Nonlinear and Nonseparable Bidimensional Multiscale Representation Based on Cell-Average Representation

Basarab Mateï,
LIPN Laboratory,
Paris XIII University, France
Tel:0033-1-49-40-35-71
FAX:0033-4-48-26-35-68
email: matei@lipn.univ-paris13.fr
Sylvain Meignen*
LJK Laboratory,
University of Grenoble, France
Tel:0033-4-76-51-43-95
email: sylvain.meignen@imag.fr

Abstract- The aim of this paper is to construct a new nonlinear and nonseparable multiscale representation of piecewise continuous bidimensional functions. This representation is based on the definition of a linear projection and a nonlinear prediction operator, which locally adapts to the function to be represented. This adaptivity of the prediction operator proves to be very interesting for image encoding in that it enables a considerable reduction in the number of significant coefficients compared with other representations. Applications of this new nonlinear multiscale representation to image compression and super-resolution conclude the paper.
Keywords nonlinear prediction, cell-average interpolation, image compression, super-resolution

I. INTRODUCTION

For the last few decades, research has been carried out to improve multiscale image representation by departing from traditional linear tensor product (bi)orthogonal wavelet representations. Despite the fact that these representations are known not to be optimal, in terms of the number of non zero detail coefficients they generate, they are supported by powerful encoders such as EZW [1] or EBCOT [2], making them very efficient when applied to image compression. Nevertheless, the fact that wavelet representations generate too many detail coefficients has motivated new research toward more compact representations. For instance:

- Frames having some anisotropic directional selectivity, such as curvelets [3] and contourlets [4].
- Bandlets [5] based on tensor products of wavelet bases combined with locally adapted edge operators.
- Edgeprint approximations [6], that are computed, in the vicinity of an edge, according to a wedge function which locally fits the image.

In all these approaches, and in order to take into account the presence of an edge, the multiscale structure is changed. Here, we introduce a new type of nonlinear multiscale image representation based on cell-average discretization that accurately represents edges using a reduced number of detail coefficients when compared with wavelet transforms but maintains the same quadtree structure as wavelet transform approaches. The main difference with respect to wavelet representations is that the detail coefficients are computed by means of a local and nonlinear prediction operator.

The new nonlinear multiscale representation (NMR) we introduce in this paper is based on cell-average discretization, and is close to the *essentially non oscillatory edge adapted* (ENO-EA) method previously discussed in [7]. The paper is organised as follows, first we recall the general framework for NMR in the cell-average discretization context in Section II. Then we define a new local and nonlinear prediction operator, in Section V, that relies on a novel bidimensional edge detection strategy, detailed in Section III, and on a new technique for edge parameter estimation, described in Section IV. Numerical simulations demonstrating the compactness of an NMR based on such a prediction operator, along with its potential use for image compression and super-resolution, conclude the paper.

II. HARTEN'S NONLINEAR MULTISCALE REPRESENTATION

A. Harten introduced in [8] a strategy to construct NMRs based on two discrete interscale operators, called *projection* and *prediction* operators respectively and denoted by P_{j-1}^{j} and P_{j}^{j-1} in the sequel.

Assuming an image is some function v defined on $[0,1]^2$ and v^j is it's approximation on the grid $(2^{-j}k_1, 2^{-j}k_2)$, $0 \leq k_1, k_2 \leq 2^j - 1$, you first define a linear projection operator P_{j-1}^{j} acting from the

fine to coarse levels, i.e. , $v^{j-1} = P_{j-1}^{j} v^{j}$. In the cell-average framework, this operator is completely characterized since v_k^j is a rescaled version of a local cell-average of v computed as:

$$v_k^j = 2^{2j} \int_{C_k^j} v(x,y) \mathrm{d}x \ \mathrm{d}y, \tag{1}$$

with $C_k^j = [2^{-j}k_1, 2^{-j}(k_1+1)] \times [2^{-j}k_2, 2^{-j}(k_2+1)]$, and where $k = (k_1, k_2)$. In what follows, C_k^j will be called a *cell*. From this, one infers that the projection operator reads:

$$v_k^{j-1} = \frac{1}{4}\left(v_{2k}^j + v_{2k+e_1}^j + v_{2k+e_2}^j + v_{2k+e_1+e_2}^j\right), \tag{2}$$

where e_1 and e_2 are unit vectors oriented to the right and upward, respectively. The prediction operator P_j^{j-1} acts from the coarse to fine levels by computing an 'approximation' $\hat{v}^j$ of v^j from v^{j-1}, i.e. $\hat{v}^j = P_j^{j-1} v^{j-1}$. This operator may be nonlinear. In addition, it is assumed that these operators satisfy the following consistency property:

$$P_{j-1}^{j} P_j^{j-1} = I, \tag{3}$$

i.e. the projection of $\hat{v}^j$ coincides with v^{j-1}:

$$v_k^{j-1} = \frac{1}{4}\left(\hat{v}_{2k}^j + \hat{v}_{2k+e_1}^j + \hat{v}_{2k+e_2}^j + \hat{v}_{2k+e_1+e_2}^j\right). \tag{4}$$

The prediction error $e^j := v^j - \hat{v}^j$ satisfies, from (2) and (3),

$$P_{j-1}^{j} e^j = P_{j-1}^{j} v^j - P_{j-1}^{j} \hat{v}^j = v^{j-1} - v^{j-1} = 0.$$

Hence $e^j \in \mathrm{K}er(P_{j-1}^{j})$ and, using a basis E of this kernel, one writes e^j in a non-redundant way to obtain the *detail coefficients* d^{j-1}, satisfying $e^j = E d^{j-1}$. Thus v^j is equivalent to (v^{j-1}, d^{j-1}). In practice, this non-redundancy means the size of the data is preserved through decomposition. Assuming the size of the original image is $2^J \times 2^J$ and iterating the proposed nonlinear procedure from the initial data v^J, we obtain its NMR

$$\mathcal{M} v^J = (v^0, d^0, \ldots, d^{J-1}). \tag{5}$$

Then you can state that a prediction operator reproduces bidimensional polynomials of global degree N if $d^j = 0$ when $v = p$, with $p(x,y) = \sum_{0 \le i,j \le N} p_{i,j} x^i y^j$. Linear and nonlinear prediction operators satisfying polynomial reproduction have been extensively used for image encoding, see [7]. In the present paper, we are interested in building a new nonlinear prediction operator that generates very few detail coefficients when v is some kind of bidimensional piecewise polynomial.

III. EDGE DETECTION

To begin, we consider edge detection at level $j-1$ since the prediction at level j is based only on the information available at level $j-1$. In this section, we consider step-edges, modeled by straight lines separating regions with constant gray level, that is, on a cell C_k^{j-1} containing an edge, the function v is assumed to have the form

$$v(x,y) = A\chi_{\{y\geq h(x)\}}(x,y) + B\chi_{\{y<h(x)\}}(x,y), \tag{6}$$

with $h(x) = mx + n$, $\chi_C(x,y)$ the indicator function of C, and A and B some constants.

The edge detection mechanism makes use of the one dimensional cost functions whose descriptions follow:

$$H_k^{j-1} := |v_k^{j-1} - v_{k-e_1}^{j-1}| + |v_{k+e_1}^{j-1} - v_k^{j-1}| \tag{7}$$

$$V_k^{j-1} := |v_k^{j-1} - v_{k-e_2}^{j-1}| + |v_{k+e_2}^{j-1} - v_k^{j-1}|. \tag{8}$$

For each k, one defines:

$$l_{h,k} = \underset{l}{\text{argmin}} \left\{ H_{k+le_1}^{j-1}, l \in \{-1,0,1\} \right\} \tag{9}$$

$$l_{v,k} = \underset{l}{\text{argmin}} \left\{ V_{k+le_2}^{j-1}, l \in \{-1,0,1\} \right\}. \tag{10}$$

$l_{h,k}$ (resp. $l_{v,k}$) corresponds to the horizontal (resp. vertical) shift with respect to k associated with the minimal cost function among $H_{k-e_1}^{j-1}$, H_k^{j-1} and $H_{k+e_1}^{j-1}$ (resp. $H_{k-e_2}^{j-1}$, H_k^{j-1} and $H_{k+e_2}^{j-1}$). As we will make use of different strategies depending on the orientation of the edge, we need an estimate of the latter. To this end, we define:

$$\begin{aligned} H_{k,1}^{j-1} &= (v_{k+e_2-e_1}^{j-1} + 2v_{k+e_2}^{j-1} + v_{k+e_2+e_1}^{j-1}) - (v_{k-e_2-e_1}^{j-1} + 2v_{k-e_2}^{j-1} + v_{k-e_2+e_1}^{j-1}) \\ V_{k,1}^{j-1} &= (v_{k+e_2-e_1}^{j-1} + 2v_{k-e_1}^{j-1} + v_{k-e_2-e_1}^{j-1}) - (v_{k+e_2+e_1}^{j-1} + 2v_{k+e_1}^{j-1} + v_{k-e_2+e_1}^{j-1}), \end{aligned}$$

and then estimate the edge orientation by $m_k^{j-1} := \frac{V_{k,1}^{j-1}}{H_{k,1}^{j-1}}$, if $H_{k,1}^{j-1}$ is non zero, and by $\pm\infty$ depending on the sign of $V_{k,1}^{j-1}$, otherwise. Then you consider the set E^{j-1} of cells C_k^{j-1} satisfying either:

$$\begin{aligned} &a) \quad l_{h,k-e_1} = -1 \text{ and } l_{h,k+e_1} = 1 \\ &b) \quad H_k^{j-1} > H_{k-e_1}^{j-1} \text{ and } H_k^{j-1} > H_{k+e_1}^{j-1}, \\ &c) \quad m_k^{j-1} > 1 \end{aligned} \tag{11}$$

or

$$\begin{aligned} a) \qquad & l_{v,k-e_2} = -1 \text{ and } l_{v,k+e_2} = 1 \\ b) \quad & V_k^{j-1} > V_{k-e_2}^{j-1} \text{ and } V_k^{j-1} > V_{k+e_2}^{j-1}, \\ c) \qquad & m_k^{j-1} \leq 1. \end{aligned} \tag{12}$$

This means that when the slope of the edge is estimated to be larger than 1 the edge detection is performed horizontally, and vertically otherwise. In the case of the step-edge model described by (6), it is easy to

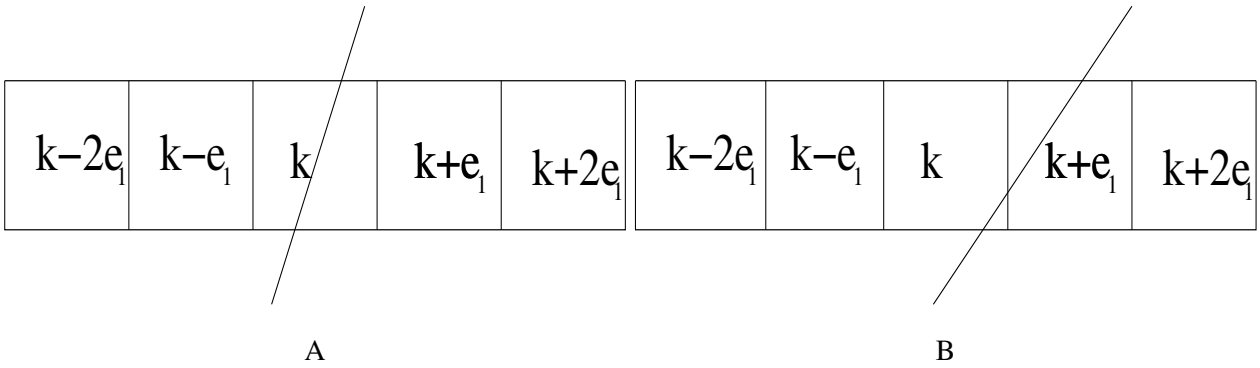


Fig. 1. A: Step-edge crossing cell C_k^{j-1}; B: Step-edge crossing cells C_k^{j-1} and $C_{k+e_1}^{j-1}$.

check that a cell containing an edge actually satisfies the properties (11) a) or (12) a), depending on the slope of the edge. As an illustration of property (11) a), we display, in Figure 1, a step-edge crossing either one or two successive cells. In such cases, $l_{h,k-e_1} = -1$ and $l_{h,k+e_1} = 1$ because the cell-average differences are smaller when you move away from the step-edge. It is worth noting here that, when the edge crosses three or more successive cells horizontally, (12) is used instead, because then m_k^{j-1} is smaller than 1. Furthermore, requirement (11) b) enables localization of the edge on a single cell, as we will now prove. Without loss of generality, we consider an edge, with a slope larger than one, crossing C_k^{j-1}. Then, we have the following result:

Lemma 3.1: Assume that v is a step-edge function with amplitude δ, as defined in (6), with a slope larger than one, crossing C_k^{j-1}. If, among $C_{k-e_1}^{j-1}, C_k^{j-1}, C_{k+e_1}^{j-1}$, only C_k^{j-1} is actually crossed by the edge, then only C_k^{j-1} belongs to E^{j-1}. Otherwise, if two cells among $C_{k-e_1}^{j-1}$, C_k^{j-1}, or $C_{k+e_1}^{j-1}$ are crossed by the edge then, again, only one of these cells belongs to E^{j-1}.

Proof: We consider two different cases separately:

Case 1. Only C_k^{j-1} is crossed by the edge. Since we have

$$H_k^{j-1} = |v_{k+e_1}^{j-1} - v_k^{j-1}| + |v_k^{j-1} - v_{k-e_1}^{j-1}| = |B - A|, \tag{13}$$

$$|v_{k+2e_1}^{j-1} - v_{k+e_1}^{j-1}| = 0 \ \text{ or } \ |v_{k-e_1}^{j-1} - v_{k-2e_1}^{j-1}| = 0. \tag{14}$$

It follows that $H_{k-e_1}^{j-1} < H_k^{j-1}$ and $H_k^{j-1} > H_{k+e_1}^{j-1}$. Now, since $H_{k-2e_1}^{j-1} = 0$ and $H_{k+2e_1}^{j-1} = 0$, $l_{h,k-e_1} = -1$ and $l_{h,k+e_1} = 1$, which means that C_k^{j-1} belongs to E^{j-1}.

Case 2. Cells C_k^{j-1} and $C_{k+e_1}^{j-1}$ are crossed by the edge. Then, we have $H_{k-e_1}^{j-1} < H_k^{j-1}$ and $H_{k+2e_1}^{j-1} < H_{k+e_1}^{j-1}$. When $H_k^{j-1} > H_{k+e_1}^{j-1}$, $l_{h,k-e_1} = -1$ and $l_{h,k+e_1} = 1$, meaning that $C_k^{j-e_1}$ is an E^{j-1} cell, else, if $H_{k+e_1}^{j-1} < H_k^{j-1}$, $C_{k+e_1}^{j-1}$ belongs to E^{j-1}. The same reasoning can be made when the two cells crossed by the edge are $C_{k-e_1}^{j-1}$ and C_k^{j-1}. ■

Note that we could write a similar lemma assuming the edge slope to be smaller than one. In contrast, if we used only (11) a) and c) then, for cases corresponding to Figure 1 B, the edge could be localized on two successive cells.

Finally, from the set E^{j-1}, we define the subset:

$$\begin{aligned}\mathcal{E}^{j-1} = \Big\{ C_k^{j-1} \in E^{j-1},\ & C_{k-e_1+e_2}^{j-1}, C_{k+e_1-e_2}^{j-1} \notin E^{j-1} \text{ if } m_k^{j-1} > 0, \\ & C_{k+e_1+e_2}^{j-1}, C_{k-e_1-e_2}^{j-1} \notin E^{j-1} \text{ if } m_k^{j-1} < 0 \Big\}.\end{aligned} \tag{15}$$

Considering

$$\begin{aligned} E_V^{j-1} &= \left\{ C_k^{j-1} \in \mathcal{E}^{j-1} \text{ s. t. } |m_k^{j-1}| \geq 1 \right\} \text{ and} \\ E_H^{j-1} &= \left\{ C_k^{j-1} \in \mathcal{E}^{j-1} \text{ s. t. } |m_k^{j-1}| < 1 \right\}, \end{aligned}$$

we have $\mathcal{E}^{j-1} = E_H^{j-1} \cup E_V^{j-1}$, which we call the set of *edge-cells* in what follows. An illustration of the extra constraint placed on E^{j-1} to obtain $\mathcal{E}^{j-1}$ is shown in Figure 2 A, when $m_k^{j-1} > 0$. Note that to define E^{j-1} as we did is crucial since, if it were defined using only (11) a) and c) or (12) a) and c), false detections would create holes in the set $\mathcal{E}^{j-1}$ where there should be a continuum of cells, as shown in Figure 2 C. This does not happen when $\mathcal{E}^{j-1}$ is defined using (11) a)-c) or (12) a)-c) (see Figure 2 D). Finally, we also note that adding conditions (11) b) and (12) b) improves the robustness of the detection since we get many fewer false detections inside smooth regions (compare again Figure 2 C and D). By using this new edge detector, we remove some false detections, but also some real ones ; subsequently we will see how to recover the latter.

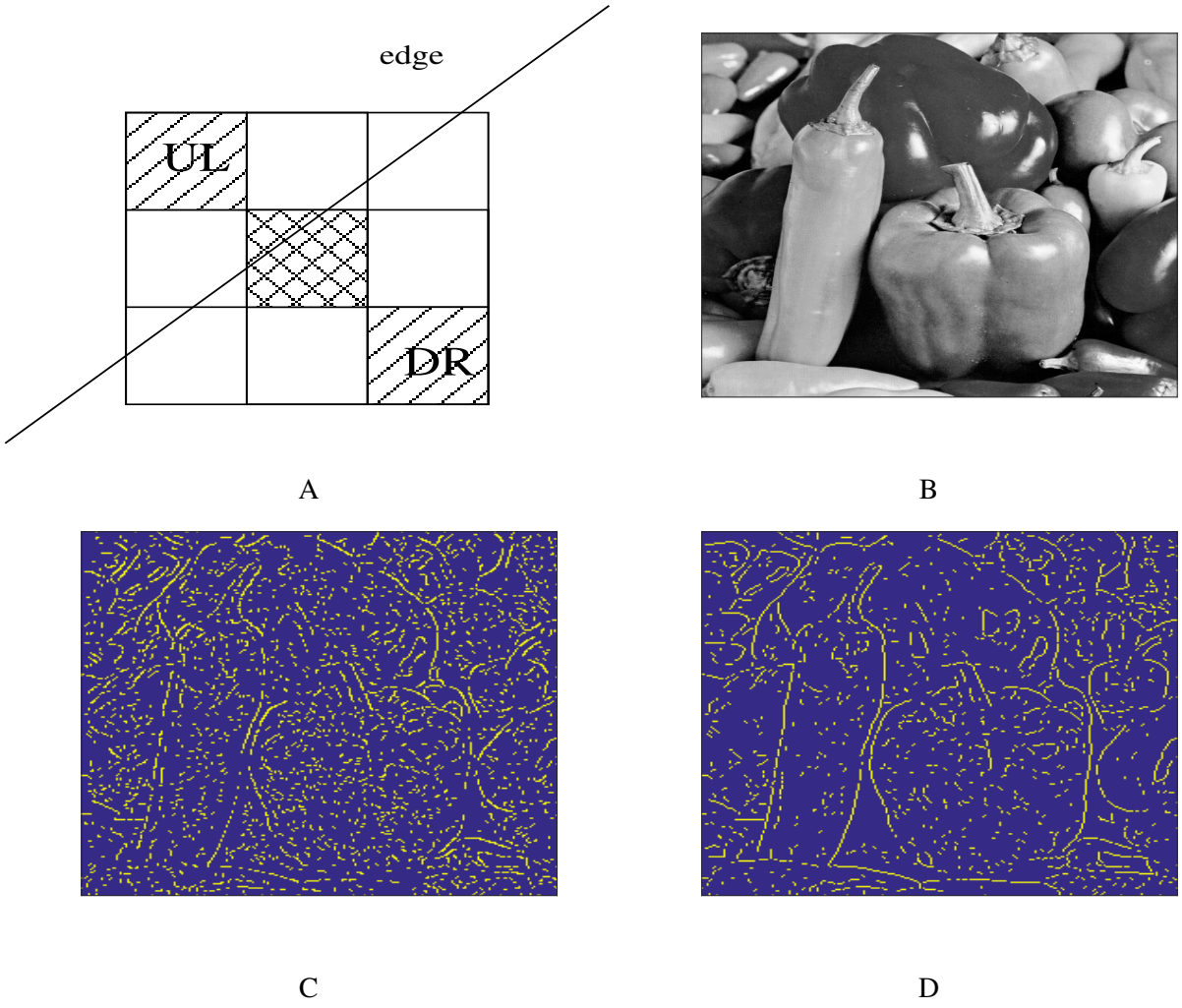


Fig. 2. A: Illustrates how the set $\mathcal{E}^{j-1}$ is built when $m_k^{j-1} > 0$, the cells denoted by UL (for upper left) and DR (for down right) should not belong to E^{j-1}; B: 512×512 image of peppers; C: Cells in $\mathcal{E}^{j-1}$ computed, at level $J-1$ and for image B, using only (11) a) and c) or (12) a) and c); D: Idem as C but using (11) a)-c) or (12) a)-c) to compute edge-cells.

IV. Estimation of Edge Parameters and Definition of S-cells

We assume, in this section, that C_k^{j-1} is an edge-cell and we consider the more general step-edge model for the function v inside C_k^{j-1}, i.e.

$$v(x,y) := a(x,y)\chi_D(x,y) + b(x,y)\chi_{D^c}(x,y), \tag{16}$$

where $D = \{(x,y), y \geq h(x)\}$ and $D^c = C_k^{j-1} \setminus D$. $h(x)$ will either be approximated by a straight line $p_{1,k}(x) = lx + q$, or by a second order polynomial $p_{2,k}(x) = mx^2 + nx + p$, where index k indicates the polynomial is different at each location k. As for $a(x,y)$ and $b(x,y)$, they will either be approximated by

constants A and B, or by biquadratic polynomials $p_{a,k}$ and $p_{b,k}$, i.e. $p_{a,k}(x,y) = \sum_{0\leq i,q\leq 2} a_{i,q}x^i y^q$ ($p_{b,k}$ being defined similarly from coefficients $b_{i,q}$). We will see later in which instances the analysis benefits from considering a curve model for the edge, and non constant $p_{a,k}$ and $p_{b,k}$. Regarding the edge model, we shall say that the approach proposed in [7], subsequently denoted by ENO-EA, involves constant approximations for $a(x,y)$ and $b(x,y)$, and edges locally modeled by straight lines. Nevertheless, this model is only valid when the scale of study 2^{-j+1} is lower than some critical scale h_c whose definition is now recalled for the sake of consistency [7]. Let v be a piecewise smooth function containing a curved edge I. For a cell C_k^{j-1} crossed by I, one defines its neighborhood D_k^{j-1} by

$$D_k^{j-1} \;:=\; \left\{ C^{j-1}_{k+me_1+ne_2}, m,n \in \{-1,0,1\} \right\}, \tag{17}$$

and, given an arclength parameterization $\gamma(t)$ of I, one considers

$$I_k^{j-1} = \{t; \gamma(t) \in D_k^{j-1}\},$$

the set of values t for which I crosses D_k^{j-1}. Following [7], the critical scale of study is given by

$$\frac{1}{h_c} := \frac{\sup_{D_k^{j-1}\setminus I} \|\nabla v(x,y)\|}{\sup_{I_k^{j-1}} \|[v](t)\|} + \sup_{I_k^{j-1}} |\gamma''(t)|, \tag{18}$$

where $\|[v](t)\|$ is the amplitude of the step at $\gamma(t)$. Our goal is to show that as, in practical situations such as image analysis, one cannot fix the scale of study to a value smaller than h_c, it is worth considering more complex edge models than straight lines. This will become clearer in the numerical section.

A. Estimation of Edge Parameters with ENO-EA Method

In ENO-EA method, function $h(x)$ is estimated on C_k^{j-1} belonging to E_V^{j-1}, and assuming m_k^{j-1} is positive, by an affine polynomial $p_{1,k}$ computed as follows (the same kind of computation can easily be transposed to any other case). To estimate $h(x)$, one needs two different points which are obtained using the following arguments. Assuming $C^{j-1}_{k-2e_1}$ and $C^{j-1}_{k+2e_1}$ are not crossed by an edge, one determines a point (x_0, y_0) on the edge by putting $y_0 = \frac{1}{2}$ and then by using the following consistency property:

$$v^{j-1}_{k-e_1} + v^{j-1}_k + v^{j-1}_{k+e_1} = x_0 v^{j-1}_{k-2e_1} + (3-x_0) v^{j-1}_{k+2e_1}. \tag{19}$$

To explain where this equality arises from, we refer to Figure 3 A in which the two hatched areas are equal. This implies that the normalized integral of v on the union of $C^{j-1}_{k-e_1}$, C^{j-1}_k and $C^{j-1}_{k+e_1}$, i.e. $v^{j-1}_{k-e_1}+v^{j-1}_k+v^{j-1}_{k+e_1}$, is equal to $x_0 v^{j-1}_{k-2e_1}+(3-x_0)v^{j-1}_{k+2e_1}$ because, in the ENO-EA method, v is supposed to be piecewise constant on each side of the edge, i.e. $v^{j-1}_{k-2e_1} = A$ and $v^{j-1}_{k+2e_1} = B$. So, in that framework and taking into account the scale factor, we get that the edge passes through $(\frac{k_1-1+x_0}{2^{j-1}}, \frac{k_2+1/2}{2^{j-1}})$. Now

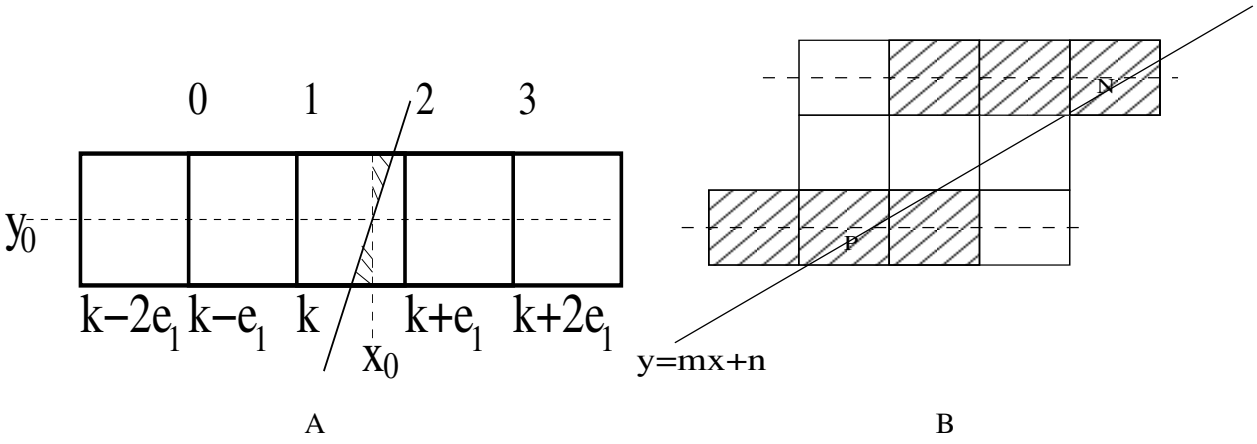


Fig. 3. A: Illustration of the consistency property on the rows, the digits above the figure correspond to the scale for x_0; B: Cells involved in the computation of edge parameters (hatched regions) along with the actual edge.

we remark that, since $m_k^{j-1} > 1$, the edge crossing C_k^{j-1} can only cross, on the row indexed by k_2+1, $C_{k+e_2}^{j-1}$, $C_{k+e_1+e_2}^{j-1}$ and $C_{k+2e_1+e_2}^{j-1}$ and, on the row indexed by k_2-1, $C_{k-2e_1-e_2}^{j-1}$, $C_{k-e_1-e_2}^{j-1}$ and $C_{k-e_2}^{j-1}$ (see Figure 3 B). We define two points N and P, using consistency rules of type (19) on rows indexed by k_2+1 and k_2-1 respectively. Then $h(x)$ is defined as the straight line passing through N and P. Also, with this method, one considers $p_{a,k}(x,y) = v_{k-e_1+e_2}^{j-1}$ and $p_{b,k}(x,y) = v_{k+e_1-e_2}^{j-1}$.

B. New Estimation of Edge Parameters

In the following sections, we will make extensive use of the unique biquadratic polynomial p_k interpolating the average of v on the cells making up D_k^{j-1}, and defined as follows:

$$v_{k+le_1+pe_2}^{j-1} \quad = \quad 2^{2(j-1)} \int_{C_{k+le_1+pe_2}^{j-1}} p_k(x,y)\ \mathrm{d}x\mathrm{d}y \quad l,p \in \{-1,0,1\}. \tag{20}$$

1) Computation of polynomials $p_{a,k}(x,y)$ and $p_{b,k}(x,y)$: Here we explain how we compute $p_{a,k}(x,y)$ and $p_{b,k}(x,y)$ on C_k^{j-1} in E_V^{j-1}, assuming m_k^{j-1} is positive, (the same kind of computation could be carried out for the other cases). We first consider that the cell studied is such that (see Figure 4 A for an illustration):

$$\left(D_{k+2(e_2-e_1)}^{j-1} \cup D_{k+2(e_1-e_2)}^{j-1}\right) \ \cap \ \mathcal{E}^{j-1} = \emptyset. \tag{21}$$

If (21) is satisfied, polynomials $p_{a,k}$ and $p_{b,k}$ are defined as $p_{k-2e_1+2e_2}$ and $p_{k+2e_1-2e_2}$ respectively, (where p_k was introduced in (20)). Note that if $a(x,y)$ and $b(x,y)$ are actually biquadratic polynomials on each side of the edge then $p_{a,k} = a$ and $p_{b,k} = b$. When (21) is not satisfied, we put $p_{a,k}(x,y) = A$ and $p_{b,k}(x,y) = B$, with $A = v_{k-e_1+e_2}^{j-1}$ and $B = v_{k+e_1-e_2}^{j-1}$.

2) Estimation of $h(x)$: to estimate $h(x)$ when it is modeled by $p_{2,k}(x)$ on C_k^{j-1} in E_V^{j-1}, still assuming m_k^{j-1} is positive, we define three points N, M and P on the rows indexed by k_2+1, k_2 and k_2-1, respectively, by using a generalization of rule (19). It is important to remark here that, by analogy, points N, M and P are computed using the consistency property of type (19) on the columns indexed by k_1-1, k_1 and k_1+1 for cells in E_H^{j-1} (this can be viewed as the transposed situation to the case studied).

Let us now write the generalization of the consistency rule (19) on the row indexed by k_2. We first integrate polynomials $p_{a,k}$ and $p_{b,k}$ with respect to y, to define:

$$A_{q,k}^{j-1}(x) := \int_{\frac{q}{2^{j-1}}}^{\frac{q+1}{2^{j-1}}} p_{a,k}(x,y)\ \mathrm{d}y,\ B_{q,k}^{j-1}(x) := \int_{\frac{q}{2^{j-1}}}^{\frac{q+1}{2^{j-1}}} p_{b,k}(x,y)\ \mathrm{d}y, \tag{22}$$

and then compute the abscissa x_M of M, through the following consistency rule:

$$v_{k-e_1}^{j-1} + v_k^{j-1} + v_{k+e_1}^{j-1} = L_{k_1,k_2}^{j-1}(x_M), \tag{23}$$

where

$$L_{k_1,k_2}^{j-1}(x_M) := 2^{2(j-1)} \left(\int_{\frac{k_1-1}{2^{j-1}}}^{\frac{k_1-1+x_M}{2^{j-1}}} A_{k_2,k}^{j-1}(x)\ dx + \int_{\frac{k_1-1+x_M}{2^{j-1}}}^{\frac{k_1+2}{2^{j-1}}} B_{k_2,k}^{j-1}(x)\ dx \right). \tag{24}$$

From Equation (23), x_M is a root on $[0,3]$ of a third order polynomial. The motivation for integrating biquadratic polynomials $p_{k,a}$ and $p_{k,b}$ in the vertical direction is that, in images, the grey level often varies smoothly along an edge. This also enables us to simplify the search for point M into a one-dimensional problem. Indeed, we again define $y_M := \frac{k_2+1/2}{2^{j-1}}$ and use the same arguments as in the piecewise constant case to justify this new consistency rule (see Section IV-A). This means that point M is not on the edge curve, but sufficiently close to it to bring a significant improvement in terms of edge estimation compared with a straight line estimation, as we will see in the numerical section.

Then, in a fashion similar to the ENO-EA case, (assuming the slope of the edge does not vary too rapidly along the vertical direction), potential edge-cells on the row indexed by k_2+1 (resp. k_2-1) are $C_{k+e_2}^{j-1}$, $C_{k+e_1+e_2}^{j-1}$ and $C_{k+2e_1+e_2}^{j-1}$ (resp. $C_{k-2e_1-e_2}^{j-1}$, $C_{k-e_1-e_2}^{j-1}$ and $C_{k-e_2}^{j-1}$), an illustration of this is shown in Figure 4 B. The consistency rule (23) on rows indexed by k_2+1 and k_2-1 then reads as follows:

$$\begin{aligned} v_{k+e_2}^{j-1} + v_{k+e_2+e_1}^{j-1} + v_{k+e_2+2e_1}^{j-1} &= L_{k_1+1,k_2+1}^{j-1}(x_N) \text{ and} \\ v_{k-e_1-2e_2}^{j-1} + v_{k-e_2-e_1}^{j-1} + v_{k-e_2}^{j-1} &= L_{k_1-1,k_2-1}^{j-1}(x_P), \end{aligned} \tag{25}$$

so that both x_N and x_P are roots of third order polynomials on $[0,3]$, and $y_N = \frac{k_2+3/2}{2^{j-1}}$ while $y_P = \frac{k_2-1/2}{2^{j-1}}$. Finally we should stress that, to define all of these consistency rules, we use polynomials $p_{a,k}$ and $p_{b,k}$ on each of the rows indexed by k_2-1, k_2 and k_2+1.

When (21) is not satisfied, to take into account a potential variation of the gray level in the direction of the edge, we adopt the following strategy:

- In $L_{k_1+1,k_2+1}(x_N)$, $A^{j-1}_{k_2+1,k}(x)$ is replaced by $2^{-2(j-1)}v^{j-1}_{k-e_1+e_2}$ and $B^{j-1}_{k_2+1,k}(x)$ by $2^{-2(j-1)}v^{j-1}_{k+3e_1+e_2}$, provided $C^{j-1}_{k-e_1+e_2}, C^{j-1}_{k+3e_1+e_2} \notin \mathcal{E}^{j-1}$.
- In $L_{k_1,k_2}(x_M)$, $A^{j-1}_{k_2,k}(x)$ is replaced by $2^{-2(j-1)}v^{j-1}_{k-2e_1}$ and $B^{j-1}_{k_2,k}(x)$ by $2^{-2(j-1)}v^{j-1}_{k+2e_1}$, provided $C^{j-1}_{k-2e_1}, C^{j-1}_{k+2e_1} \notin \mathcal{E}^{j-1}$.
- In $L_{k_1-1,k_2-1}(x_P)$, $A^{j-1}_{k_2-1,k}(x)$ is replaced by $2^{-2(j-1)}v^{j-1}_{k-3e_1-e_2}$ and $B^{j-1}_{k_2-1,k}(x)$ by $2^{-2(j-1)}v^{j-1}_{k+e_1-e_2}$, provided $C^{j-1}_{k-3e_1-e_2}, C^{j-1}_{k+e_1-e_2} \notin \mathcal{E}^{j-1}$.

Note that this framework is easily transposable to E^{j-1}_V cells associated with negative m^{j-1}_k. Indeed, in that case, the same reasoning applied previously can be carried out noting that the cells potentially containing an edge are those indicated in Figure 4 C (assuming the central cell is C^{j-1}_k).

While trying to compute edge parameters, we determine a subset $\tilde{\mathcal{E}}^{j-1}$ of $\mathcal{E}^{j-1}$, corresponding to the edge-cells for which the parameters are actually computable, so that $\mathcal{E}^{j-1} = \tilde{\mathcal{E}}^{j-1} \cup \mathcal{D}^{j-1}$. It is worth remarking here that T-junctions will be contained in the set $\mathcal{D}^{j-1}$, because these edge-cells are typically outside the proposed step-edge model.

C. Definition of S-cells

In our formalism, *S-cells* are going to be defined as the cells actually crossed by an edge for which the computation of the edge parameters is possible. The set of S-cells will definitely contain $\mathcal{E}^{j-1}$, but many more cells are crossed by an edge as explained earlier. To build this set, we proceed iteratively starting from $\tilde{\mathcal{E}}^{j-1}$ and using the parameters of the edge computed on the cells making up that set. To illustrate the procedure, we consider the case of a cell C^{j-1}_k in $\tilde{\mathcal{E}}^{j-1} \cap E^{j-1}_V$. As already noted, the edge studied can also potentially cross cells $C^{j-1}_{k-e_1}$ and $C^{j-1}_{k+e_1}$. One then simply checks whether the edge estimated on cell C^{j-1}_k crosses one of the latter cells. If so, one tries to recompute the parameters of the edge on this cell and, if it is successful, the latter is added to the set of S-cells. The same type of procedure is applied to the set of cells $\tilde{\mathcal{E}}^{j-1} \cap E^{j-1}_H$, for which we check whether the edge computed on C^{j-1}_k crosses $C^{j-1}_{k-e_2}$ or $C^{j-1}_{k+e_2}$.

Finally, because the addition of new S-cells is based only on an estimate of the edge parameters, we may add cells in such a way that some S-cells C^{j-1}_k such that $m^{j-1}_k > 0$ have neighbors $C^{j-1}_{k-e_1+e_2}$ and $C^{j-1}_{k+e_1-e_2}$ also in the set of S-cells. To avoid this, when $m^{j-1}_k > 1$, we keep C^{j-1}_k in the set if $H^{j-1}_k > H^{j-1}_{k-e_1+e_2}$ and $H^{j-1}_k > H^{j-1}_{k+e_1-e_2}$ and, when $m^{j-1}_k < 1$, we keep C^{j-1}_k in the set if $V^{j-1}_k > V^{j-1}_{k-e_1+e_2}$ and $V^{j-1}_k > V^{j-1}_{k+e_1-e_2}$. The same kind of reasoning is applied when m^{j-1}_k is negative.

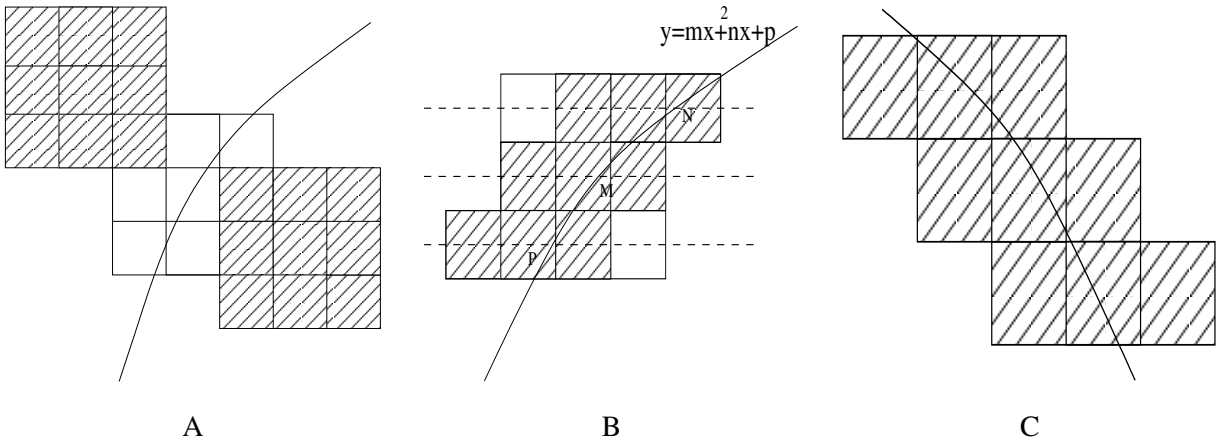


Fig. 4. A: The upper and lower hatched regions represent the neighborhood $D^{j-1}_{k-2(e_1-e_2)}$ and $D^{j-1}_{k+2(e_1-e_2)}$ respectively used to compute $p_{a,k}$ and $p_{b,k}$ when condition (21) is satisfied; B: Points M, N, P along with the polynomial $p_{2,k}(x) = mx^2 + nx + p$ interpolating these points, the hatched zone corresponds to the region potentially crossed by the edge; C: The hatched region corresponds to the cells that may be intersected by the edge when m_k^{j-1} is smaller than -1.

V. Definition of the Nonlinear Prediction Operator

Once the S-cells are determined, the other cells are automatically labeled by O. In the following sections, we detail how we build the prediction operator knowing the nature of the cells, and their corresponding edge parameters, (for S-cells).

A. Prediction on O-cells

To build the prediction operator on an O-cell C_k^{j-1}, we use a biquadratic polynomial p_{k^*} approximating function v. This is built by interpolation of the average of v on the cells making up the stencil $D_{k^*}^{j-1}$. To find the latter, the ENO-EA method needs the definition of $F_k^{j-1} = H_{k,2}^{j-1} + V_{k,2}^{j-1}$, where

$$H_{k,2}^{j-1} := \sum_{p=-1}^{1} \mid v^{j-1}_{k-e_1+pe_2} - v^{j-1}_{k+pe_2} \mid + \mid v^{j-1}_{k+pe_2} - v^{j-1}_{k+e_1+pe_2} \mid$$

$$V_{k,2}^{j-1} := \sum_{l=-1}^{1} \mid v^{j-1}_{k+le_1-e_2} - v^{j-1}_{k+le_1} \mid + \mid v^{j-1}_{k+le_1} - v^{j-1}_{k+le_1+e_2} \mid,$$

which corresponds to the sum of all the moduli of the first order vertical and horizontal finite differences on the stencil D_k^{j-1}. The ENO-EA method [9] finds k^* associated with the least oscillatory 3×3 stencil containing C_k^{j-1}, i.e.

$$k^* = \underset{q}{\operatorname{argmin}} \left\{ F_q^{j-1},\ q = k + le_1 + pe_2\ l, p \in \{-1, 0, 1\} \right\}. \tag{26}$$

However, it is well known that this method for stencil selection is particularly unstable, since a slight change in the data may entail a change of stencil. To overcome this problem, we put $k^* = k$ when D_k^{j-1} does not contain a S-cell. In other cases, rather than using the cost function F_k^{j-1}, we preferentially use the following strategy that minimizes the displacement with respect to the centered stencil:

- Find k^* associated with the first stencil not containing a S-cell (if any) in the list $\{D_{k-e_2}^{j-1}, D_{k+e_1}^{j-1}, D_{k+e_2}^{j-1}, D_{k-e_1}^{j-1}\}$.
- If no stencil is selected by the above process, find k^* associated with the first stencil not containing a S-cell (if any) in the list $\{D_{k+e_1+e_2}^{j-1}, D_{k+e_1-e_2}^{j-1}, D_{k-e_1-e_2}^{j-1}, D_{k-e_1+e_2}^{j-1}\}$.

If the above mechanism does not find any suitable stencil, we again put $k^* = k$. By using such a procedure based on the location of S-cells instead of F_k^{j-1}, we no longer use a cost function to determine the stencils we use for prediction on O-cells. Note also that, to choose $D_{k^*}^{j-1}$ means we use polynomial p_{k^*}, defined as follows, for prediction:

$$\hat{v}_{2k+le_1+qe_2}^j = 2^{2j} \int_{C_{2k+le_1+qe_2}^j} p_{k^*}(x,y)\ \mathrm{d}x\mathrm{d}y,\ \ l,q \in \{0,1\}.$$

To illustrate this procedure, in the case of a straight line edge, we display, in Figure 5 A, S-cells labeled by 3, the O-cells labeled by 1, on which p_k is used to predict, and the O-cells labeled by 1, on which stencil p_{k^*} is used instead.

B. Prediction on S-cells

We now present the prediction scheme on an S-cell, we model the step-edge by a second order polynomial and the image, on each side of the edge, by a biquadratic polynomial. This means that $C_k^{j-1} \in \tilde{\mathcal{E}}^{j-1}$ is associated with two biquadratic polynomials $p_{a,k}$ and $p_{b,k}$, and a second order polynomial $p_{2,k}$, which define an approximation of v inside C_k^{j-1} of the form:

$$p_S(x,y) := p_{a,k}(x,y)\chi_{\{y \geq p_{2,k}(x)\}}(x,y) + p_{b,k}(x,y)\chi_{\{y < p_{2,k}(x,y)\}}(x,y). \tag{27}$$

We then estimate the average at level j, on the four subcells of cell C_k^{j-1} by:

$$\hat{v}_{2k+le_1+qe_2}^j = 2^{2j} \int_{C_{2k+le_1+qe_2}} p_S(x,y)\ \mathrm{d}x\mathrm{d}y,\ \ l,q \in \{0,1\}.$$

However, the prediction based on the piecewise biquadratic polynomial p_S is, strictly speaking, not consistent with (4), because $p_{2,k}$ is only an estimate of the real edge. Therefore, we force the prediction into being consistent on C_k^{j-1} by defining m_a, m_b, m_c and m_d, the final prediction on the four subcells of C_k^{j-1}, (as depicted in Figure 5 B), as follows. If m_k^{j-1} is positive and if $p_S(\frac{k_1+1/2}{2^{j-1}}, \frac{k_2+1/2}{2^{j-1}}) = p_{b,k}(\frac{k_1+1/2}{2^{j-1}}, \frac{k_2+1/2}{2^{j-1}})$, we are in the situation depicted in Figure 5 B, and:

$$m_a := \hat{v}_{2k}^j,\ m_d := \hat{v}_{2k+e_1}^j,\ m_c := \hat{v}_{2k+e_1+e_2}^j,\ m_b = 4v_k^{j-1} - (m_a + m_b + m_c). \tag{28}$$

In contrast, when $p_S(\frac{k_1+1/2}{2^{j-1}}, \frac{k_2+1/2}{2^{j-1}}) = p_{a,k}(\frac{k_1+1/2}{2^{j-1}}, \frac{k_2+1/2}{2^{j-1}})$, we leave m_a, m_b and m_c unchanged and use (4) to define m_d. The same kind of computation can be carried out when m_k^{j-1} is negative. It is important to note that, when the edge intersects only one of the four subcells, such a procedure enables the lack of accuracy in the determination of the location of the edge to be compensated for.

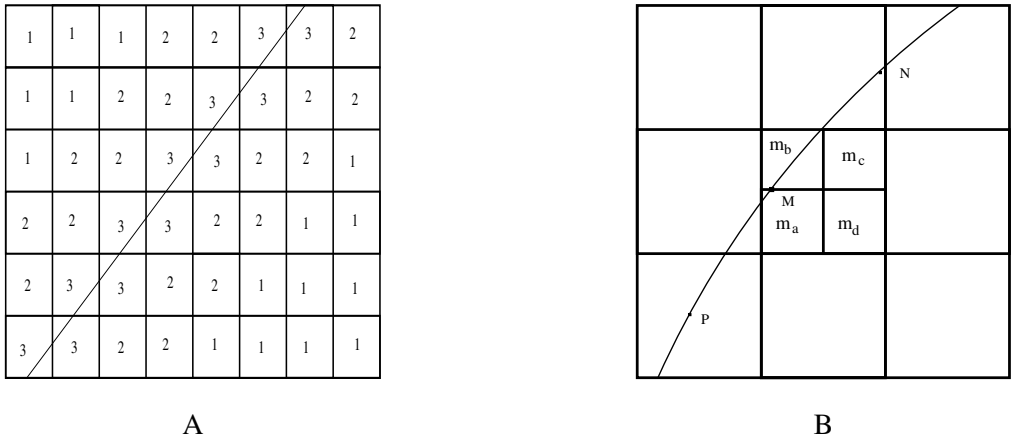


Fig. 5. A: An illustration, in the case of a straight line edge, of the different type of cells (3: S-cells, 2: O-cells associated with shifted stencil for prediction, 1: O-cells associated with centered stencil for prediction); B: Illustration of the procedure that ensures the prediction satisfies (4).

C. Nonlinear Multiscale Representation

Once a prediction operator is defined, to obtain a non redundant representation, one rewrites the prediction error on a basis of the projection operator. Denoting $e_k^j := v_k^j - \hat{v}_k^j$, the consistency property satisfied by the prediction operator implies:

$$e_{2k}^j + e_{2k-e_1}^j + e_{2k-e_2}^j + e_{2k-e_1-e_2}^j = 0,$$

and then, by rearranging the prediction error into the detail coefficients, we obtain:

$$\begin{aligned} d_k^{1,j-1} &:= \frac{1}{4}(e_{2k-e_1-e_2}^j - e_{2k-e_2}^j + e_{2k-e_1}^j - e_{2k}^j) \\ d_k^{2,j-1} &:= \frac{1}{4}(e_{2k-e_1-e_2}^j + e_{2k-e_2}^j - e_{2k-e_1}^j - e_{2k}^j) \\ d_k^{2,j-1} &:= \frac{1}{4}(e_{2k-e_1-e_2}^j - e_{2k-e_2}^j - e_{2k-e_1}^j + e_{2k}^j). \end{aligned}$$

One can easily check that $d^{1,j-1}$, $d^{2,j-1}$ and $d^{3,j-1}$ respectively correspond to detail coefficients in the horizontal, vertical and oblique directions, so that we have the same type of representation as in the

orthogonal wavelet transform. This implies that we should be able to use powerful encoders such as EZW [1] to encode the decomposition, since these encoders are designed for such a quadtree structure.

VI. Experimental Results

In the previous section, we defined the prediction operator on O-cells using biquadratic polynomials based on a stencil either centered or shifted, and, on S-cells, using piecewise polynomials. Our goal here is to investigate how different choices for the prediction operator impact the NMR. To this end, we consider first NMRs built assuming the image is piecewise constant on each side of a straight line edge, the parameters of the edge being computed using the ENO-EA strategy detailed in Section IV-A. The associated prediction is denoted by SR_1 in the simulations presented, SR standing for subcell resolution. Second, we also investigate the case where the image is supposed to be piecewise constant on each side of a curved-edge, locally modeled by a second order polynomial (prediction denoted by SR_2). Finally, we also study the case of an image modeled by a biquadratic polynomial on each side of a curved-edge (prediction denoted by SR_3). The idea underlying the motivation for using of nonlinear prediction operators is to drastically reduce the number and the amplitude of the detail coefficients generated by the presence of the edges. We first illustrate this by studying the decay of the detail coefficients on various synthetic images. Then, we investigate the NMR performance in terms of compression, using an adapted version of the EZW algorithm [1]. Finally, we show how our approach can be used for super-resolution problems. Since most image processing algorithms are designed for L^2-normalized transform, and since the studied NMR are L^1-normalized, we first define the renormalized representations as follows:

$$\tilde{\mathcal{M}}v = (2^{J-1}v^0, 2^{J-1}d^0, ..., 2^{p-1}d^{J-p}, ..., d^{J-1}). \tag{29}$$

A. Comparison of NMRs on Synthetic Images: Decay of the Detail Coefficients

In this section, we perform several numerical tests to illustrate the faster decay of the renormalized detail coefficients, when one uses a nonlinear, rather than a linear, prediction. Our goal is to show how the edge model impacts the decay of the detail coefficients. For that purpose, we consider 3 different synthetic images (see Figure 6 first row), all of size 128×128, that is $J = 2^7$. We consider the linear predictions corresponding either to the Haar basis (Haar prediction) or to the use of polynomial p_k on each cell (Linear prediction) and the three different types of nonlinear predictions SR_1, SR_2 and SR_3 described above. We depict, on the second row of Figure 6, the decay of the amplitude of the renormalized

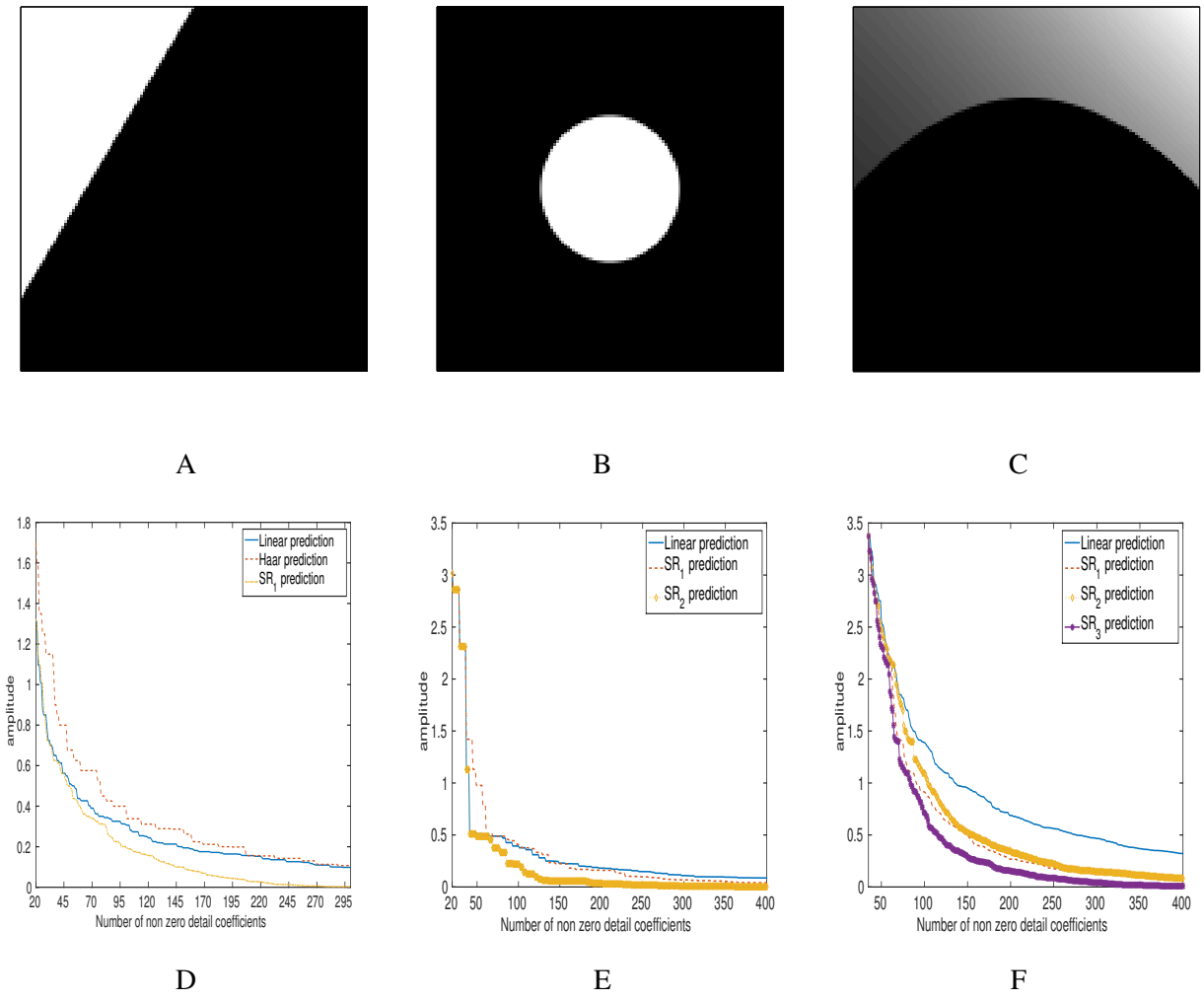


Fig. 6. A: 128×128 image of a straight-line edge (i.e. averaged over cells of size $1/128 \times 1/128$); B: 128×128 image of a circle (i.e. averaged over cells of size $1/128 \times 1/128$); C: 128×128 image of a piecewise biquadratic model; D: Decay of the renormalized detail coefficients using different linear and nonlinear methods on image A (nonlinear prediction is only performed at the three finest levels of decomposition); E: idem as D but for image B; F: idem as D but for image C.

detail coefficients (nonlinear prediction is performed only at the first 3 levels of decomposition for SR_1, SR_2 and SR_3, linear prediction being used at other levels). For the image of Figure 6 A, SR_1, SR_2 and SR_3 are the same, so we only compare linear predictions with SR_1. What is remarkable on Figure 6 D is that the linear and SR_1 predictions coincide for the first 20 or so coefficients: these coefficients are associated with levels smaller or equal to $J-4$. Then the decay of the detail coefficients for SR_1 is much faster, and is controlled by the behavior of the detail coefficients at levels larger or equal to $J-3$. It is important to note here that the nonlinear prediction does not generate detail coefficients, at level

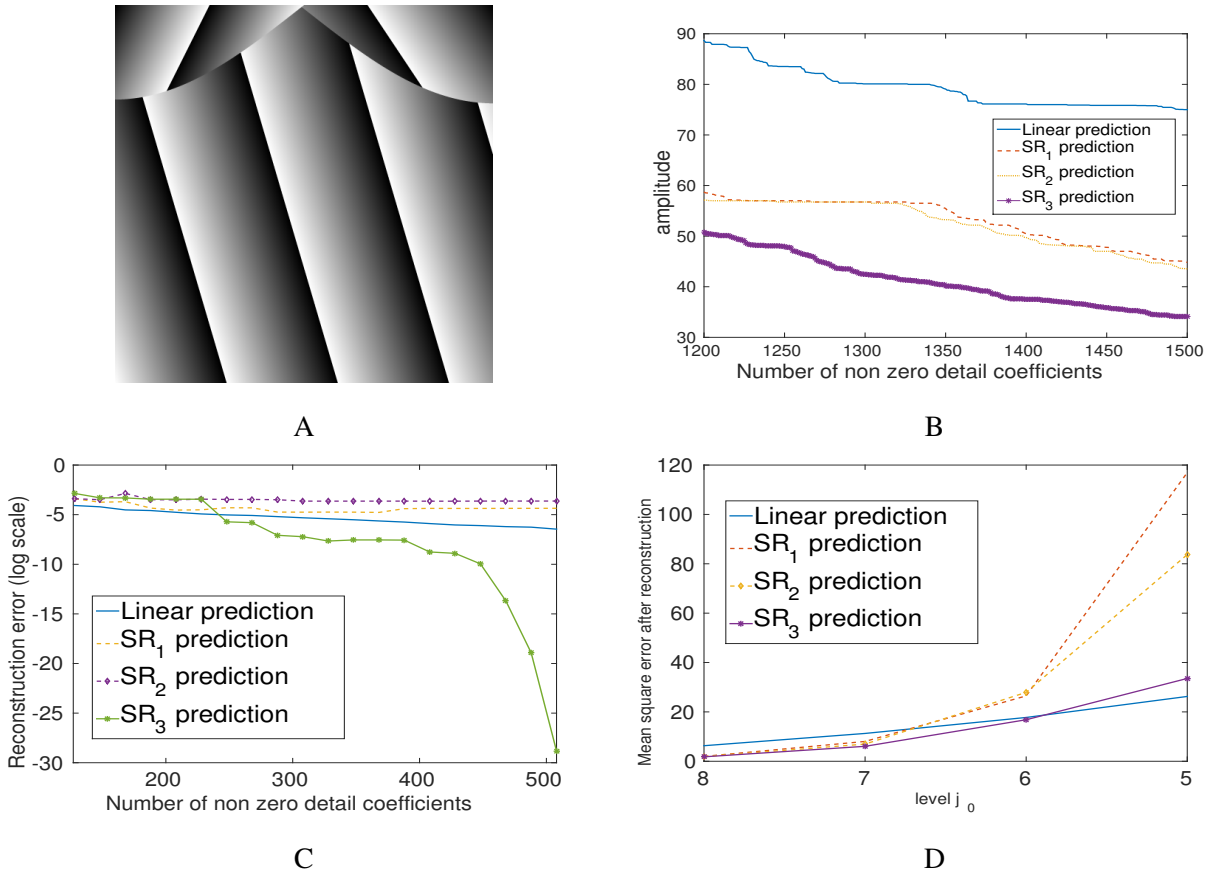


Fig. 7. A: 512×512 geometric image containing linear, curvilinear edges and T-junctions; B: Decay of detail coefficients using linear and nonlinear methods computed on the finest three levels of decomposition for image A; C: Nonlinear reconstruction error corresponding to image A; D: Mean square reconstruction error associated with super-resolution procedure using only the information available at level j_0 for image A.

larger than $J-3$, having an amplitude larger than that of the detail coefficients obtained at level smaller than $J-4$.

Then, for the image of Figure 6 B, SR_2 and SR_3 are the same since $p_{a,k}$ and $p_{b,k}$ used by SR_3 reduce to constants, so we only depict the results associated with SR_2. For such an image, the behavior of the different decompositions, depicted on Figure 6 E, are the same as shown before: the first detail coefficients of the transforms are the same (they are associated with levels where the decomposition is linear) and then the decay is faster with SR_2 which is the only method to take into account the curvature of the edge. The bad behavior of SR_1 on this image indicates that it is worth taking into account the curvature of the edge when possible: the straight-line model on S-cells is too crude an approximation of the edge at the scales of study (some of which are thus larger than the critical scale h_c mentioned in

Section IV).

Finally, for a piecewise biquadratic signal as the one in Figure 6 C, notice again that the largest detail coefficients are the same for each type of decomposition. Then, as far as the remaining coefficients are concerned, to consider a biquadratic model on each side of the edge significantly improves the prediction, as SR_3 behaves much better than the other methods. Another illustration of the decay of the detail coefficients is given in Figure 7 B for the more sophisticated image of Figure 7 A and for which the advantage of using SR_3 to predict is again significant.

B. Comparison of NMRs on Synthetic Images: Nonlinear Approximation

Let us define $\tilde{v}^{N,J}$ as the approximation of v^J obtained by keeping the N largest detail coefficients, and then $\varepsilon_N := \|\tilde{v}^{N,J} - v^J\|_2$. The L^2 error obtained by keeping the N largest detail coefficients of an L^2-normalized wavelet transform satisfies $\|\varepsilon_N\|_2 \leq CN^{-s}$, and one obtains $s = 1$ for bounded variation images [10]. Our goal here is to illustrate that a faster decay of the normalized detail coefficients entails a faster decay of the approximation error ε_N. As before, for the SR_1, SR_2 and SR_3 methods, we consider that the prediction is nonlinear only at the three finest levels of decomposition and linear otherwise. We display ε_N corresponding to the different multiscale decompositions, in Figure 7 C, for the image of Figure 7 A. Notice that the results are in accordance with the decay of the detail coefficients observed on Figure 7 B. Indeed, since the largest detail coefficients are related to the coarsest scales, the algorithm first integrates the detail coefficients belonging to these scales, thus it progressively reconstructs the set of S-cells while adding new detail coefficients. This point is very important since it satisfactorily circumvents the problem of the absence of synchronization, which can be stated as: there is no reason for S-cells to be the same at the encoding and decoding steps. Since the proposed NMRs do not create large coefficients at the fine scale, S-cells are properly recovered by adding progressively new large amplitude detail coefficients, leading to a faster decay of the reconstruction error. Moreover, this study shows that the edge parameters are accurately estimated even when all the detail coefficients are not available. Finally, we also mention that the reconstruction error drastically decreases with all the other methods, but for a much larger N.

C. Application of NMRs to Super-Resolution Image Reconstruction

Super-resolution is a class of techniques used to enhance the resolution of an imaging system. In principle, the resolution is limited by the diffraction limit, which, in short, means that the image is

lowpass filtered at acquisition. This has the consequence that some of the high frequencies are not resolved.

When transposed to our context, the diffraction limit means the information on the image is not available at levels larger than some j_0. The challenge is thus to reconstruct an approximation of the original image v^J, as faithful as possible, by using only the coarse approximation v^{j_0}, $j_0 < J$. The literature on super-resolution techniques relative to image processing is huge and our goal is not to compare to existing techniques, but to show the relevance of using nonlinear prediction techniques in that context. For a review on super-resolution image reconstruction, we refer the reader to [11], and references therein.

The fact that nonlinear prediction operators lead to very few significant high frequency detail coefficients should naturally result in good behavior with regards to super-resolution techniques. Indeed, the knowledge of the high frequency detail coefficients should matter much less than when a linear prediction is used. We check this property by computing the mean square error of linear and nonlinear reconstruction processes, i.e. $\frac{1}{2^{2J}} \sum_k (\tilde{v}_k^{j_0,J} - v_k^J)^2$, where $\tilde{v}^{j_0,J}$ is the reconstructed image at level J using only the information available at level j_0. The results are depicted in Figure 7 D for the image of Figure 7 A. Since the image studied is 512×512, $J = 9$ and we reconstruct the image using v^{j_0}, $j_0 < J$. The use of nonlinear prediction techniques appears to improve super-resolution performance for level $j_0 \geq 7$. Furthermore, by considering a more accurate edge model, i.e. SR_3, one lowers significantly the reconstruction error compared with other studied linear and nonlinear methods, provided $j_0 \geq 6$.

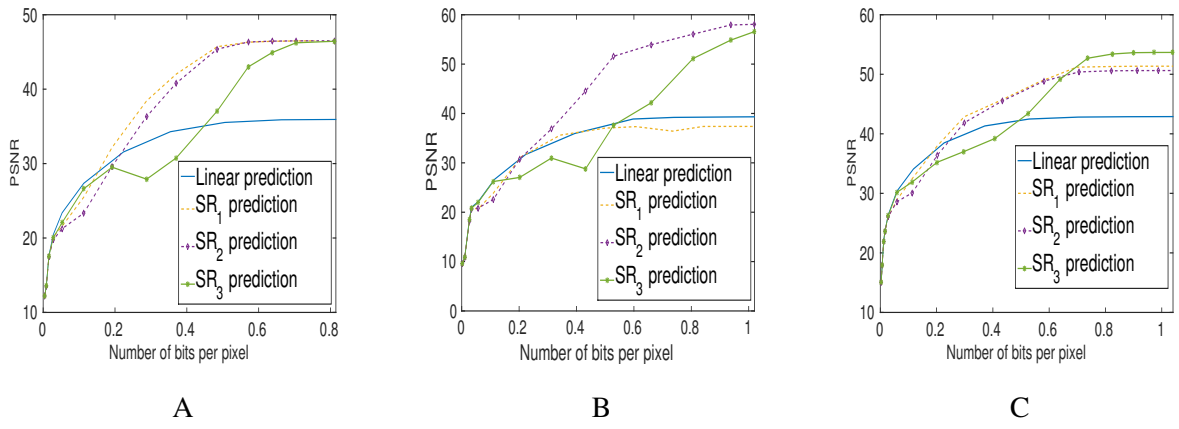


Fig. 8. A: Compression results with EZW on image of Figure 6 A; B: Idem for image of Figure 6 B; C: Idem for image of Figure 6 C.

D. Image Compression

We have just shown that the proposed NMRs are well behaved with regards to the nonlinear approximation and super-resolution. As previously mentioned, this can be explained by the fact that coarse levels of decomposition contain the information on the location of S-cells, (see super-resolution performance), and since nonlinear approximation integrates first the coarse scales, S-cells are mostly recovered before the detail coefficients computed nonlinearly are added. Now, if the interest is in image compression, the problem being addressed is different because, not only is partial information available at the decoding step, but also the former is quantized. In that context, it is natural to consider if our algorithm can detect S-cells from this quantized and partial information and, if so, if the edge parameters can be estimated accurately.

Since the proposed NMRs are associated with a quadtree structure, they can be compressed using one of the most efficient compression algorithm, the so-called EZW (Embedded Zero-Tree) algorithm [1]. First, note that the renormalized NMR $\tilde{\mathcal{M}}v$ can be written in a matrix form, following the same display of detail coefficients as used by orthogonal wavelet representation. In what follows, we call this matrix V^J . The EZW algorithm, developed for wavelet transforms, (supposed to be L^2-normalized, therefore we use $\tilde{\mathcal{M}}v$), exploits the quadtree structure generated by the wavelet decomposition. It is based on progressive encoding: the data is compressed through multiple passes with increasing accuracy. The rational behind the EZW encoder is to build zero-tree structures from the quadtree, based on the observation that the wavelet coefficients decrease as j increases.

In practice, we set the initial threshold for compression to $T_0 = 2^{\lfloor \log_2(\max |V^J|) \rfloor}$. The encoder then scans matrix V^J using the Morton scan [12], compares each scanned coefficient with threshold T_0, and issues 'p', 'n', 'z' or 't' as outputs. If the magnitude of a scanned coefficient is larger than T_0, the output is 'p', if the coefficient is positive, and 'n' otherwise. Else the algorithm constructs a tree with the considered element as root. If it is a zero-tree, i.e. the values at the nodes are all smaller or equal to the threshold, the output is 't', and 'z' (isolated zero) otherwise. The EZW encoder assumes that there is a very high probability that all of the coefficients in a quadtree are smaller than a certain threshold if the root is smaller than this threshold. To encode elements 'p' or 'n', one puts each of them in a so-called 'subordinate list' associated with $\frac{3T_0}{2}$ (resp. $-\frac{3T_0}{2}$), when the associated coefficient is positive (resp. negative) with magnitude larger than $\frac{3T_0}{2}$. Once all the elements are scanned, the threshold is set to $T_0/2$, and the algorithm starts a new pass, and finally stops after a number of passes corresponding to a predefined minimal value for the threshold. This procedure can be viewed as a bit-plane coding algorithm,

the level of quantization depending on the number of passes. The encoding algorithm described above thus computes a sequence of quantized coefficients:

$$\overline{\tilde{\mathcal{M}}v} = (2^{J-1}\bar{v}^0, 2^{J-1}\bar{d}^0, ..., 2^{p-1}\overline{d^{J-p}}, ..., \overline{d^{J-1}}). \tag{30}$$

To recover an approximation of the original image from the quantized decomposition, the inverse operator, i.e. decoding step, is written as follows:

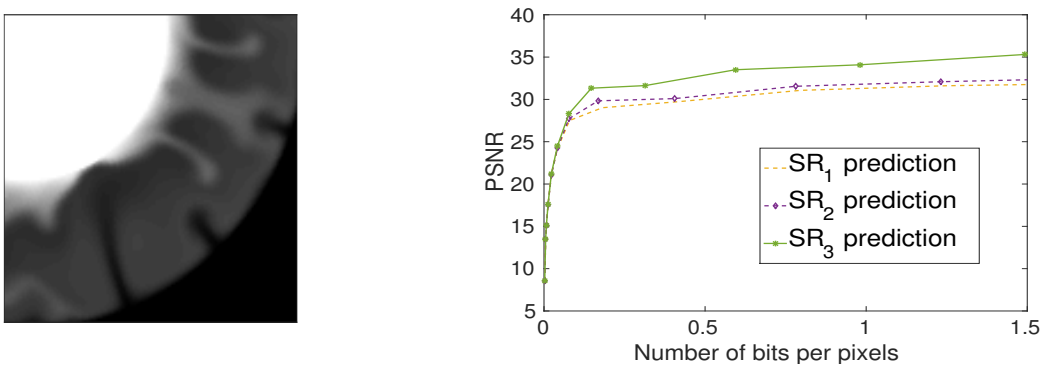
$$\tilde{v}^J = \tilde{\mathcal{M}}^{-1}(2^{J-1}\bar{v}^0, 2^{J-1}\bar{d}^0, ..., 2^{p-1}\overline{d^{J-p}}, ..., \overline{d^{J-1}}).$$

It is worth noting that due to quantization and compression, the prediction operator at the encoding and decoding steps may not be the same, which, as already mentioned, is known as the absence of synchronization between the encoder and the decoder. To avoid this problem, a possible choice would be to memorize the location of S-cells at each level during the encoding step, and to reuse them in the decoding step [13]. However, this would require the allocation of too many bits for that operation, which would result in a deterioration of the compression results. In contrast, and in a similar fashion to what has been shown previously (in the application of NMRs to super-resolution and nonlinear approximation), we expect that the quantized decomposition will enable the recovery of the S-cells automatically, and the estimation of the edge parameters, provided the number of bits per pixel (bpp) is sufficiently high. In this regard, the compression results using EZW algorithm expressed in terms of the PNSR with respect to the bpp after compression is displayed in Figure 8. For the sake of consistency, we recall that the PSNR corresponds to the formula:

$$PSNR \;=\; 10\log_{10}\left(\frac{Q^2}{\frac{1}{2^{2J}}\sum_k(\tilde{v}_k^J - v_k^J)^2}\right),$$

where Q is the maximum of the amplitude of the image coefficients. It is important to note here that, for the different NMRs, only the first level of decomposition is computed nonlinearly.

Looking at the results depicted in Figure 8, we see that, when the image only contains a straight-line edge, SR_1 and SR_2 behave similarly which means that SR_2 adapts to non curved edges (see Figure 8 A). Furthermore, when the edge is curved, better compression results are obtained by taking into account the curvature at the encoding step (compare SR_1 and SR_2 in Figure 8 B). In these first two cases, notice also that, when one uses a more sophisticated edge model such as SR_3, the number of bpp to recover the edge parameters is more important than with SR_1 and SR_2. Finally, we notice, in Figure 8 C, that SR_3 behaves better than SR_1 and SR_2, when the image is a piecewise biquadratic polynomial and provided enough bits are transmitted. Finally an illustration of the compression performance of the



A B

Fig. 9. A: image of a convection flow, the heat source is at the top left; B: compression results using the different NMRs

proposed NMRs on the image of a convection flow is shown, (see Figure 9 A), for which we note, in Figure 9 B, better behavior by SR_3 in comparison to SR_1 and SR_2 .

VII. Conclusion

In this paper, we have derived a new type of nonlinear multiscale representation based on a nonlinear prediction operator in a cell-average framework. As the structure follows the same quadtree structure as the orthogonal wavelet transform, a compression algorithm such as EZW can be applied to the nonlinear multiscale representation. We noted significant improvement in terms of compression performance when compared with linear multiscale representations. Another application of the proposed nonlinear representation is super-resolution, for which we have shown that accurate reconstruction of piecewise regular images can be achieved by using an approximation of the image at a coarse resolution level.

References

[1] Shapiro J., "Embedded image coding using zerotrees of wavelet coefficients," *IEEE Transactions on Signal Processing*, vol. 12, pp. 3445–3462, 1993.

[2] Taubman D., "High performance scalable image compression with ebcot," *IEEE Transactions on Image Processing*, vol. 9, pp. 1158–1170, 2000.

[3] E. Candes and D.L. Donoho, "New tight frames of curvelets and optimal representations of objects with piecewise c^2 singularities," *Comm. on Pure and Appl. Mathematics*, vol. 57, pp. 219–266, 2002.

[4] M.N. Do and M. Vetterli, "The contourlet transform: an efficient directional multiresolution image representation," *IEEE Trans. Imag. Proc.*, vol. 14, pp. 2091–2016, 2005.

[5] E. Le Pennec and S. Mallat, "Sparse geometric image representations with bandelets," *IEEE Transactions on Image Processing*, vol. 14, pp. 423–438, 2005.

[6] R. Baraniuk, H. Choi, J. Romberg, and M. Wakin, "Wavelet-domain approximation and compression of piecewise smooth images," *IEEE Trans. Im. Proc.*, vol. 15, pp. 1071–1087, 2006.

[7] F. Arandiga, A. Cohen, R. Donat, N Dyn, and Mateï B., "Approximation of piecewise smooth images by edge-adapted techniques," *Appl. Comput. Harmon. Anal.*, vol. 24, pp. 225–250, 2008.

[8] A. Harten, "Discrete multiresolution analysis and generalized wavelets," *J. of App. Num. Math.*, vol. 12, pp. 153–193, 1993.

[9] A. Harten, B. Enquist, S. Osher, and S. Chakravarthy, "Uniformly high order accurate essentially non-oscillatory schemes iii," *J. Comp. Phys.*, vol. 71, pp. 231–303, 1987.

[10] A. Cohen, R. DeVore, P. Petrushev, and H. Xu, "Non-linear approximation and the space BV($\mathbb{R}^2$)," *American Journal of Mathematics*, vol. 121, pp. 587–628, 1999.

[11] S. C. Park, M. K. Park, and G. K. Moon, "Super-resolution image reconstruction: A technical overview," *IEEE Signal Processing Magazine*, vol. 20, no. 3, pp. 21–36, 2003.

[12] V.R. Algazi and R.R.: Estes, "Analysis based coding of image transform and subband coefficients," in *Proceedings of the SPIE*, 1995, 2564, pp. 11–21.

[13] V. Chappelier and C. Guillemot, "Oriented wavelet transform for image compression and denoising," *IEEE Trans. Image Process.*, vol. 15, pp. 2892–2903, 2006.

Sylvain Meignen received his Ph.D degree in applied mathematics in 2001, and its "habilitation à diriger des recherches" in 2011, both from the university of Grenoble, France. Since, 2002, he has been an associate professor at Grenoble institute of technology. His research interests include nonlinear multiscale image and signal processing, time-frequency analysis (empirical mode decomposition,synchrosuqeezing) and approximation theory. In 2010, he was a visitor at the GIPSA-Lab Grenoble and at the IDCOM Laboratory of the university of Edinburgh, U.K., in 2011.

Basarab Mateï was born in Bucharest, Romania. He graduated from the university of Bucharest, faculty of mathematics, in 1996. He received the Ph.D.degree in applied mathematics from the Paris VI university, France, in 2002. He is an associate professor in the machine learning and applications team of the computer science laboratory of Paris-north university (LIPN). His research interests include wavelets, adaptative representations, irregular sampling, quasicrystals, tillings, compressed sensing and machine learning.