 Elsevier Inc. All rights reserved.
doi:10.1016/j.acha.2006.12.002

that we are presenting in this paper has also been constructed in the same philosophy. It is edge-adapted but remains linear, at least partially. It is constructed of two main ingredients that are:

(1) an edge detection procedure producing a map of segmentation points,
(2) a linear multi-directional map-dependent multi-scale algorithm based on the bi-variate Harten's multi-resolution framework.

This paper is organized as follows: the 1D, segmentation-point-dependent, Harten's interpolatory multi-resolution framework is first briefly described in Section 2 and a new multi-directional map-dependent transform is then presented. A short numerical analysis focused on the error estimate associated to the M biggest coefficients for a piecewise regular function with Lipschitz edges is provided. In Section 3, a map construction technique is proposed, based on a simple algorithm for edge detection. The encoding technique is presented in Section 4. Section 5 is devoted to numerical comparisons between our algorithm, the ENO scheme and a translation-invariant one (i.e. non-adapted to the image edges) in terms of compression rate and of quality of the reconstructed images. Finally we close on Section 6 by a summary and some perspectives.

2. Segmentation-point-dependent Harten's interpolatory framework for multi-resolution

In all the paper, $j \in \mathbb{Z}$ stands for the scale parameter and $k \in \mathbb{Z}$ is the position parameter. For each value of j, associated to the scale 2^{-j}, a uniform grid $X^j = \{x_k^j,\ k \in \mathbb{Z}\}$ with $x_k^j = k2^{-j}$ is defined.

2.1. General univariate setting

Following the general approach described in [2,12] and references therein, the Harten's framework can be defined as a family of triplet $(V_j, D_j^{j-1}, P_{j-1}^j)$, $j \in \mathbb{Z}$. V_j is a separable space of approximation connected to a resolution level j and can be interpreted in the signal processing framework as the set of discrete values, $f^j = \{f_k^j\}_{k\in\mathbb{Z}}$, at the resolution level 2^{-j}, associated to a signal, f, defined on the whole line. D_j^{j-1} and P_{j-1}^j are two operators acting on the denumerable space V_j connecting adjacent resolution levels, $J_0 \leqslant \cdots \leqslant j \leqslant j+1 \leqslant \cdots \leqslant J_{\max}$. The decimation, D_j^{j-1}, is a linear operator that goes from high (V_j) to low (V_{j-1}) resolution spaces while the prediction, P_{j-1}^j, goes from low to high resolution spaces.

The operator P_{j-1}^j defines a subdivision scheme [10] acting between the level $j-1$ and the level j. More precisely, if we write

$$\forall \{f_k^{j-1}\}_{k\in\mathbb{Z}} \in V_{j-1}, \quad \left(P_{j-1}^j f^{j-1}\right)_k = \sum_{m\in\mathbb{Z}} a_{k-2m}^{j,k} f_m^{j-1}, \tag{1}$$

then, the coefficients $\{a_m^{j,k}\}_{m\in\mathbb{Z}}$, $(j,k) \in \mathbb{Z}^2$, can be interpreted as the masks of a subdivision scheme. When $\{a_m^{j,k}\}_{m\in\mathbb{Z}}$ are independent of f^{j-1}, the subdivision scheme is said to be linear and moreover stationary (respectively uniform) if these coefficients do not depend on j (respectively on k).

In general, in the Harten's framework, P_{j-1}^j is not required to be a linear operator and therefore, the corresponding subdivision scheme is non-linear.

For any signal, the discretization in V_j, f^j, can be substituted by the discretization in V_{j-1}, f^{j-1}, plus the detail coefficients $d^{j-1} = \{d_k^{j-1}\}_{k\in\mathbb{Z}}$ defined as follows: $\forall k \in \mathbb{Z}$, $d_k^{j-1} = f_k^j - (P_{j-1}^j f^{j-1})_k$.

Iterating this process, two multi-scale algorithms can be defined, associated to the levels $J_0 \leqslant j$: the decomposition algorithm, $(f^j \mapsto \{f^{J_0}, d^{J_0}, \ldots, d^{j-1}\})$, and the reconstruction algorithm, $(\{f^{J_0}, d^{J_0}, \ldots, d^{j-1}\} \mapsto f^j)$.

In the interpolatory framework where $f_k^j = f(k2^{-j})$ and D_{j+1}^j is a sub-sampling operator, the operator P_j^{j+1} is constructed from interpolation. In this work, Lagrange interpolations of a fixed degree D are used. The predicted values $\{(P_j^{j+1} f^j)_k\}_{k\in\mathbb{Z}} = \{\hat{f}_k^{j+1}\}_{k\in\mathbb{Z}}$ are then computed using Lagrange interpolants defined through a stencil of points associated to each couple (j,k) as follows.

Algorithm 2.1.

For any $k \in \mathbb{Z}$

$$\hat{f}_{2k}^{j+1} = f_k^j,$$

$$\hat{f}_{2k-1}^{j+1} = \sum_{m=-l_{j+1,2k-1}}^{r_{j+1,2k-1}-1} L_m^{l_{j+1,2k-1},r_{j+1,2k-1}}(-1/2) f_{k+m}^j,$$

end,

with

$$L_m^{l_{j+1,2k-1},r_{j+1,2k-1}}(x) = \prod_{n=-l_{j+1,2k-1},\, n\neq m}^{r_{j+1,2k-1}-1} \frac{x-n}{m-n}. \tag{2}$$

The integers $l_{j+1,2k-1}$ (respectively $r_{j+1,2k-1}$) denote the number of left (respectively right) points in the interpolating stencil associated to the couple (j,k) and to the quantity $\hat{f}_{2k-1}^{j+1}$. They satisfy $l_{j+1,2k-1} + r_{j+1,2k-1} = D+1$.

From Algorithm 2.1 and with the notation of expression (1), it is straightforward to define the mask of the associated subdivision scheme

$$a^{j,2k}\colon \begin{cases} a_0^{j,2k} = 1, \\ a_m^{j,2k} = 0 \quad \text{for } m \neq 0, \end{cases} \tag{3}$$

and

$$a^{j,2k-1}\colon \begin{cases} a_{-2m-1}^{j,2k-1} = L_m^{l_{j,2k-1},r_{j,2k-1}}(-1/2) & \text{for } m = -l_{j,2k-1}, \ldots, r_{j,2k-1}-1, \\ a_{2m}^{j,2k-1} = 0 & \text{otherwise.} \end{cases} \tag{4}$$

Following [4], we consider, in the sequel, three natural strategies to define, for any couple (j,k), the values $(l_{j+1,2k-1}, r_{j+1,2k-1})$: the first one is $l_{j+1,2k-1} = l$ and $r_{j+1,2k-1} = r = D-l+1$, where r and l are some reference values. It leads to the classical translation-invariant prediction. The corresponding subdivision scheme is stationary and uniform (and thus linear). When $l = r$, a direct correspondence between the interpolatory Harten's multi-scale framework (especially the prediction operator) and the interpolating subdivision scheme of Deslauriers and Dubuc (see [8]) can be exhibited.

The second one consists in taking $(l_{j+1,2k-1}, r_{j+1,2k-1})$ depending on a family of segmentation points (called a map) a priori defined on the line. This leads to what will be called a map-dependent prediction and defines a linear but non-stationary and non-uniform subdivision scheme.

The third strategy is $(l_{j+1,2k-1}, r_{j+1,2k-1})$ remaining functions of f^j according to local criteria. It leads to a data-dependent prediction and to non-linear subdivision schemes. A classical interpolating scheme belonging to this third class of strategy is the ENO scheme [2,11]. In this case, the stencil selection is based on the local regularity of the data estimated using divided differences.

In what follows, we will concentrate on linear strategies (strategies 1 and 2) and mainly on the map-dependent one. We delay to Remark 2.1 some comments about the analysis of data-dependent schemes.

To continue our investigations, we need to specify the map dependency of the stencil that we are interested in. It is motivated by the approximation of piecewise smooth functions. We then propose the following definition of the stencil, associated to a family of separated[1] segmentation points $\mathcal{S}_s$:

Definition 2.1 *(Map dependency).* Given a family of segmentation points $\mathcal{S}_s$, a map-dependent stencil is controlled by

- a triplet (D,l,r), $l \geqslant 0$, $r \geqslant 1$ and $l+r = D+1$ such that $(l_{j,2k-1}, r_{j,2k-1}) = (l,r)$ when $[x_{k-l}^{j-1}, x_{k+r-1}^{j-1}] \cap \mathcal{S}_s = \emptyset$,
- a rule for the selection of $(l_{j,2k-1}, r_{j,2k-1})$ when $[x_{k-l}^{j-1}, x_{k+r-1}^{j-1}] \cap \mathcal{S}_s \neq \emptyset$.

[1] The family of points $\mathcal{S}_s$ is said to be separated if and only if $\exists \varepsilon_{\mathcal{S}_s} > 0$ such that $\forall x, y \in \mathcal{S}_s, x \neq y \Rightarrow |x-y| \geqslant \varepsilon_{\mathcal{S}_s}$.

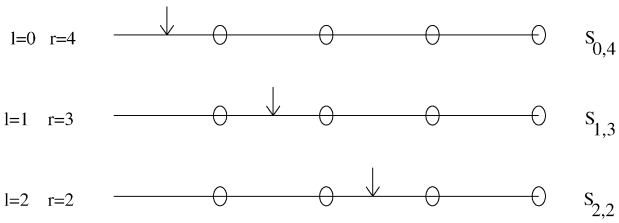


Fig. 1. Different stencils for 4-point interpolation associated to Definition 2.2. $S_{l,r}$ stands for the stencil with l left points and r right points.

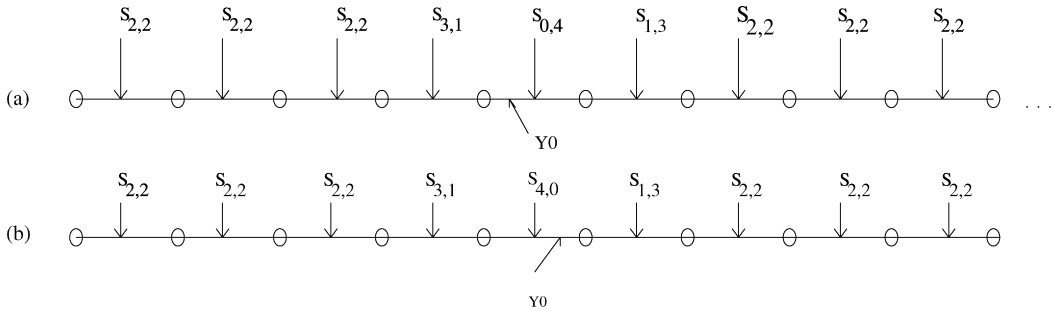


Fig. 2. Example of stencil selection following Definition 2.2. (a) $y_0 \in [x^j_{2k_{j-1}-2}, x^j_{2k_{j-1}-1}[$, (b) $y_0 \in [x^j_{2k_{j-1}-1}, x^j_{2k_{j-1}}[$.

Within this definition, the interpolating stencil is fixed by (l, r) at any point but in the vicinity of the segmentation points where the selection rule applies.
We continue this section by specifying the selection rule.

Assuming that the segmentation points are separated, there exists a scale from which each segmentation point can be considered as isolated with respect to the stencil length. Therefore we focus in the following definition on a single segmentation point y_0 corresponding to $\mathcal{S}_s = \{y_0\}$.

The rule for the stencil selection is constructed in order to choose stencils that avoid the segmentation point y_0.

More precisely, given (D, l, r), the selection rule is defined as follows.

Definition 2.2 *(Stencil selection).* For all $j \in \mathbb{Z}$, the index k_{j-1} is defined as the unique integer satisfying $y_0 \in [x^{j-1}_{k_{j-1}-1}, x^{j-1}_{k_{j-1}}[$. For all j and k such that $y_0 \in [x^{j-1}_{k-l}, x^{j-1}_{k+r-1}]$, we set

- If $y_0 \in [x^j_{2k_{j-1}-2}, x^j_{2k_{j-1}-1}[$, then

$$\begin{cases} \text{If } k < k_{j-1} & \text{then } r_{j,2k-1} = \inf\{k_{j-1} - k, r\} \text{ and } l_{j,2k-1} = D + 1 - k_{j-1} + k, \\ \text{If } k \geqslant k_{j-1} & \text{then } r_{j,2k-1} = D + 1 + k_{j-1} - k \text{ and } l_{j,2k-1} = \inf\{k - k_{j-1}, l\}. \end{cases} \tag{5}$$

- If $y_0 \in [x^j_{2k_{j-1}-1}, x^j_{2k_{j-1}}[$, then

$$\begin{cases} \text{If } k \leqslant k_{j-1} & \text{then } r_{j,2k-1} = \inf\{k_{j-1} - k, r\} \text{ and } l_{j,2k-1} = D + 1 - k_{j-1} + k, \\ \text{If } k > k_{j-1} & \text{then } r_{j,2k-1} = D + 1 + k_{j-1} - k \text{ and } l_{j,2k-1} = \inf\{k - k_{j-1}, l\}. \end{cases} \tag{6}$$

Figures 1 and 2, Table 1 display this selection rule when $D = 3$, $l = 2$ and $r = 2$.

The map-dependent multi-resolution algorithms associated to the previously defined prediction operator does not fit into the classical families for which general results for the numerical analysis of subdivision schemes are available [10]. However, once the segmentation points have been fixed, the prediction operator defined above is linear even though it is not translation-invariant. Therefore, the problem of convergence of the subdivision scheme and the question of the stability of the multi-resolution transform remain connected. In [4] it has been proved that the subdivision scheme previously defined is uniformly convergent in the space $\mathcal{C}(\mathbb{R})$ of functions f defined and continuous on $\mathbb{R} \setminus \mathcal{S}_f$, where $\mathcal{S}_f$ is a non-empty set of reals (depending of f) such that $\forall x \in \mathcal{S}_f$, $\lim_{y \to x, y > x} f(y) = f^+(x)$

Table 1
Lagrange coefficients associated to the stencil $S_{l,r}$. The stencils $S_{3,1}$ and $S_{4,0}$ are deduced by symmetry

$S_{0,4}$	$\frac{35}{16}$	$-\frac{35}{16}$	$\frac{21}{16}$	$-\frac{5}{16}$
$S_{1,3}$	$\frac{5}{16}$	$\frac{15}{16}$	$-\frac{5}{16}$	$\frac{1}{16}$
$S_{2,2}$	$-\frac{1}{16}$	$\frac{9}{16}$	$\frac{9}{16}$	$-\frac{1}{16}$

and $\lim_{y\to x, y<x} f(y) = f^-(x)$ exist in $\mathbb{R}$. More precisely, the convergence of the subdivision scheme is given by the following definition.

Definition 2.3 *(Convergence of the subdivision scheme).* The subdivision scheme associated to P_{j-1}^j is said to be L^∞-convergent if for any real sequence $\{f_k^0\}_{k\in\mathbb{Z}} \in V^0$, there exists a function $f \in \mathcal{C}(\mathbb{R})$ (called the limit function associated to f^0) such that: $\forall\varepsilon, \exists J$ such that $\forall j \geqslant J$ either

$$\left\| P_{j-1}^j \cdots P_1^2 P_0^1 f^0 - f^+\left(\frac{\cdot}{2^j}\right)\right\|_\infty \leqslant \varepsilon \tag{7}$$

or

$$\left\| P_{j-1}^j \cdots P_1^2 P_0^1 f^0 - f^-\left(\frac{\cdot}{2^j}\right)\right\|_\infty \leqslant \varepsilon, \tag{8}$$

where $\|\cdot\|_\infty$ stands for the uniform norm.

The convergence implies the existence of limit functions, being defined as the function towards which converges the subdivision process starting from a single non-zero value (equal to 1) at level 0. These functions are constructed from a finite set of compactly supported functions by translation. This is enough to get, following [10] for example, that the subdivision scheme is stable and, thanks to linearity, that the multi-resolution is stable. That means precisely that

Definition 2.4 *(Stability of the multi-resolution).* The *decomposition algorithm is stable* with respect to the norm $\|\cdot\|$ if: $\exists C$ such that $\forall j, \forall(f^j, f_\varepsilon^j)$, if $f^j \mapsto \{f^{J_0}, d^{J_0}, \ldots, d^{j-1}\}$ and $f_\varepsilon^j \mapsto \{f_\varepsilon^{J_0}, d_\varepsilon^{J_0}, \ldots, d_\varepsilon^{j-1}\}$, then

$$\left\| f_\varepsilon^{J_0} - f^{J_0}\right\| \leqslant C\left\| f_\varepsilon^j - f^j\right\|, \quad \text{and} \quad \forall m < j, \quad \left\| d_\varepsilon^m - d^m\right\| \leqslant C\left\| f_\varepsilon^j - f^j\right\|. \tag{9}$$

The *reconstruction algorithm is stable* with respect to the norm $\|\cdot\|$ if: $\exists C$ such that $\forall j > J_0$, $\forall\{f^{J_0}, d^{J_0}, \ldots, d^{j-1}\}$ and $\{f_\varepsilon^{J_0}, d_\varepsilon^{J_0}, \ldots, d_\varepsilon^{j-1}\}$, if $\{f^{J_0}, d^{J_0}, \ldots, d^{j-1}\} \mapsto f^j$ and $\{f_\varepsilon^{J_0}, d_\varepsilon^{J_0}, \ldots, d_\varepsilon^{j-1}\} \mapsto f_\varepsilon^j$, then

$$\left\| f_\varepsilon^j - f^j\right\| \leqslant C \sup\left(\left\| f_\varepsilon^{j-1} - f^{j-1}\right\|, \left\| d_\varepsilon^{j-1} - d^{j-1}\right\|\right). \tag{10}$$

The question of stability is of course crucial when one intends to use the multi-resolution transform in the context of signal compression. Indeed, the difference between the initial image and the image reconstructed after compression is controlled by stability constants.

Remark 2.1. As mentioned previously, the linearity of the map-dependent subdivision scheme plays here a key role to ensure its stability. This is not the case when considering ENO-type schemes [2,6,11]: for such schemes, the evaluation of divided differences which is coupled with the Lagrange interpolation to select the "best" stencil is performed at each position and at each resolution level on the reconstructed data: a perturbation of the data (such as these resulting from the thresholding of detail coefficients) can modify the prediction by selecting different stencils than those initially chosen in the non-perturbated situation and therefore leads to instabilities (see Fig. 10). The construction of a stable prediction requires the use of stabilization techniques such as the error control stabilization [1].

2.2. Generalization to a 2D map-dependent prediction

Considering the data as a two-dimensional array $\{f^j_{m,n}\}_{(m,n)\in\mathbb{Z}^2}$, a 2D bi-directional prediction is first constructed as a generalization of Algorithm 2.1. In this case, the segmentation points are replaced by segmentation curves that give rise by intersection with any vertical or horizontal line to segmentation points. It turns out to be that curvature of the segmentation curves may lead to non-separated points (see Fig. 4). In our framework, no adaptivity is possible in the vicinity of such points (see Section 2.2.1). However if the segmentation curves are assumed to be piecewise Lipschitz, the number of non-separated points remains bounded.

To begin with, we consider that the segmentation curves are given and defined by a family of curves, $\{C_i\}_{i\in I}$.

2.2.1. Bi-directional map-dependent prediction

Here, the 1D prediction Algorithm 2.1 is applied successively to the rows and the columns of the two-dimensional array. Generalizing the notations of Section 2.1 to the 2D case, we introduce the following 2D interpolating prediction procedure

Algorithm 2.2.

For any $(m,n)\in\mathbb{Z}^2$

$$\hat{f}^{j+1}_{2m,2n} = f^j_{m,n},$$

$$\hat{f}^{j+1}_{2m,2n-1} = \sum_{i=-l^H_{j+1,2m,2n-1}}^{r^H_{j+1,2m,2n-1}-1} L_i^{l^H_{j+1,2m,2n-1},r^H_{j+1,2m,2n-1}}(-1/2)\, f^j_{m,n+i},$$

$$\hat{f}^{j+1}_{2m-1,2n} = \sum_{i=-l^V_{j+1,2m-1,2n}}^{r^V_{j+1,2m-1,2n}-1} L_i^{l^V_{j+1,2m-1,2n},r^V_{j+1,2m-1,2n}}(-1/2)\, f^j_{m+i,n},$$

$$\hat{f}^{j+1}_{2m-1,2n-1} = \sum_{i=-l^V_{j+1,2m-1,2n-1}}^{r^V_{j+1,2m-1,2n-1}-1} L_i^{l^V_{j+1,2m-1,2n-1},r^V_{j+1,2m-1,2n-1}}(-1/2)\, \hat{f}^{j+1}_{2m+2i,2n-1},$$

end.

The superscript H (respectively V) refers to horizontal (respectively vertical) interpolating stencils. The number of left and right points in the stencils remains function of the family of segmentation curves following the 1D rule when the intersected points are separated. When this is not the case, the number of left (respectively right) points in the stencil take the reference values l (respectively r). Note that when this family is empty, the 2D bi-directional prediction is nothing but the classical tensor product one [1]. One can associate a map to the family of segmentation curves (see Section 3) and therefore this 2D multi-resolution is called map-dependent.

Note that the prediction directions, staying horizontal or vertical, do not take into account the orientation of the curves. Therefore, we introduce in the sequel another generalization of the 1D case incorporating a multi-directional map dependency.

2.2.2. Multi-directional map-dependent prediction

The multi-directional map-dependent prediction operator is constructed as a perturbation of the bi-directional one previously defined, incorporating the local orientation of the map [3]. We focus on four directions for the prediction (one horizontal, one vertical and two diagonal). Since the points of the following types, $\hat{f}^{j+1}_{2m-1,2n}$ and $\hat{f}^{j+1}_{2m,2n-1}$, cannot be easily computed by diagonal interpolations using the family $\{f^j_{m,n}\}_{(m,n)\in\mathbb{Z}^2}$, the difference with Algorithm 2.2 lies in the prediction of $\hat{f}^{j+1}_{2m-1,2n-1}$ that is performed using possibly diagonal interpolations according to the orientation of the map (Fig. 3).

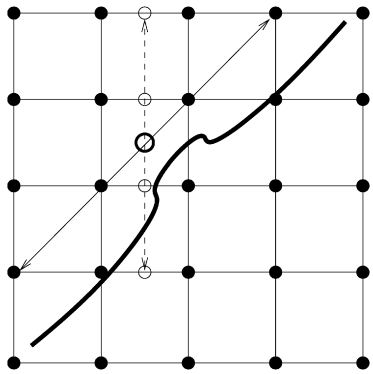


Fig. 3. Direction of prediction for a point of type $\hat{f}^{j+1}_{2m-1,2n-1}$. The dashed arrow stands for the direction when using a bi-directional map-dependent strategy while the continuous line arrow is used in the multi-directional prediction.

2.2.3. A short numerical analysis

The question of convergence of the underlying 2D map-dependent subdivision scheme is not addressed in this paper. In [4], the convergence has been established for a very close subdivision scheme.

We analyze here the efficiency of the associated multi-resolution transform. When it is not necessary to specify, we denote $\{d_\lambda\}_{\lambda\in\Lambda_j}$ the family of detail coefficients involved in the multi-resolution process at level j. The compression of the data uses detail ε-thresholding.

Classically, the efficiency of the multi-resolution process can be evaluated by studying the behavior of the L^2-norm of the reconstruction error versus the number of retained detail coefficients after ε-thresholding [14]. Let us first introduce the reconstruction error.

Definition 2.5 *(Reconstruction error).* For any $j \in \mathbb{Z}$, the reconstruction error at level j is the piecewise constant function e^j such that, $\forall(k_1,k_2)\in\mathbb{Z}\times\mathbb{Z}$,

$$\forall(x,y)\in[k_1,k_1+1[\times[k_2,k_2+1[,\quad e^j(x,y)=f^j_{k_1,k_2}-\left(P^j_{j-1}f^{j-1}\right)_{k_1,k_2}, \tag{11}$$

where P^j_{j-1} denotes the 2D map-dependent prediction operator described by Algorithm 2.2.

The L^2-norm of the reconstruction error is therefore, thanks to the definition of the details

$$\left\|e^j\right\|_{L^2}=2^{-j}\left(\sum_{\lambda\in\Lambda_j}|d_\lambda|^2\right)^{\frac{1}{2}}. \tag{12}$$

The relevant quantity to evaluate is the reconstruction error, $e_{[M]}$, when keeping the M biggest coefficients of the decomposition. According to (12), due to L^2-normalization, the quantity to reorder is the family of renormalized detail coefficients $\{\{2^{-j}d_\lambda\}_{\lambda\in\Lambda_j}\}_{j\in\mathbb{Z}}$ that we denote $\{\{\tilde{d}_\lambda\}_{\lambda\in\Lambda_j}\}_{j\in\mathbb{Z}}$ in the sequel. Introducing $\{\tilde{d}^o_{\lambda_\alpha}\}_{\alpha\in\mathbb{N}}$ the sequence of renormalized detail coefficients rearranged in a decreasing order, the L^2-norm of the reconstruction error $e_{[M]}$ can then be written as

$$\|e_{[M]}\|_{L^2}=\left(\sum_{\alpha>M}\left|\tilde{d}^o_{\lambda_\alpha}\right|^2\right)^{\frac{1}{2}}. \tag{13}$$

We want to find the largest s satisfying

$$\left(\sum_{\alpha>M}\left|\tilde{d}^o_{\lambda_\alpha}\right|^2\right)^{\frac{1}{2}}\leqslant CM^{-s}, \tag{14}$$

that provides the order of decay of the reconstruction error with respect to M, the number of retained coefficients.

Let us first recall the following proposition taken from [14].

Proposition 2.1. *If* $\sum_{\alpha=0}^{\infty} |\tilde{d}^o_{\lambda_\alpha}|^\gamma < \infty$ $(\gamma < 2)$, *then*

$$\sum_{\alpha > M} |\tilde{d}^o_{\lambda_\alpha}|^2 \leqslant C \left(\sum_{\alpha=0}^{\infty} |\tilde{d}^o_{\lambda_\alpha}|^\gamma \right)^{\frac{2}{\gamma}} M^{1-\frac{2}{\gamma}}. \tag{15}$$

It follows that the optimal order of decay of the detail coefficients is controlled by the smallest γ ensuring that $\sum_{\alpha=0}^{\infty} |\tilde{d}^o_{\lambda_\alpha}|^\gamma < \infty$. It turns out that this index is related to the Lipschitz regularity outside the curves $\{C_i\}_{i\in I}$ [2] of the function f representing the image. This result is provided by the following proposition.

Proposition 2.2. *If* f *is uniformly* $\mathcal{K}$*-Lipschitz over* $([a,b]\times[c,d])$ *outside the family of curves* $\{C_i\}_{i\in I}$, *then*

$$\sum_{\alpha=0}^{\infty} |\tilde{d}^o_{\lambda_\alpha}|^\gamma < \infty, \quad \text{if } \gamma > \frac{2}{\mathcal{K}+1}. \tag{16}$$

Note that Propositions 2.1 and 2.2 give

$$\|e_{[M]}\|_{L^2} \leqslant C M^{-\frac{\mathcal{K}}{2}}. \tag{17}$$

Proof. By construction, the detail coefficients are related to the prediction error. The accuracy of the prediction depends on whether or not the prediction is perturbated by a segmentation curve. We therefore introduce two types of detail coefficients: the first type (I) coefficients are associated to predicted points which associated stencil does not cross a segmentation curve while the remaining coefficients belong to the second type (II).

The proof is split into two parts. We first show (16) for type I detail coefficients, then for type II.

- Type I coefficients:

Since the interpolating procedure reproduces polynomials up to degree D and since f is uniformly $\mathcal{K}$-Lipschitz over $([a,b]\times[c,d])$ outside the family of curves $\{C_i\}_{i\in I}$, the amplitude of the detail coefficients at level j is controlled by $C2^{-\mathcal{K}j}$. Therefore, denoting $d^{(I),j}_{k_1,k_2}$, respectively $\tilde{d}^{o,(I),j}_{k_1,k_2}$, the corresponding detail, respectively renormalized and reordered, coefficients we have

$$\sum_{\alpha=0}^{\infty} |\tilde{d}^{o,(I)}_{\lambda_\alpha}|^\gamma \leqslant \sum_{j=J_0}^{\infty} \sum_{k_1,k_2=0}^{2^j} |2^{-j} d^{(I),j}_{k_1,k_2}|^\gamma \leqslant C \sum_{j=J_0}^{\infty} \sum_{k_1,k_2=0}^{2^j} 2^{-j\gamma(\mathcal{K}+1)} \leqslant C \sum_{j=J_0}^{\infty} 2^{-j((\mathcal{K}+1)\gamma-2)},$$

and the convergence of the sum is ensured provided $(\mathcal{K}+1)\gamma - 2 > 0$, i.e. $\gamma > \frac{2}{\mathcal{K}+1}$.

- Type II coefficients:

The detail coefficients of this type are associated to points located in the vicinity of non-separated segmentation points. As mentioned earlier, these points appear in the vicinity of a corner of a segmentation curve (see Fig. 4).

We first concentrate on vertical or horizontal predicted points. Using the notations of Fig. 4, a predicted point P at scale j has its associated stencil perturbated by the corner as soon as there are less than $D+1$ points at scale $j-1$ between $A(P)$ and $B(P)$ defined as the intersection of the horizontal line containing P with D^+ and D^-, or between $A'(P)$ and $B'(P)$, defined as the intersection of the vertical line containing P with D^+ and D^-.

In what follows, we show that, at each scale, the number of such points is independent of the scale j.

Let us introduce D^+: $y = a^+x + b^+$ and D^-: $y = a^-x + b^-$, the two half lines delimiting the corner (this is always true, up to the limit of large values of j). We assume, without loss of generality, that D^+ and D^- are in the same quarter plane with D^+ standing above D^- and we call $(x_0, y_0) = D^+ \cap D^-$.

[2] A function f is uniformly $\mathcal{K}$-Lipschitz over a domain A outside a family of curves $\{C_i\}_{i\in I}$ if there exists R such that for all $(\nu,t)\in A^2$ with $[\nu,t]\cap\{C_i\}_{i\in I} = \emptyset$, $|f(t) - P_{\langle\mathcal{K}\rangle}(t-\nu)| \leqslant R|t-\nu|^{\mathcal{K}}$, where $\langle\,\rangle$ stands for the integer part and $P_{\langle\mathcal{K}\rangle}$ is a polynomial of degree $\langle\mathcal{K}\rangle$.

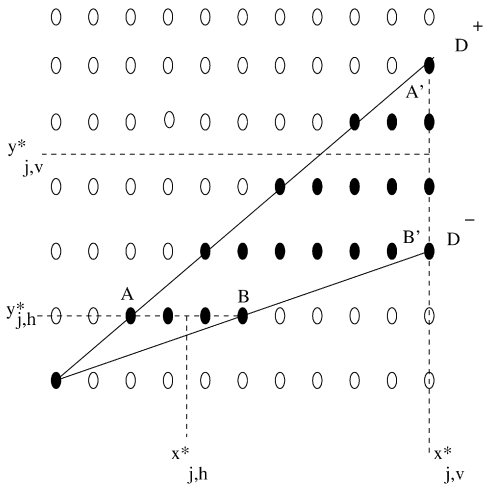


Fig. 4. Prediction near a corner, $D = 3$.

For any $j \in \mathbb{Z}$, we call $x_j^\star$ and $y_j^\star$ the quantities such that all the points at level j with an associated stencil perturbated by the corner lie into the domain $([x_0, x_j^\star] \times [y_0, y_j^\star]) \cap \{(x, y)$ with $x \geqslant x_0$ and $a^- x + b^- \leqslant y \leqslant a^+ x + b^+\}$. We now evaluate $x_j^\star$ and $y_j^\star$.

The vertical prediction is perturbated by the corner as soon as $x_0 \leqslant x \leqslant x_{j,v}^\star$ with $a^+ x_{j,v}^\star + b^+ - (a^- x_{j,v}^\star + b^-) = (D+1)2^{-(j-1)}$, that gives

$$x_{j,v}^\star = \frac{D+1}{a^+ - a^-} 2^{-(j-1)} + x_0, \tag{18}$$

and also

$$y_{j,v}^\star = a^- \frac{D+1}{a^+ - a^-} 2^{-(j-1)} + \frac{(D+1)}{2} 2^{-(j-1)} + y_0. \tag{19}$$

Similarly, we get for the horizontal prediction

$$x_{j,h}^\star = a^- \frac{D+1}{a^+ - a^-} 2^{-(j-1)} + \frac{(D+1)}{2} 2^{-(j-1)} + x_0, \tag{20}$$

$$y_{j,h}^\star = a^+ a^- \frac{D+1}{a^+ - a^-} 2^{-(j-1)} + y_0. \tag{21}$$

Finally defining $x_j^\star = \max(x_{j,v}^\star, x_{j,h}^\star)$ and $y_j^\star = \max(y_{j,v}^\star, y_{j,h}^\star)$, it follows from (18)–(21), that at each level j, the number of points associated to vertical or horizontal prediction stencils perturbated by the corner does not depend on j.

A similar estimate stands for the points associated to diagonal prediction stencils. Therefore, the number of detail coefficients of type II, at each level j is bounded by a constant.

Since the prediction for type II coefficients uses a stencil crossing segmentation points, the decay rate of type II coefficients is slow. More precisely, since the segmentation points coincide with discontinuities, the amplitude of type II coefficients is controlled by $C2^{-j}$. Coming back to the sum, we therefore have

$$\sum_{\alpha=0}^{\infty} |\tilde{d}_{\lambda_\alpha}^{o,(\mathrm{II})}|^\gamma \leqslant \sum_{j=J_0}^{\infty} \sum_{k_1,k_2=0}^{2^j} |2^{-j} d_{k_1,k_2}^{(\mathrm{II}),j}|^\gamma \leqslant C \sum_{j=J_0}^{\infty} 2^{-2j\gamma},$$

and the convergence of this sum is always ensured, that concludes the proof. □

Remark 2.2. In the case of a classical tensor product approach, similar calculations lead to the following estimate:

$$\|e_{[M]}\|_{L^2} \leqslant C M^{-\frac{1}{2}}. \tag{22}$$

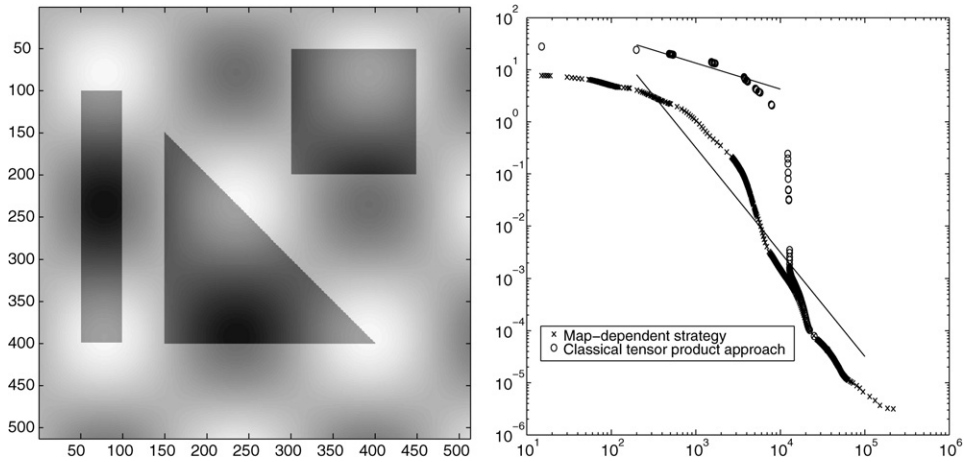


Fig. 5. Reconstruction error for a piecewise C^∞ function. (Left) Test function. (Right) $\|e_{[M]}\|_{L^2}$ versus M.

Without adaption to the edges, the decay of the reconstruction error L^2-norm is therefore much slower than in the case of a map-dependent approach.

We have plotted in Fig. 5 (right), the L^2-norm of the reconstruction error $e_{[M]}$ versus M, the number of non-zero normalized detail coefficients after thresholding. The multi-scale algorithms used for the decomposition/reconstruction of f are based either on the classical tensor product approach [1] or on the multi-directional map-dependent strategy (Algorithm 2.2). In both cases, the number of points in the interpolating stencil is set to 4.

The 2D-test function $f:[0,1]\times[0,1]\to\mathbb{R}$ is piecewise C^∞ (Fig. 5 (left)) and it is discretized on a 2D-dyadic grid of step size $\frac{1}{512}$.

When using the 2D multi-directional map-dependent algorithm, the L^2-norm of the reconstruction error is decreasing as M^{-2} (slope -2 in the log scale). This is in full agreement with (17) since for a piecewise C^∞ test function, the upper-script $\mathcal{K}$ in (17) is fixed by the degree of polynomial reproduction in the interpolating process (i.e. $\mathcal{K}=D+1=4$).

Concerning the tensor product approach, two different behaviors are noticeable:

- When $10^2\leqslant M\leqslant 8\times 10^3$, the L^2-norm is decreasing as $M^{-1/2}$ (slope $-\frac{1}{2}$ in the log scale), which is in full agreement with inequality (22).
- When $M\geqslant 10^4$, the decay is the same as in the case of a map-dependent strategy. Indeed, for such values of M, the thresholded coefficients correspond to points located in smooth regions of the image where, for both approaches the associated decay is M^{-2}.

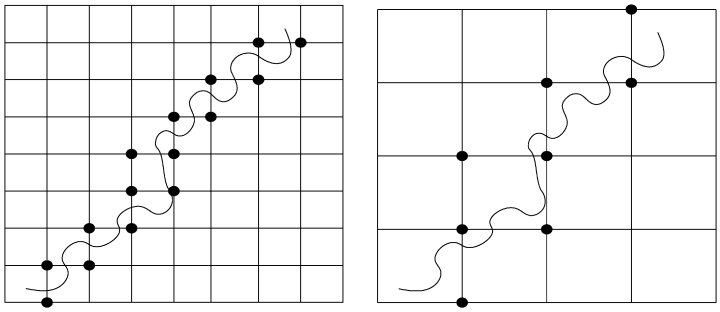
A key point for map-dependent transformations is the construction of a map that is related to the positions of the segmentation curves. When considering images, these segmentation curves should coincide with the edges of the image. Therefore, in the sequel, we describe a construction technique leading to a map of edges.

3. Construction of the map

In all that follows, we consider images of size $2^{J_{\max}}\times 2^{J_{\max}}$ and, therefore, $(m,n)\in\{0,\dots,2^{J_{\max}}-1\}^2$.

Ideally, assuming that the image edges are given and coincide with segmentation curves, $\{C_i\}_{i=1,\dots,K}$, the map of edges at level j (see Fig. 6a) is the binary matrix, $[\mathrm{map}^j(m,n)]_{0\leqslant m,n\leqslant 2^j-1}$ where the value 1 is given to the couples (m,n) such that the cross of length 2^{-j+1} centered at the dyadic point $(m2^{-j},n2^{-j})$ intersects the segmentation curves. More precisely $\mathrm{map}^j(m,n)=1$ if and only if $\exists 1\leqslant i\leqslant K$ such that $C_i\cap([((m-1)2^{-j},n2^{-j});((m+1)2^{-j},n2^{-j})]\cup[(m2^{-j},(n-1)2^{-j});(m2^{-j},(n+1)2^{-j})])\neq\emptyset$.

It is clear that a simple analysis of map^j provides the dyadic intervals intersected by the segmentation curves and can trigger the stencil selection.



(a) (b)

Fig. 6. Example of map, the curve stands for an edge while the black points are the pixels belonging to the map. (a) Map at level j, (b) map at level $j-1$.

Practically, the map of edges is first constructed at level $J_{\max}$ (Section 3.1). Then, the map at any level $J_0 \leqslant j \leqslant J_{\max}-1$ is deduced following Section 3.2.

3.1. Construction of $map^{J_{\max}}$

This construction is performed in three steps.

3.1.1. Step 1: edge detection

The detection is performed with the Sobel filter [16]. After a convolution of the original image with the Sobel directional mask, a sign checking and an α-thresholding (in our example, in Fig. 7, $\alpha = 0.5(\max - \min)$, where max, respectively min, denotes the maximum, respectively the minimum, of the gray levels of the original image) provide two outputs matrices: the first one displays a map of edges (called the step-1-map) while the second one is a matrix of orientations.

The number of points belonging to the step-1-map (see Fig. 7 (top, right)) must be reduced. It is done by the two following steps.

3.1.2. Step 2: reduction of the width of the detected zones

This step uses the matrix of orientations. If a vertical sequence of points of length $m_2 - m_1$, $\{X^{J_{\max}}_{m,n}\}_{m_1 \leqslant m \leqslant m_2}$, with horizontal and diagonal orientations is detected, the following differences are computed, $|f^{J_{\max}}_{m-1,n} - 2f^{J_{\max}}_{m,n} + f^{J_{\max}}_{m+1,n}|$, $m_1 \leqslant m \leqslant m_2$. If $m'_1 \leqslant m'_2$, correspond to the indexes of the two biggest differences, the sequence of detected pixels is reduced to $\{X^{J_{\max}}_{m,n}\}_{m'_1 \leqslant m \leqslant m'_2}$.

The same procedure is applied to horizontal sequences of detected points with vertical and diagonal orientations. That step produces the step-2-map (see Fig. 7 (bottom, left)).

3.1.3. Step 3: deletion of the false alarms related to noise and to texture

The remaining marked points are chained one to the other using North–East–South–West (NESW) chain codes [13]. A control of the length of the obtained chains is performed and, according to a threshold, L_{chain}, leads to a rejection from the map of the marked points belonging to chains of length smaller than L_{chain}. Figure 7 (bottom, right) displays the step-3-map when $L_{\text{chain}} = 20$.

3.2. Map of edges for each resolution level

In order to use the information contained in the map at each resolution level, the following procedure transforms the map at level $j+1$ to a new map at level j (see Fig. 6): for each even value of m, if $\text{map}^{j+1}(m,n) = 1$ and $\text{map}^{j+1}(m,n+1) = 1$ then:

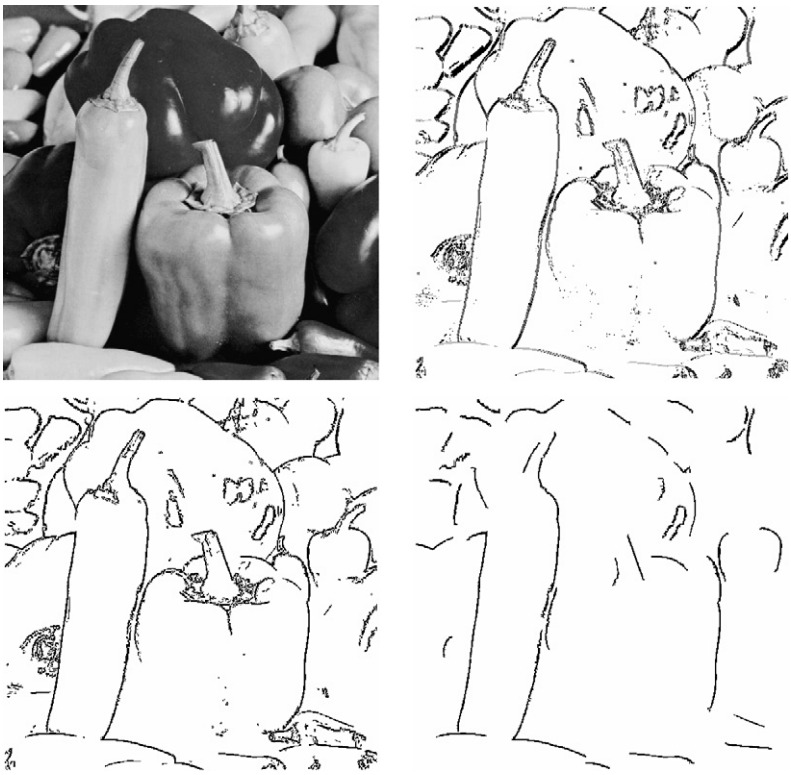


Fig. 7. From top left to bottom right: Original image. Step-1-map (number of non-zero coefficients = 25,263). Step-2-map (number of non-zero coefficients = 18,384). Step-3-map (number of non-zero-coefficients = 6824).

- If n is even, then $\mathrm{map}^j\left(\frac{m}{2}, \frac{n}{2}\right) = 1$ and $\mathrm{map}^j\left(\frac{m}{2}, \frac{n}{2} + 1\right) = 1$.
- If n is odd, then $\mathrm{map}^j\left(\frac{m}{2}, \frac{n+1}{2} - 1\right) = 1$ and $\mathrm{map}^j\left(\frac{m}{2}, \frac{n+1}{2}\right) = 1$.

The same procedure is defined for even value of n when $\mathrm{map}^{j+1}(m, n) = 1$ and $\mathrm{map}^{j+1}(m + 1, n) = 1$, switching m and n.

Remark 3.1. Note that the full procedure (the construction of the map and the map-dependent prediction introduced in Section 2.2.2) defines a data-dependent (i.e. non-linear) multi-resolution comparable to the classical ENO approach [2] or to the more recent ENO-EA algorithm [7]. However, the main differences between our approach and these two last techniques are the following:

- The map-dependent prediction takes into account the orientation of the detected edges. Therefore, the quality of the prediction is not weakened close to non-vertical or non-horizontal edges in opposition to what happens when bi-directional approaches (where prediction directions are vertical or horizontal) are used (see, for instance, the zooming in Fig. 9).
- Keeping apart the problem of map definition, the map-dependent multi-resolution procedure described in this paper is linear and therefore stable with respect to perturbations due to thresholding after decomposition. This is not the case for ENO-based approaches since the detection step and the prediction step are coupled at each resolution level (see Remark 2.1).

However, the price to pay in our approach is the storage of the map. This last point as well as the encoding of the compressed information are discussed in the next section.

4. Encoding of the compressed information

The efficiency of the encoding is related to the nature and structure of the data. The data to be compressed are here: the map, the low-resolution image and the ε-thresholded then rounded detail coefficients.

Using step 3 of the map construction (Section 3.1.3), the map of an image can be represented by two files. In the first one, we store the beginning of the chains and their length, while we use the second one to save the NESW chains which represent the direction from one pixel to the next.

Then, the low-resolution image, the ε-thresholded then rounded detail coefficients and the map are encoded using the *lossless* PPMZ algorithm [5], based on the (prediction by partial matching PPM) approach.

We delay to Remark 5.1, some comments concerning the choice of an optimal encoder for the detail coefficients.

5. Applications and numerical comparisons

We provide, here, several numerical tests to compare three compression algorithms:

- the map-dependent compression (MDC), constructed with a 1D 4-point stencil and the corresponding multi-directional map-dependent prediction (Section 2.2.2) associated to a Sobel-based map construction (Section 3),
- the 2D tensor product ENO scheme (ENO) based on the non-linear selection of 4-point stencils [1],
- the classical tensor product compression (TPC) constructed with a 1D translation-invariant symmetric 4-point stencil.

For both algorithms, the encoder is the PPMZ algorithm (Section 4).

The comparison is performed in terms of:

- compression rate: it is defined as the bits per pixel ratio $\text{bpp} = \frac{N}{512^2}$, where N stands for the number of bits required to store the compressed image,
- quality of the reconstructed image: it is quantified using the (peak signal-to-noise ratio PSNR) defined as, $\text{PSNR} = 10\log_{10}\left(\frac{256^2 2^{2J_{\max}}}{\| f^{J_{\max}} - \hat{f}^{J_{\max}} \|_{l^2}}\right)$, where $\hat{f}^{J_{\max}}$ is the image reconstructed from the compressed data. Note that the visual quality of the reconstructed image is also an important criterion.

5.1. The case of a geometric image

Table 2 and Fig. 8 (left) display the bpp ratio and the corresponding PSNR using the three compression algorithms in the case of a geometric image. It comes out that MDC is very efficient for such an image since, for the same PSNR, its bpp ratio is ten times smaller than the corresponding ratio reached with TPC. Comparing with ENO, instability effects associated to the specific non-linearity of ENO makes that the results obtained with MDC are much better. The high performance of MDC for a geometric image can also be evaluated in term of visual quality of the reconstructed image (Fig. 9); reconstruction failure occurs only close to the corners.

When the geometric image is perturbated by a noise, MDC outperforms again TPC and ENO (see Fig. 8 (center)). Moreover, the analysis of the quality of the reconstructed image after thresholding can be used to evaluate the ability of the algorithm to act as a edge-enhancer. The results of Fig. 10 and Table 3 show that, in that framework MDC is more efficient than the other algorithms.

5.2. The case of a real image

We provide, in Table 4 and Fig. 8 (right), the bpp ratio and the PSNR in the case of a real image.

It appears that the efficiency of the map-dependent approach is reduced since, for a same PSNR, the bpp ratio is only slightly smaller than the corresponding value associated to TPC.

However, as can be seen in Fig. 11, the MDC provides a better edge reconstruction (and therefore a better visual quality of the reconstructed image) even if the corresponding PSNR is smaller than the one obtained with TPC. The comparison with ENO is also in favor of MDC.

Table 2
Non-zeros and bytes of the map and details, bits per pixel (bpp) and PSNR. Case of a geometric image

	nnz		Bytes		bpp	PSNR
	Map	Details	Map	Details		
TPC	0	7191	0	7262	0.22	36.26
	0	5800	0	6162	0.19	34.41
	0	4971	0	5293	0.16	32.85
	0	3321	0	3542	0.11	29.22
ENO	0	6968	0	7163	0.22	32.22
	0	6201	0	6593	0.20	31.64
	0	4668	0	5251	0.16	30.03
	0	2286	0	2350	0.07	25.62
MDC	6871	198	436	539	0.030	53.72
	6871	143	436	429	0.026	49.89
	6871	139	436	390	0.025	49.84
	6871	93	436	300	0.022	43.38

Table 3
Non-zeros and bytes of the map and details, bits per pixel (bpp) and PSNR. Case of a geometric image with noise

	nnz		Bytes		bpp	PSNR
	Map	Details	Map	Details		
TPC	0	11,060	0	15,501	0.47	35.86
	0	6116	0	10,008	0.30	32.43
	0	3873	0	6203	0.19	29.54
	0	3234	0	5015	0.15	27.96
ENO	0	7611	0	13,963	0.43	33.27
	0	5687	0	10,511	0.32	31.55
	0	3477	0	6795	0.21	28.15
	0	2104	0	4206	0.13	25.92
MDC	6892	951	598	2209	0.085	36.21
	6892	268	598	866	0.045	35.16
	6892	189	598	635	0.038	34.65
	6892	161	598	549	0.035	34.14

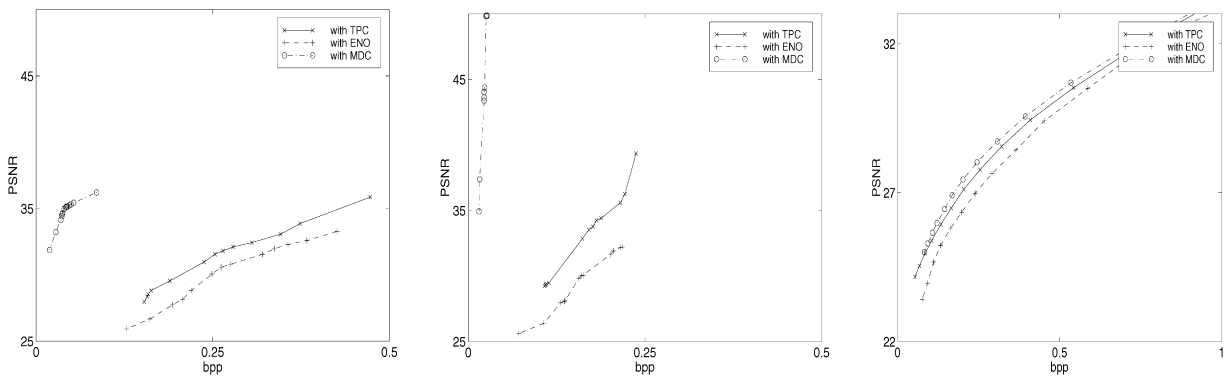


Fig. 8. PSNR versus bpp. (Left) Geometric image. (Center) Geometrical with noise. (Right) Real image.

Remark 5.1. The PPMZ encoder is not the best encoder for the transformed data since it does not take into account the tree structure of the details which is classically exploited by EZW [17] or EBCOT [18] for example. With these

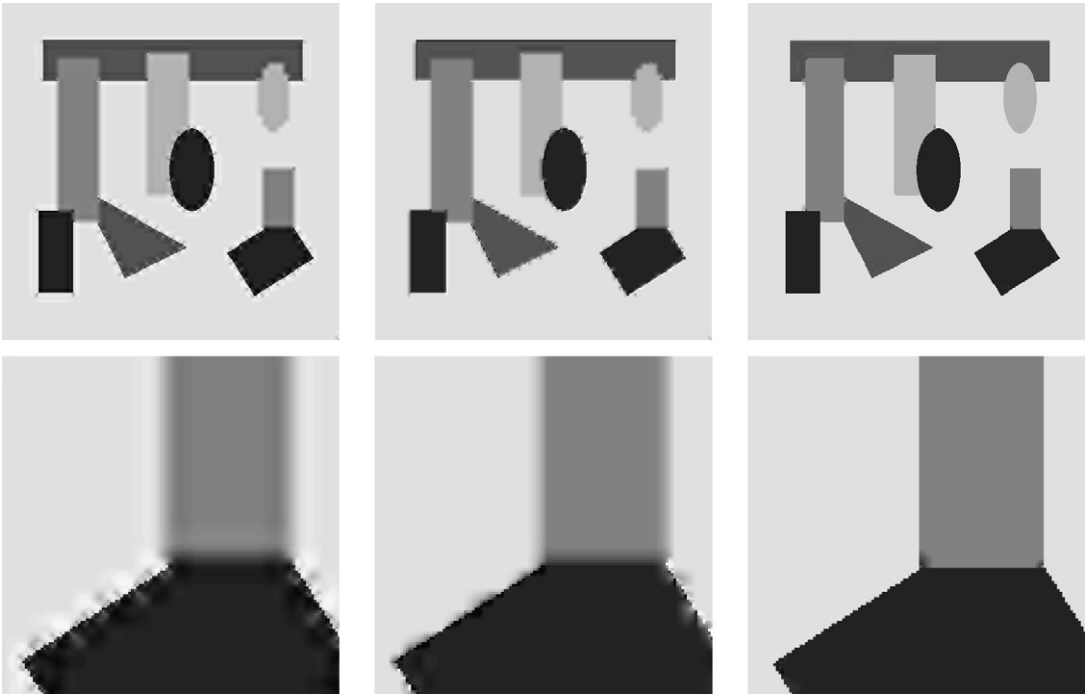


Fig. 9. Reconstruction of a geometric image, $J_0 = 6$, $J_{\max} = 9$. Top (from left to right): with TPC, bpp = 0.11 and PSNR = 29.22; with ENO, bpp = 0.07 and PSNR = 25.62; with MDC, bpp = 0.02 and PSNR = 43.38. Bottom: zooming.

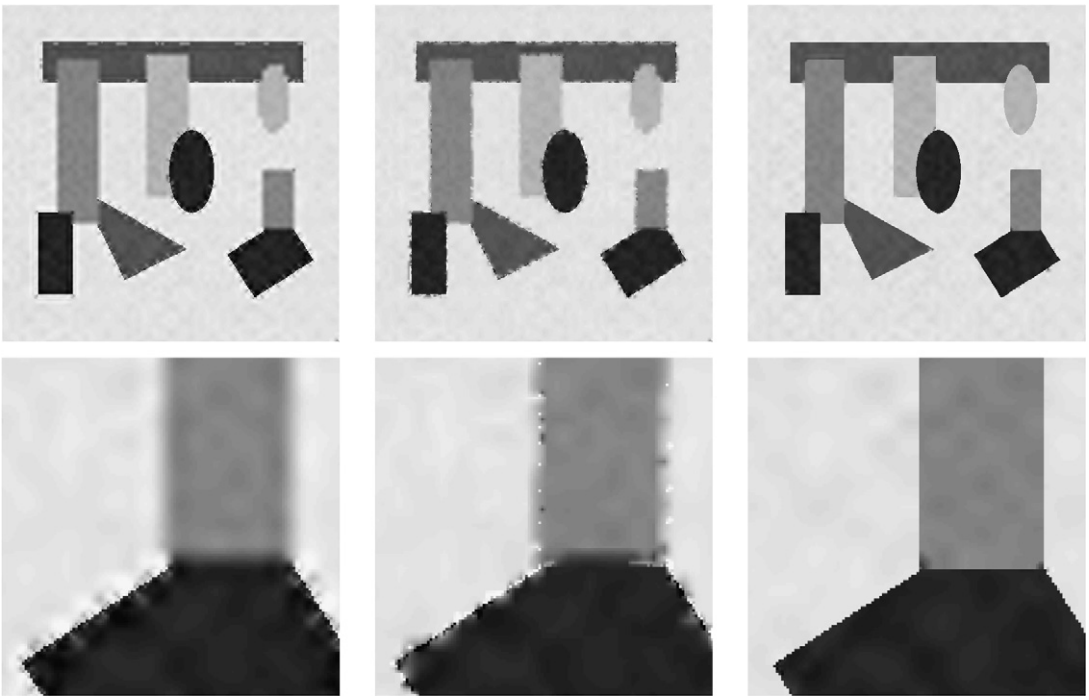


Fig. 10. Reconstruction of a geometric image with noise, $J_0 = 6$, $J_{\max} = 9$. Top (from left to right): with TPC, bpp = 0.15 and PSNR = 27.96; with ENO, bpp = 0.13 and PSNR = 25.92; with MDC, bpp = 0.03 and PSNR = 34.14. Bottom: zooming.

Table 4
Non-zeros and bytes of the map and details, bits per pixel (bpp) and PSNR. Case of a real image

	nnz		bytes		bpp	PSNR
	Map	Details	Map	Details		
TPC	0	5133	0	10,544	0.32	28.55
	0	3972	0	8384	0.25	27.77
	0	3099	0	6731	0.20	27.11
	0	2450	0	5456	0.17	26.48
	0	1948	0	4418	0.13	25.93
ENO	0	5742	0	11,970	0.36	28.43
	0	4463	0	9555	0.29	27.64
	0	3531	0	7840	0.24	26.97
	0	2450	0	6481	0.20	26.34
	0	2293	0	5337	0.16	25.79
MDC	6824	4149	1181	8941	0.31	28.72
	6824	3115	1181	6880	0.25	28.02
	6824	2386	1181	5467	0.20	27.44
	6824	1838	1181	4356	0.17	26.92
	6824	1461	1181	3603	0.15	26.45

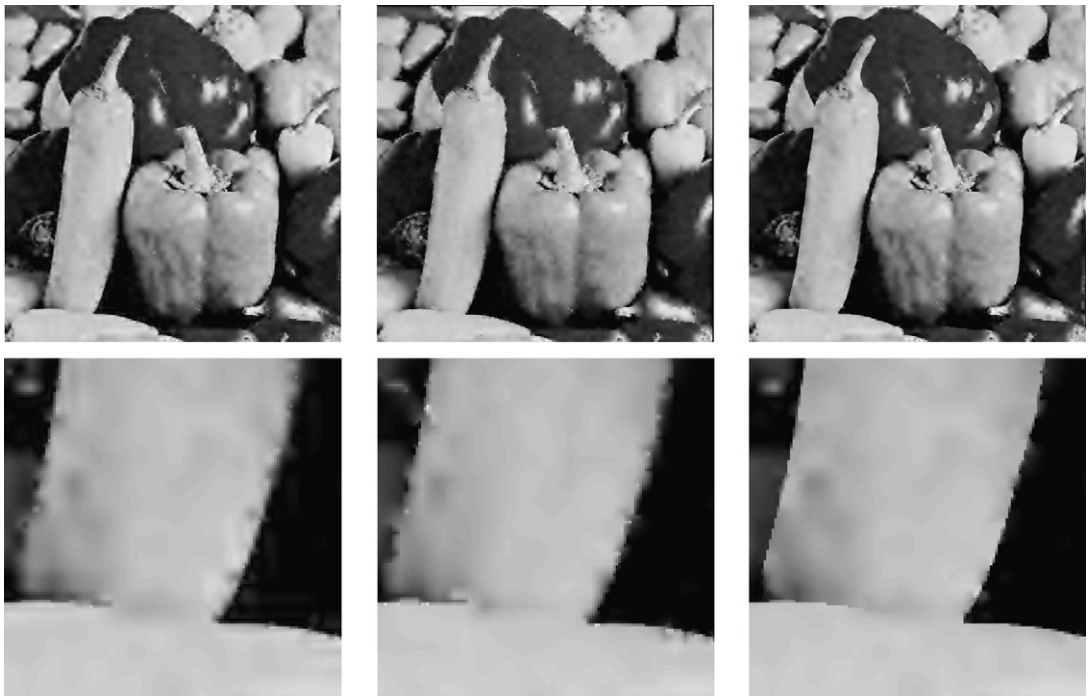


Fig. 11. Reconstruction of the peppers image, $J_0 = 6$, $J_{\max} = 9$. Top (from left to right): with TPC, bpp = 0.20 and PSNR = 27.11; with ENO, bpp = 0.16 and PSNR = 25.79; with MDC, bpp = 0.15 and PSNR = 26.45. Bottom: zooming.

encoders, we would obtain better rates of compression but we would reach the same conclusions with respect to the comparison between the MDC, TPC, and ENO algorithms.

6. Conclusion

In this paper, a new multi-scale algorithm has been introduced using the Harten's framework. Its construction couples an edge-detection step leading to a map with a multi-directional map-dependent prediction. The advantages of this algorithm are the following:

- the multi-scale transform incorporates the information about the geometry of the image,
- keeping apart the detection procedure, the map-dependent multi-scale process is fully linear. Therefore, the convergence of the associated subdivision scheme leads to stability with respect to perturbations,
- the map and the multi-scale decomposition coefficients can be efficiently stored.

The obtained results are of two kinds: on one hand, the map-dependent compression algorithm is shown to be theoretically optimal to compress piecewise regular images. On the other hand, the applications show that the map-dependent compression algorithm outperforms standards algorithms such as ENO and the classical translation-invariant tensor product one even in the case of a real image.

In addition to efficiency improvement connected to the use of zero-tree or block-type encoders, significant gains are expected in connection to the optimization of the detection techniques in the first step of the map-construction algorithm and to the incorporation of a stronger map dependency in the prediction procedure.

References

[1] S. Amat, F. Arandiga, A. Cohen, R. Donat, Tensor product multiresolution analysis with error control for compact image representation, Signal Process. 4 (2002) 587–608.
[2] F. Arandiga, R. Donat, Nonlinear multi-scale decompositions: The approach of A. Harten, Numer. Algorithms 23 (2000) 175–216.
[3] J. Baccou, Analyses multirésolutions et problèmes de bords : Applications au traitement d'images et à la résolution numérique d'équations aux dérivées partielles, Ph.D. thesis, The University of Provence, 2004.
[4] J. Baccou, J. Liandrat, Position dependent Lagrange interpolating multi-resolutions, Int. J. Wavelets Multiresolut. Inf. Proc., in press.
[5] C. Bloom, Solving the problems of context modeling, http://www.cbloom.com/papers/, 1998.
[6] A. Cohen, N. Dyn, B. Matei, Quasilinear subdivision schemes with applications to ENO interpolation, Appl. Comput. Harmon. Anal. 15 (2003) 89–116.
[7] A. Cohen, B. Matei, Compact representations of images by edge adapted multiscale transforms, in: IEEE-ICIP Conf., Thessaloniki, 2001.
[8] G. Deslauriers, S. Dubuc, Interpolation dyadique, in: Fractals, dimensions non entières et applications, Masson, Paris, 1987, pp. 44–45.
[9] M.N. Do, M. Vetterli, The contourlet transform: An efficient directional multiresolution image representation, IEEE Trans. Image Process. 14 (2005) 2091–2106.
[10] N. Dyn, Subdivision schemes in computer-aided geometric design, in: W.A. Light (Ed.), Wavelets, Subdivision Algorithms and Radial Basis Functions, Advances in Numerical analysis, vol. II, Clarendon Press, Oxford, 1992, pp. 36–104.
[11] A. Harten, Discrete multiresolution analysis and generalized wavelets, J. Appl. Numer. Math. 12 (1993) 153–192.
[12] A. Harten, Multiresolution representation of data: A general framework, SIAM J. Numer. Anal. 33 (3) (1996) 1205–1256.
[13] R. Estes Jr., V. Ralph Agazi, Efficient error free chain coding of binary documents, in: Proc. Data Compression Conference, Snowbird, Utah, IEE Computer Society Press, Los Alamitos, CA, 1995, pp. 122–132.
[14] S. Mallat, A Wavelet Tour of Signal Processing, Academic Press, San Diego, 1998.
[15] E. Le Pennec, S. Mallat, Sparse geometric image representations with bandelets, IEEE Trans. Image Process. 14 (4) (2005) 423–438.
[16] A. Peyronne, Utilisation de methodes adaptatives pour l'agrandissement d'images, GrAN Reports, University of Valencia, 2001.
[17] J. Shapiro, Embedded image coding using zerotrees of wavelet coefficients, IEEE Signal Process. 12 (1993) 3445–3462.
[18] D. Taubman, Embedded Block Coding with Optimized Truncation, Signal Processing: Image Communication, vol. 17, Elsevier, 1998.