Supervised classification for textured images

Jean-François Aujol, Gilles Aubert, Laure Blanc-Féraud

INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Supervised classification for textured images

Jean-François Aujol — Gilles Aubert — Laure Blanc-Féraud

N° 4640

November 2002

THÈME 3

ISSN 0249-6399 ISRN INRIA/RR--4640--FR+ENG

Supervised classification for textured images

Jean-François Aujol [*] [†] , Gilles Aubert [‡] [§] , Laure Blanc-Féraud [¶]

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

Rapport de recherche n° 4640 — November 2002 — 52 pages

Abstract: In this report, we present a supervised classification model based on a variational approach. This model is specifically devoted to textured images. We want to get an optimal partition of an image which is composed of textures separated by regular interfaces. To reach this goal, we represent the regions defined by the classes as well as their interfaces by level set functions. We define a functional on these level sets whose minimizers define an optimal partition. In particular, this functional owns a data term specific to textures. We use a packet wavelet transform to analyze the textures, these ones being characterized by their energy distribution in each sub-band of the decomposition. The partial differential equations (PDE) related to the minimization of the functional are embeded in a dynamical scheme. Given an initial interface set (zero level set), the different terms of the PDE's govern the motion of interfaces such that, at convergence, we get an optimal partition as defined above. Each interface is guided by external forces (regularity of the interface),

[*] également membre du laboratoire J.A Dieudonné, Université de Nice-Sophia-Antipolis
[†] jfaujol@sophia.inria.fr
[‡] Laboratoire J. A. Dieudonné, UMR CNRS 6621, Université de Nice-Sophia-Antipolis, Parc Valrose, 06108 Nice Cedex 2, France
[§] gaubert@math.unice.fr
[¶] blancf@sophia.inria.fr

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

and internal ones (data term and partition constraints). We have conducted several experiments on both synthetic and real images.

Key-words: Texture, classification, variational approach, level set, active regions, active contours, multiphase, wavelets, PDE.

Classification supervisée d'images texturées

Résumé : Dans ce rapport, nous présentons un modèle de classification supervisée basé sur une approche variationnelle. Ce modèle s'applique spécifiquement aux images texturées. Nous souhaitons obtenir une partition optimale de l'image constituée de textures séparées par des interfaces régulières. Pour cela, nous représentons les régions définies par les classes ainsi que leurs interfaces par des fonctions d'ensemble de niveaux. Nous définissons une fonctionnelle sur ces ensembles de niveaux dont le minimum est une partition optimale. Cette fonctionnelle comporte en particulier un terme d'attache aux données spécifique aux textures. Nous utilisons une transformée en paquets d'ondelettes pour analyser les textures, ces dernières étant caractérisées par la distribution de leur énergie dans chaque sous-bande de la décompositon. Les équations aux dérivées partielles (EDP) relatives à la minimisation de la fonctionnelle sont couplées et plongées dans un schéma dynamique. En fixant un ensemble de niveaux initial, les différents termes des EDP guident l'évolution des interfaces (ensemble de niveau zéro) vers les frontières de la partion optimale, par le biais de forces externes (régularité de l'interface) et internes (attache aux données et contraintes partition). Nous avons effectué des tests sur des images synthétiques et sur des images réelles.

Mots-clés : Texture, classification, approche variationnelle, lignes de niveaux, régions actives, contours actifs, multiphases, ondelettes, EDP.

Contents

1. Introduction

1.1 Overview

We propose an algorithm for supervised classification of textured images. This algorithm is based on a variational approach.

Image classification consists of assigning a label to each pixel of an observed image. This label indicates to which class belongs a pixel. Classification can be seen as a partition problem. This is one of the basic problem in image processing. This concerns many applications as for instance landscape management in teledetection.
The classification problem is closely related to the segmentation one, in the sense that the goal is to get a partition of the image composed of homogeneous regions. In the classification problem, each partition represents a class.
Many classification models have been developed, especially from regions growing algorithms [25, 32, 53], or by a stochastic approach [5, 7, 20, 21, 30, 39, 27], and most recently by a variational approach [4, 47, 48, 49, 44, 52].
The approach which is used here is inspired from [48], and is based on active contours ([4, 8, 35, 9, 28, 3, 51, 46, 43, 55, 56]). The partition we seek is a minimizer of a functional. We compute this minimizer by solving the associated PDE's system. These PDE's guide the interfaces (zero level sets) towards the boundary of the optimal partition thanks to external (regularity of the interface) and internal (data term and partition) forces.

The classification is supervised, since we assume that we know the parameters of the different classes of the partition in the image.

1.2 Position of our model in the existing literature

There exist many classification models for textured images in the literature [54]. Two main directions emerge:

1. Traditional statistical models, which assume that the statistic of each texture are stationnary, especially in the case of Markov random field approaches (see [5, 20, 21, 39, 7, 27, 30]).

2. Models based on the filtering theory, especially Gabor filters cf [23, 44]) or wavelets (cf [53, 16, 6, 24, 19, 50, 10]).

We have chosen to use the wavelets approach which give an excellent way to decompose a signal in different sub-bands in which it is easier to caracterized it.

The use of wavelet to analyze textures is not new. There exists mainly two approaches:

1. Models which assume the independence of the distribution of the wavelets coefficients (see [37, 53, 19, 50, 22]).

2. Models which assume a spatial dependence (interactions between the pixels) and through the scales (interactions between the wavelet coefficients through the scales) (see [16, 17, 34, 10, 12, 33, 14, 13, 15]). These models, based on hidden Markov random fields, are nevertheless heavy to implement, and it does not seem that the benefits are valuable.

We have chosen to use a model of the type proposed by Unser [53]. We consider that a texture is caracterized by the energy of its wavelet coefficients in each sub-bands of the decomposition. Our choice is confirmed by the quality of the results we get.

Our model is new since, as far as we know, it is the only one which uses simultaneously an active contour approach guided by wavelets.

In the existing literature, the closest model is the one by Paragios and Deriche proposed in [44]. But it differs on many points:

1. In [44], the authors use a model presented in [56], and caracterize textures with Gabor filters. Historically, Gabor filters have been introduced before wavelets bases (see [41]). But the storage cost is very important, and there exist functions in L^2 which cannot be decomposed into a convergent Gabor wavelet expansion. The construction of wavelet bases by I. Daubechies (see [18]) had overcome these defaults. Moreover, Gabor

filters have a more important cost, especially regarding to the computation of the low freqencies components. Gabor filters are not orthogonal, which may introduce a significant correlation between textures. Lastly, these transformations are in general not reversible, and this limit their application to texture synthesis (see [53]).

All these reasons make the use of wavelets more appropriate.

2. The introduction of the active contours in our method is natural: indeed, it comes directly from the functional we seek to minimize. Moreover, in [44], the energy does not have any coupling terms.They are added in the Euler-Lagrange equations.

3. The model in [44] is less general than the one we propose: it only deals with the case of an image having textures dispatched on a textured background. In particular, [44] or [45] show no segmentation or classification result of an image having triple junctions (or more).

4. Let us notice that contrary to the model developed in [48], we do not have a stopping function on the contours: this function was based on the gradient of the image, and as we are interested here with textured images, such an approach is no longer possible.

   In [44], the authors also use a stopping term on the contours. But to implement it, they need the fact that their model deal with textures components dispatched on a background.

   In order to keep our classification model the more general as possible, we have decided not to use such a detector of contours.

   Moreover, it can be checked experimentally that contours do stop without the use of such a stopping function.

1.3 Principle

Our model is inspired by the work of C. Samson *et al.* who have developed a supervised classification algorithm for non textured images in [48].

The number of classes K present in the image is supposed to be known, as well as the caracteristics of each class. If $1 \leq k \leq K$, we will denote by Cl_k the corresponding class.
The aim is to get an optimal partition of the image (we consider the classification problem as a partition problem). To do so, we are going to minimize a functional with three terms:

(i) A partition term.

(ii) A term which control the regularity of the contours of the classes.

(iii) A data term.

We use a variational level set approach. The domain of the image, Ω, is the union of disjoints sets Ω_k. Each Ω_k is caracterized by a level set function Φ_k.

The paper is organized as follows. Section 2 presents the framework of our classification algorithm. We need to construct a data term specific to textures. That is why in section 3 we make a few reminds on wavelets. We then caracterize textures through their wavelet expansion in section 4. We deduce our data term from this study in section 5, and we present the algorithm we use to minimize it. We then finish by giving some numerical results on both synthetic and real textured images in section 6.

2. Classification

Our approach is inspired from the work [48].

2.1 Partition, level sets approach

The image is considered as a function $u_0 : \Omega \mapsto \mathbb{R}$ (where Ω is an open subset of $\mathbb{R}^2$)

We will denote $\Omega_k = \{x \in \Omega \,/\, \text{x belongs to the class k}\}$. The collection of open sets $\{\Omega_k\}$ form a partition of Ω if and only if

$$\Omega = \bigcup_k \Omega_k \bigcup_k \Gamma_k \text{ , and if } k \neq l \ \Omega_k \bigcap \Omega_l = \emptyset$$

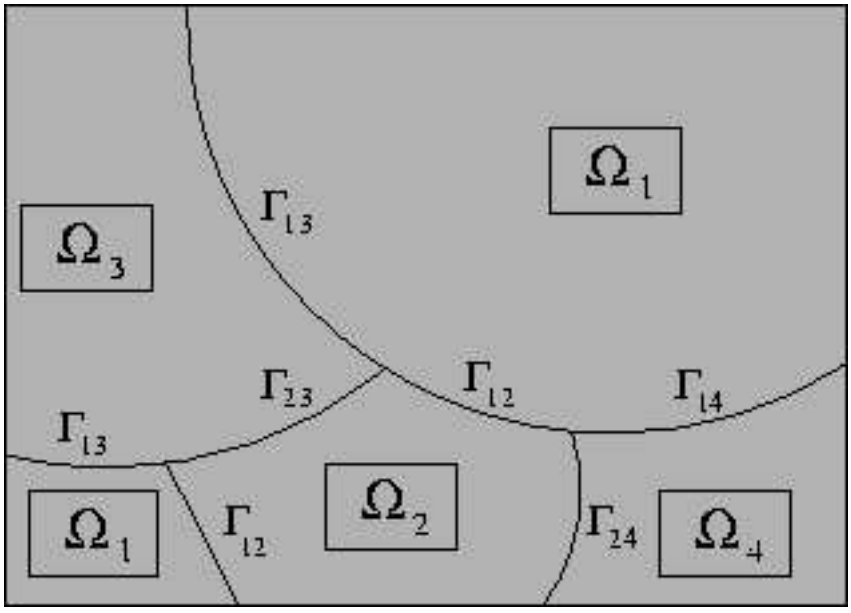

Figure 1: Classification seen as a partition problem

We will denote $\Gamma_k = \partial\Omega_k \bigcap \Omega$ the boundary of Ω_k (except points belonging also to $\partial\Omega$), and $\Gamma_{kl} = \Gamma_{lk} = \Gamma_k \bigcap \Gamma_l \bigcap \Omega$ the interface between Ω_k and Ω_l (see figure 1).

In order to get a functional formulation rather than a set formulation, we will suppose that for each Ω_k there exists a lipschitz function $\Phi_k : \Omega \to \mathbb{R}$ such that:

$$\begin{cases} \Phi_k(x) > 0 & \text{if } x \in \Omega_k \\ \Phi_k(x) = 0 & \text{if } x \in \Gamma_k \\ \Phi_k(x) < 0 & \text{otherwise} \end{cases}$$

Ω_k is thus completely determined by Φ_k.

2.2 Regularization

In our equations, there will appear some Dirac and Heaviside distributions δ and H. In order that all the expressions we write have a mathematical

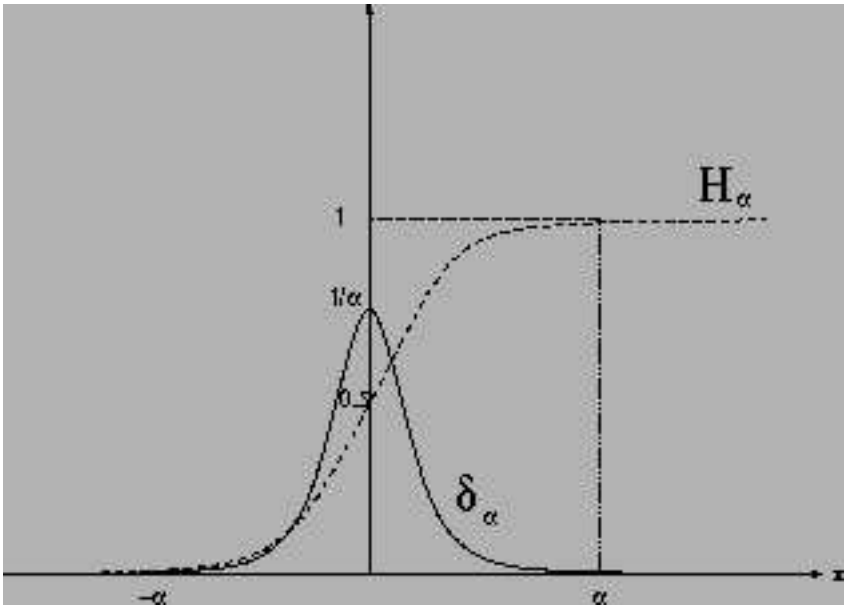

Figure 2: Approximations δ_α and H_α of the Dirac and Heaviside distributions

meaning, we will use the classical regular approximations of these distributions (see figure 2):

$$\delta_\alpha = \begin{cases} \frac{1}{2\alpha}\left(1+\cos\frac{\pi s}{\alpha}\right) & \text{if } |s| \le \alpha \\ 0 & \text{if } |s| < \alpha \end{cases}$$

$$H_\alpha = \begin{cases} \frac{1}{2}\left(1+\frac{s}{\alpha}+\frac{1}{\pi}\sin\frac{\pi s}{\alpha}\right) & \text{si } |s| \le \alpha \\ 1 & \text{if } s > \alpha \\ 0 & \text{if } s < \alpha \end{cases}$$

When $\alpha \to 0$, we have $\delta_\alpha \to \delta$ and $H_\alpha \to H$ (in the distributional sens).

Figure 3 shows how the regions are defined by these distributions and the level sets.

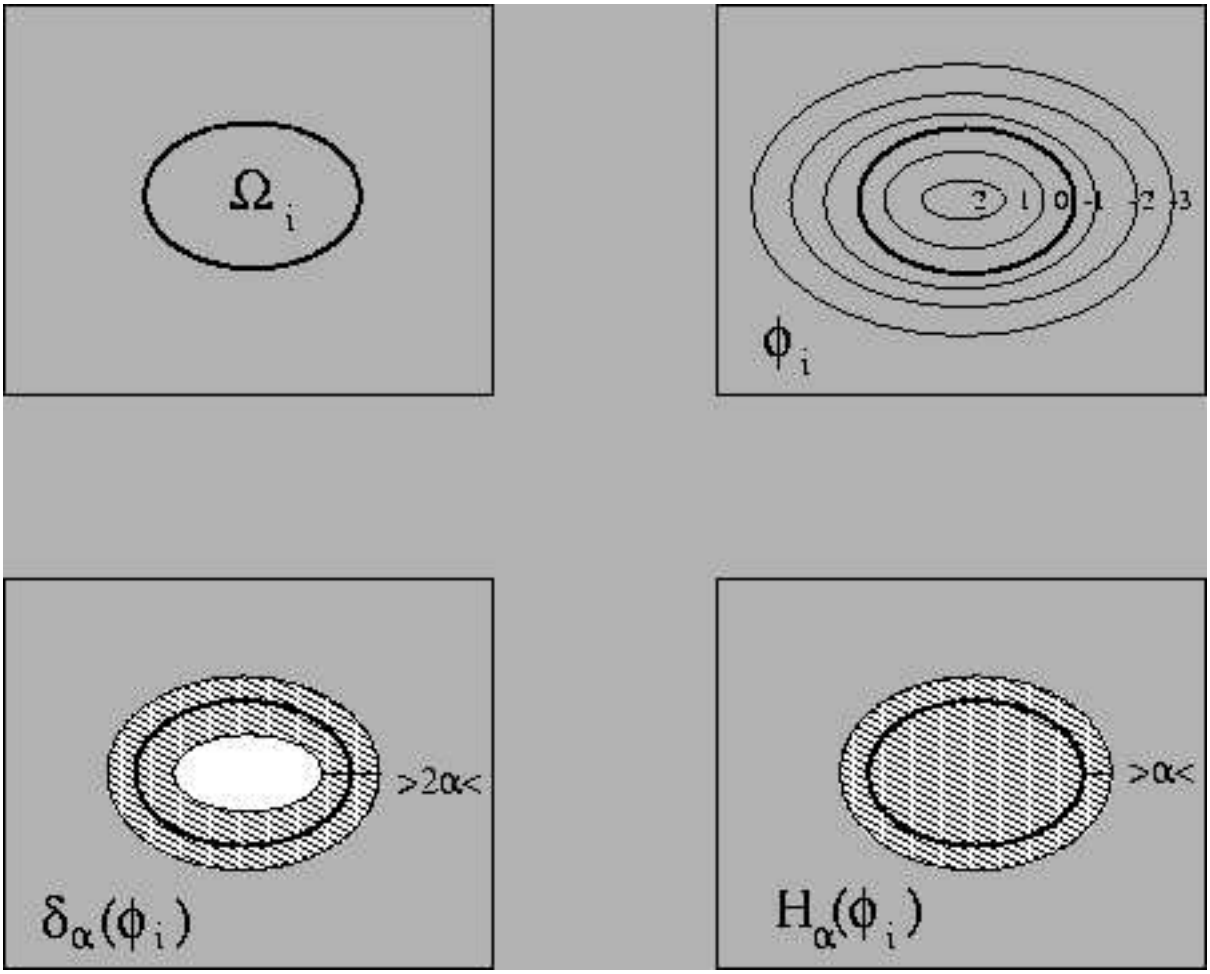

Figure 3: Definitions of the regions and of their level sets

2.3 Functional

Our functional will have three terms:

1.
$$F_\alpha^A(\Phi_1,\ldots,\Phi_K) = \lambda \int_\Omega \left(\sum_{k=1}^K H_\alpha(\Phi_i) - 1 \right)^2 \tag{2.1}$$

This energy term ensures that the result of the functional minimization is indeed a partition of the image. It penalizes the pixels which are unclassified, as well as the ones classified in at least two regions simultaneously. In fact, to get a partition of Ω, we would need to have $\sum_{k=1}^K H(\Phi_k(x)) = 1\ ,\ \forall x \in \Omega$. We rather minimize the quadratic error, which is easier.

2.
$$F^B(\Phi_1,\ldots,\Phi_K) = \sum_{k=1}^K \gamma_k |\Gamma_k| \tag{2.2}$$

This term penalizes the contours length, which prevent from having too irregulars contours. By using the co-area formula, it is possible to show that (the proof is given in [48]):

$$\lim_{\alpha\to 0} \int_\Omega \delta_\alpha(\Phi_k)|\nabla\Phi_k| dx = |\Gamma_k| \tag{2.3}$$

In practice, we seek to minimize:

$$F_\alpha^B(\Phi_1,\ldots,\Phi_K) = \sum_{k=1}^K \gamma_k \int_\Omega \delta_\alpha(\Phi_k)|\nabla\Phi_k| \tag{2.4}$$

3.
$$F^C(\Phi_1,\ldots,\Phi_K) \tag{2.5}$$

This last term stands for the data term, and we will get it from our textures modelization and the maximum likelihood principle.

Complete functional: The functional we want to minimize is the sum of the three previous terms:

$$F(\Phi_1,\dots,\Phi_K) = F^A(\Phi_1,\dots,\Phi_K) + F^B(\Phi_1,\dots,\Phi_K) + F^C(\Phi_1,\dots,\Phi_K) \quad (2.6)$$

Remark: Let us notice that, contrary to the model proposed in [48], we do not have a stopping function on contours: the function which was used in [48] was based on the gradient of the image, and as we are interested here in textured images, such an approach is no longer possible. In general, in the case of a textured image, high gradients do not occur only at the interface between two textures, and two different textures may have in average the same gray level.

Before constructing our data term $F^C(\Phi_1,\dots,\Phi_K)$, we need to make a few reminds on wavelets.

3. About wavelets

We recall here some fundamental properties of wavelets. We refer the interested reader to [38, 41, 18, 37, 36].

3.1 1-D case

Définition 3.1. A mother wavelet $\psi(t)$ is a function which satisfies the following properties:

1. ψ is a regular function (with $r-1$ continuous derivatives and a bounded derivative of order r).
2. ψ and its derivatives of order less than r have fast decrease at infinity.
3. The family $\psi_{j,k}(t)$ defined by $\psi_{j,k}(t) = 2^{\frac{j}{2}}\psi(2^j t - k)$, $j,k \in \mathbb{Z}$, is an orthonormal basis of $L^2(\mathbb{R})$.

Définition 3.2. A multi-resolution analysis of $L^2(\mathbb{R})$ is a sequence V_j, $j \in \mathbb{Z}$ of subspaces of $L^2(\mathbb{R})$ with the following properties:

1. $$\bigcap_j V_j = \varnothing$$

2. $$V_j \subset V_{j+1}$$

3. $$\bigcup_j V_j = L^2$$

4. $f(t) \in V_j$ if and only if $f(2t) \in V_{j+1}$

5. There exists a regular function $\phi(t)$ (with the same regularity as the mother wavelet) with compact support such that the familly $\phi(t-k)$, $k \in \mathbb{Z}$ is an orthonormal basis of V_0 (for the scalar product of $L^2(\mathbb{R})$). Such a function ϕ is called a scaling function.

It is immediate to check that:

$$\text{The family } \phi_{j,k}(t) \text{ defined by } \phi_{j,k}(t) = 2^{\frac{j}{2}}\phi(2^j t - k),\ k \in \mathbb{Z}, \text{ is an orthonormal basis of } V_j. \tag{3.1}$$

We now give the link between wavelet basis and multi-resolution analysis (we just give main ideas, all details can be found in the works of S.G. Mallat [37, 38, 36]):

Assume that we have a multi-resolution analysis, and let us build an orthonormal wavelet basis W_0 the orthogonal complement of V_0 in V_1. We build the mother wavelet by imposing the following condition:

$$\psi(t-k),\ k \in \mathbb{Z}, \text{ is an orthonormal basis of } W_0. \tag{3.2}$$

This property does not define a unique mother wavelet. One also have to impose regularity and localization conditions on ψ to get uniqueness.

By change of scale, one gets:

The familly $\psi_{j,k}(t)$ defined by $\psi_{j,k}(t) = 2^{\frac{j}{2}}\psi(2^j t - k)$, $k \in \mathbb{Z}$, is an orthonormal wavelet basis of W_j orthogonal complement of V_j in V_{j+1}. (3.3)

Remark: For $j \in \mathbb{Z}$, since W_j is the orthogonal complement of V_j in V_{j+1}, we have:

$$V_j \oplus W_j = V_{j+1} \tag{3.4}$$

and since the V_j form a multi-resolution analysis,

$$V_J = \oplus_{j=-\infty}^{J-1} W_j \tag{3.5}$$

It is then clear that:

The family $\psi_{j,k}(t)$, $j,k \in \mathbb{Z}$ is an orthonormal basis of $L^2(\mathbb{R})$. (3.6)

For each function z in $L^2(\mathbb{R})$, we have the decomposition:

$$z = \sum_{j=-\infty}^{+\infty} \sum_{k} w_{j,k}\psi_{j,k} \tag{3.7}$$

Interpretation: Let $f \in L^2(\mathbb{R})$. We note $A_{2^j} f$ (resp. $D_{2^j} f$) the operator which approximates f (resp. the details of f) at resolution 2^j. More precisely, $A_{2^j} f$ (resp. $D_{2^j} f$) is the projection of f on V_j (resp. on W_j).

$$A_{2^j} f(x) = \sum_{k=-\infty}^{+\infty} \langle f(u), \phi_{j,k}(u)\rangle \, \phi_{j,k}(x) \tag{3.8}$$

$A_{2^j} f$ is caracterized by the sequence of scalar products $A^d_{2^j} f$:

$$A^d_{2^j} f = (\langle f(u), \phi_{j,k}(u)\rangle)_{k\in\mathbb{Z}} \tag{3.9}$$

We call $A^d_{2^j} f$ the discrete approximation of f at resolution 2^j. Each scalar product can be re-interpreted as a convolution product at point $2^{-j}k$:

$$\langle f(u), \phi_{j,k}(u)\rangle = (f(u) * \phi_{j,0}(-u))\,(k) \tag{3.10}$$

Hence:

$$A^d_{2^j} f = \left(\left(f(u) * \phi_{j,0}(-u) \right)(k) \right)_{k \in \mathbb{Z}} \tag{3.11}$$

(3.11) can be seen as a low-pass filtering of f.

When we study a signal, we just have a representation $A^d_1 f$, and we assume that $A_1 f = f$.

In the same way, we have:

$$D_{2^j} f(x) = \sum_{k=-\infty}^{+\infty} \langle f(u), \psi_{j,k}(u) \rangle \, \psi_{j,k}(x) \tag{3.12}$$

$D_{2^j} f$ is caracterized by the sequence of scalar products $D^d_{2^j} f$:

$$D^d_{2^j} f = \left(\langle f(u), \psi_{j,k}(u) \rangle \right)_{k \in \mathbb{Z}} \tag{3.13}$$

We call $D^d_{2^j} f$ the details of f at the resolution 2^j. Each scalar product can be re-interpreted as a convolution product at point $2^{-j}k$:

$$\langle f(u), \psi_{j,k}(u) \rangle = \left(f(u) * \psi_{j,0}(-u) \right)(k) \tag{3.14}$$

Hence:

$$D^d_{2^j} f = \left(\left(f(u) * \psi_{j,0}(-u) \right)(k) \right)_{k \in \mathbb{Z}} \tag{3.15}$$

(3.15) can be seen as a high-pass filtering of f.

In what follows, to simplify notations, we will no longer make difference between A_{2^j} and $A^d_{2^j}$ (resp. D_{2^j} and $D^d_{2^j}$).

Relation (3.4) means:

$$A_{2^{-j}} f + D_{2^{-j}} f = A_{2^{-2j+1}} f \tag{3.16}$$

and we also have:

$$A_1 f = \left(A_{2^{-J}} f, \ (D_{2^j} f)_{-J \leq j \leq -1} \right) \tag{3.17}$$

J is the order of the wavelet decomposition.

We can rewrite 3.17 as:

$$A_1 f = \sum_k u_{J,k} \phi_{J,k} + \sum_{j=-J}^{-1} \sum_k w_{j,k} \psi_{j,k} \tag{3.18}$$

The coefficients $u_{J,k}$ and $w_{j,k}$ are the wavelet coefficients of $A_1 f$, and (3.18) is called the wavelet decomposition.

$A_{2^{-1}}f$	$D^1_{2^{-1}}f$
$D^2_{2^{-1}}f$	$D^3_{2^{-1}}f$

Figure 4: 2-D case

3.2 2-D case

One way to get from the 1-D case to the 2-D case is to use tensor product. Scaling function and mother wavelet are then given by the following formulae:

- Scaling function:
$$\Phi(x,y) = \phi(x)\phi(y)$$
- Mother wavelet: (Ψ^1, Ψ^2, Ψ^3), with:
$$\begin{aligned} \Psi^1(x,y) &= \phi(x)\psi(y) \\ \Psi^2(x,y) &= \psi(x)\phi(y) \\ \Psi^3(x,y) &= \psi(x)\psi(y) \end{aligned}$$

Figure 4 shows how the wavelet transform of an image $A_1 f$ is classically represented.

The indice k in $D^k_{2^j} f$ is the same as in Ψ^k.

On can interpret $D^1_{2^j} f$ as the horizontal details of f, $D^2_{2^j} f$ as the vertical details, and $D^3_{2^j} f$ as the other details. It is possible to check it on figure 5.

$$A_1 f = \left(A_{2^{-J}} f, \left(D^1_{2^j} f\right)_{J \le j \le -1}, \left(D^2_{2^j} f\right)_{J \le j \le -1}, \left(D^3_{2^j} f\right)_{J \le j \le -1} \right)$$

Thanks to the wavelet coefficients, it is thus theoretically possible (see [37]) to differentiate textures out of orientation criteria.

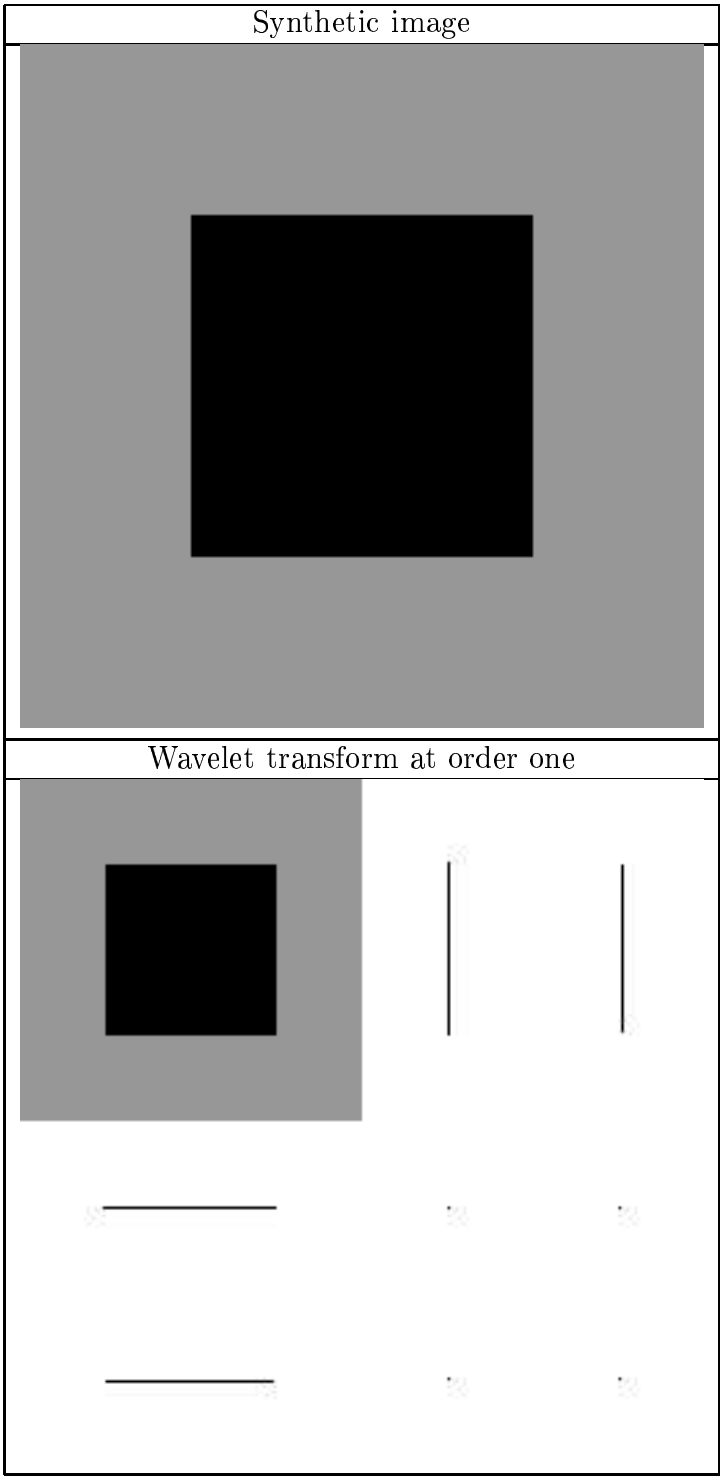

Figure 5: Orientation and wavelet transformation

Remark: The 2-D wavelets have been obtained here by separation of variables. They present priviledged directions: horizontal and vertical (and in a weaker way diagonal). But this is not the only possible construction (see for instance [40, 12, 11]).

3.3 Undecimated wavelets

The algorithm proposed by S.G. Mallat [37, 38, 36] gives a non redundant representation of a signal: once the wavelet transformation has been computed, the number of coefficients to store remains the same as for the initial signal. This is called decimated wavelets (in this algorithm, to get from an order of the decomposition to the next one, one uses a decimation after the high-pass and low-pass filtering).

Nevertheless, the use of decimated wavelets has the drawback to be (a priori) not translation invariant.

But the aim of our work is to classify textured images, and to obtain the same classification if a translation is made on a texture.

That is why we will use wavelet frames (see [53]). The DWF (discrete wavelet frame) corresponds to the DWT (discrete wavelet transform) applied to the input signal with all the possible translations.

It amounts to use undecimated wavelets.

3.4 Packet wavelet transform

A large number of textures can be modelized as quasi-periodic signals (repetition of the same stucture with slight variations) whose dominant frequencies lie in median frequencies channels (see [10]).

But the DWT decomposes a signal in a set of channels whose bandwidth is smaller in the low frequency regions. This happens to be sufficient for regular signals whose information is concentrated at low frequencies, but not for textures.

We note $h_{LL} = A_{2^{-1}f}$, $h_{LH} = D^1_{2^{-1}}f$, $h_{HL} = D^2_{2^{-1}}f$, $h_{HH} = D^3_{2^{-1}}f$. Figure 6 represents then the wavelet transform of an image. (h_{LL} corresponds to the low frequencies, h_{HH} to the high frequencies, and h_{HL} and h_{LH} to median frequencies).

h_{LL}	h_{LH}
h_{HL}	h_{HH}

Figure 6: Wavelet transform and filtering

In the case of the DWT, h_{LL} is the only channel which is redecomposed. In the case of the packet wavelets, each block can be decomposed again (see figure 7).

3.5 Gray level independence

When an image is decomposed with a wavelet transform or a packet wavelet transform, all the blocks of the decomposition have a zero mean, except the block corresponding to the low frequencies (h_{LL}, see figure 6 and 7). Its mean corresponds in fact to the mean gray level of the image. But our aim is to classify textured images. We want that two textures with the same structure, but with different means of gray level, be classified as the same texture: the mean of the gray level must not be a feature for a texture. That is why we modify the low frequencies block by setting its mean to zero.

4. Texture modelisation

We are now in position to caracterize textures through their wavelet decomposition.

4.1 Idea

We consider that a texture is caracterized by the energy of its wavelets coefficients. If we note z the function which represents this texture, we can write (using the notations of the preceeding section) (see (3.18)):

$$z = \sum_k u_{J,k}\phi_{J,k} + \sum_{j=-J}^{-1}\sum_k w_{j,k}\psi_{j,k} \tag{4.1}$$

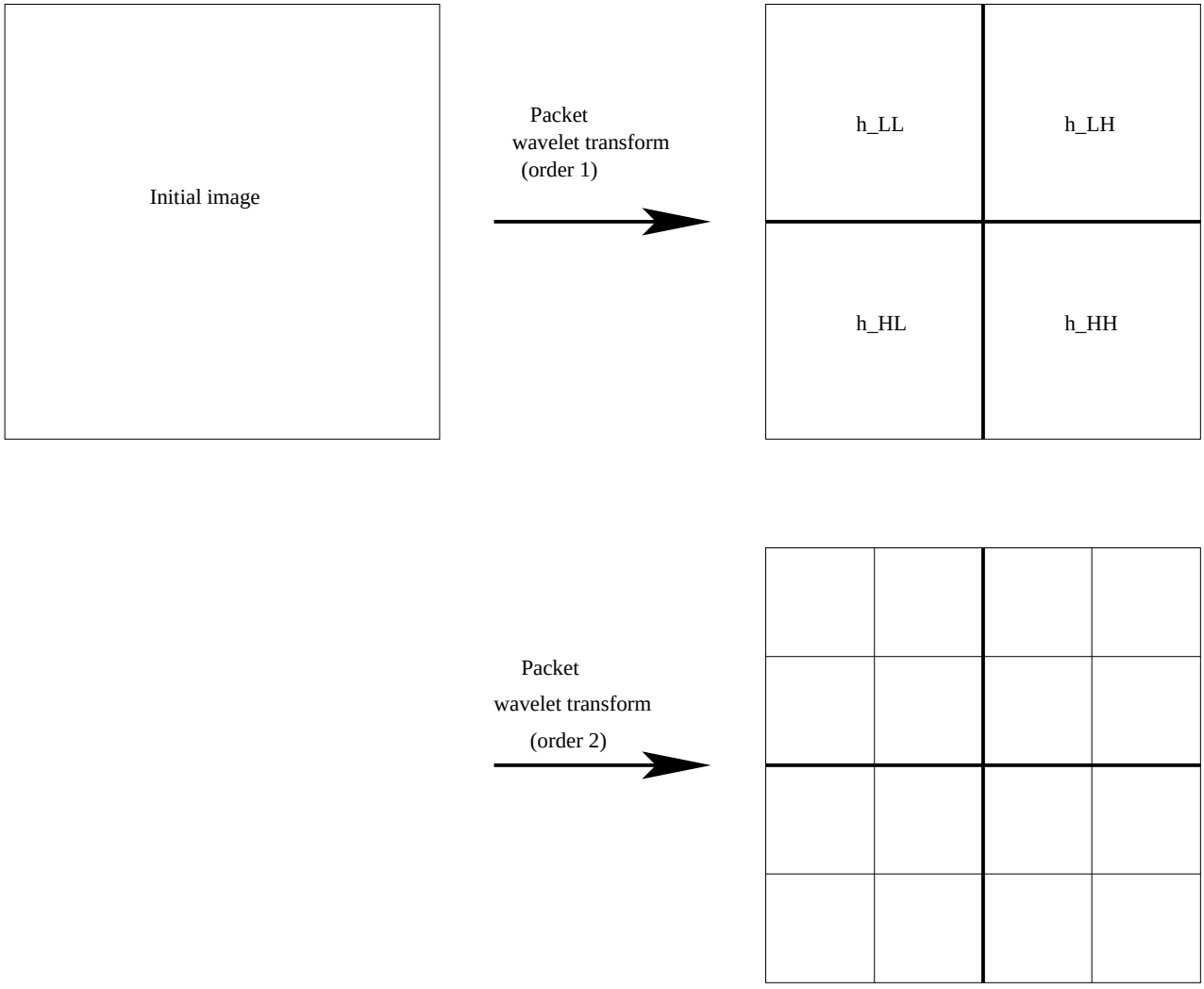

Figure 7: Packet wavelet transform: at each step, all the blocks of the wavelet transform are redecomposed

where ψ is the mother wavelet, ϕ the scaling function and J the order of the decomposition. Thus, we consider that a texture is caracterized by the sequence:

$$\big((|u_{J,k}|^2, k \in \mathbb{Z}), (|w_{j,k}|^2, k \in \mathbb{Z}, -J \leq j \leq -1)\big) \tag{4.2}$$

We compute a packet wavelet transform, since we consider that the main information of a texture lies also in the middle frequencies (and not only in the high frequencies) [10]. We use non decimated wavelets (in practice, the DWT is decimated but the needed translations are performed).

4.2 Probability distribution of the energy

S.G. Mallat checked experimentally (see [37, 38]) that the distribution of the modulus of the wavelets coefficients in a sub-band follows a generalized gaussian law of the form:

$$p_X(x) = A \exp\left(-\left(\frac{x}{\alpha}\right)^{\beta}\right) \mathbb{I}_{x \geq 0} \tag{4.3}$$

These ditributions are very picky. The parameter β modifies the decrease of the pick, and α models the variance. As we consider that textures are caracterized by their energy, we are going to compute the distribution law of the square of the wavelet coefficients in a sub-band. We get, thanks to (4.3) (with $X \geq 0$):

$$\begin{aligned} P(X^2 \leq a) &= P(X \leq \sqrt{a}) \\ &= \int_0^{\sqrt{a}} p_X dx \\ &= A \int_0^{\sqrt{a}} \exp\left(-\left(\frac{x}{\alpha}\right)^{\beta}\right) \mathbb{I}_{x \geq 0} dx \end{aligned}$$

We make the change of variable $x = \sqrt{y}$, and we thus get (for $a \geq 0$):

$$P(X^2 \leq a) = \frac{A}{2} \int_0^a \exp\left(-\left(\frac{\sqrt{y}}{\alpha}\right)^{\beta}\right) \frac{dy}{\sqrt{y}}$$

Hence:

$$p_{X^2}(y) = \frac{A}{2\sqrt{y}} \exp\left(-\left(\frac{\sqrt{y}}{\alpha}\right)^{\beta}\right) \mathbb{I}_{y \geq 0} \tag{4.4}$$

This kind of distribution is even pickier than the one given by (4.3). The parameter β modifies the decrease of the pick, and α models the variance.

Experimentally, we have checked that the distribution of the energy inside a sub-band is well approximated by a law of this type (see figure 8).

4.3 Computation of the parameters of the energy distribution

Our goal is to carry out supervised classification for textured images. "Supervised" means the fact that we know a priori the number of classes which can be found in the image, as well as the parameters of each class. The class parameters α and β can be computed from the first and second order moments of the energy distribution in each sub-band (and we will know these moments). Let us indeed compute these moments.

Gamma function: The function Γ is defined on $\mathbb{R}_+$ by:

$$\Gamma(t) = \int_0^{+\infty} e^{-u} u^{t-1} du \tag{4.5}$$

By using the fact that $\int_{\mathbb{R}} p_X(x)dx = 1$, we get:

$$A = \frac{\beta}{\alpha \Gamma\left(\frac{1}{\beta}\right)} \tag{4.6}$$

Let us denote $Y = X^2$. We compute the first and second order moments of Y, $M_1 = E(Y)$ and $M_2 = E(Y^2)$. Hence:

$$E(Z) = \int_{\mathbb{R}} p_Z(z)dz$$

We find:

$$M_1 = E(Y) = \frac{\alpha^3 A}{\beta} \Gamma\left(\frac{3}{\beta}\right) \tag{4.7}$$

$$M_2 = E(Y^2) = \frac{\alpha^5 A}{\beta} \Gamma\left(\frac{5}{\beta}\right) \tag{4.8}$$

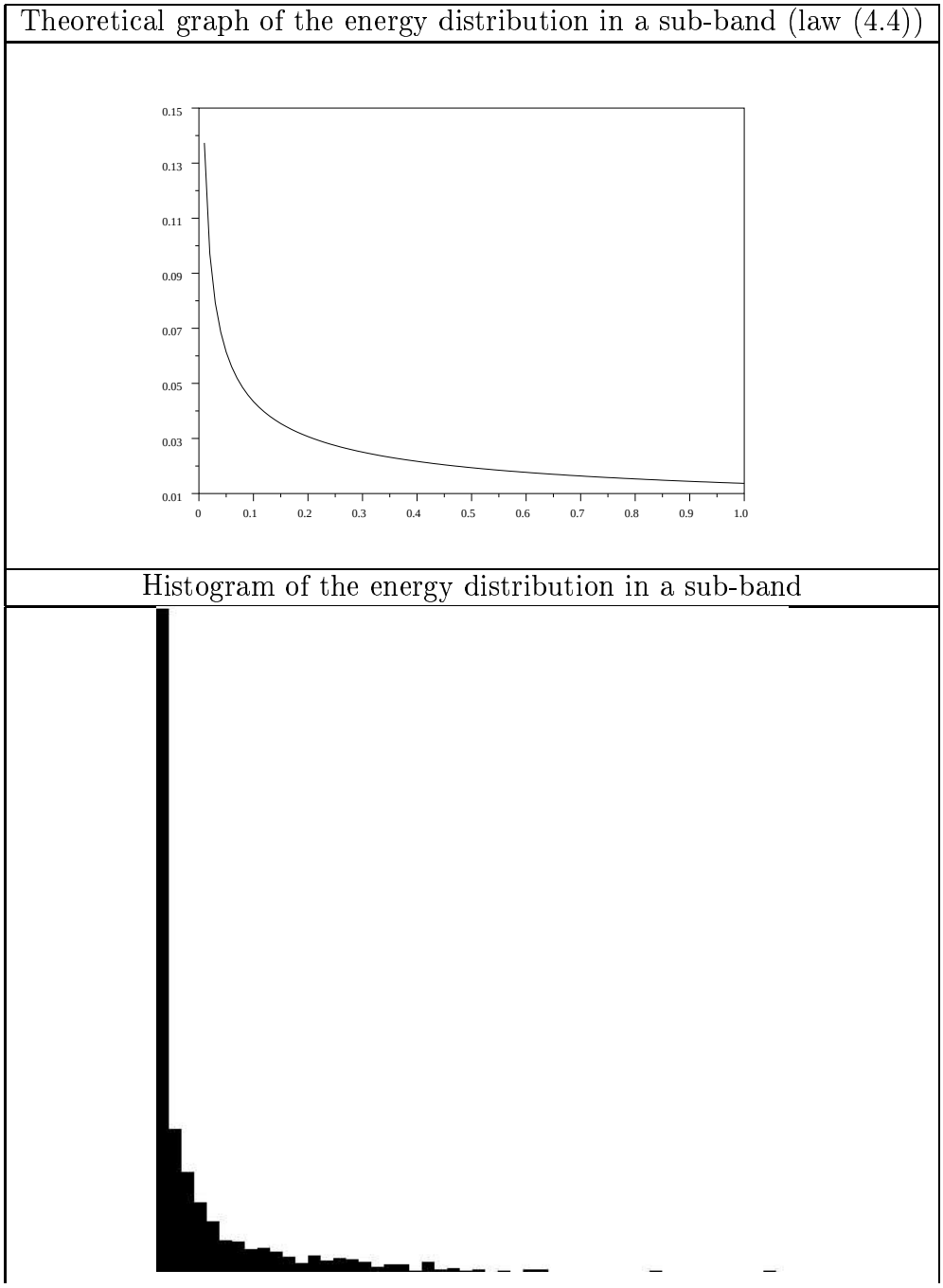

Figure 8: Theoretical graph of the energy distribution in a sub-band (law (4.4)) and experimental histogram

From (4.7) and (4.8), we get:

$$\alpha = \frac{\Gamma\left(\frac{3}{\beta}\right)}{\Gamma\left(\frac{5}{\beta}\right)}\frac{M_2}{M_1} \tag{4.9}$$

and

$$\frac{\Gamma^2\left(\frac{3}{\beta}\right)}{\Gamma\left(\frac{1}{\beta}\right)\Gamma\left(\frac{5}{\beta}\right)} = \frac{M_2}{M_1} \tag{4.10}$$

Thus:

$$\beta = F^{-1}\left(\frac{M_1^2}{M_2}\right) \tag{4.11}$$

with:

$$F(x) = \frac{\Gamma^2\left(\frac{3}{x}\right)}{\Gamma\left(\frac{1}{x}\right)\Gamma\left(\frac{5}{x}\right)} \tag{4.12}$$

The graph of F^{-1} is computed numerically and is shown on figure 9.

4.4 Proofs of the preceeding formulae

Proof of (4.6):

$$1 = \int_{\mathbb{R}} p_X(x)dx = A\int_0^{+\infty} \exp\left(-\left(\frac{x}{\alpha}\right)^{\beta}\right)dx \tag{4.13}$$

We then set $z = \frac{x}{\alpha}$:

$$1 = \alpha A\int_0^{+\infty} \exp\left(-z^{\beta}\right)dz \tag{4.14}$$

Then $u = z^{\beta}$:

$$\begin{aligned} 1 &= \frac{\alpha A}{\beta}\int_0^{+\infty} e^{-u}u^{\frac{1-\beta}{\beta}}du \\ &= \frac{\alpha A}{\beta}\Gamma\left(\frac{1}{\beta}\right) \end{aligned}$$

(thanks to (4.5)). Hence formula (4.6) holds.

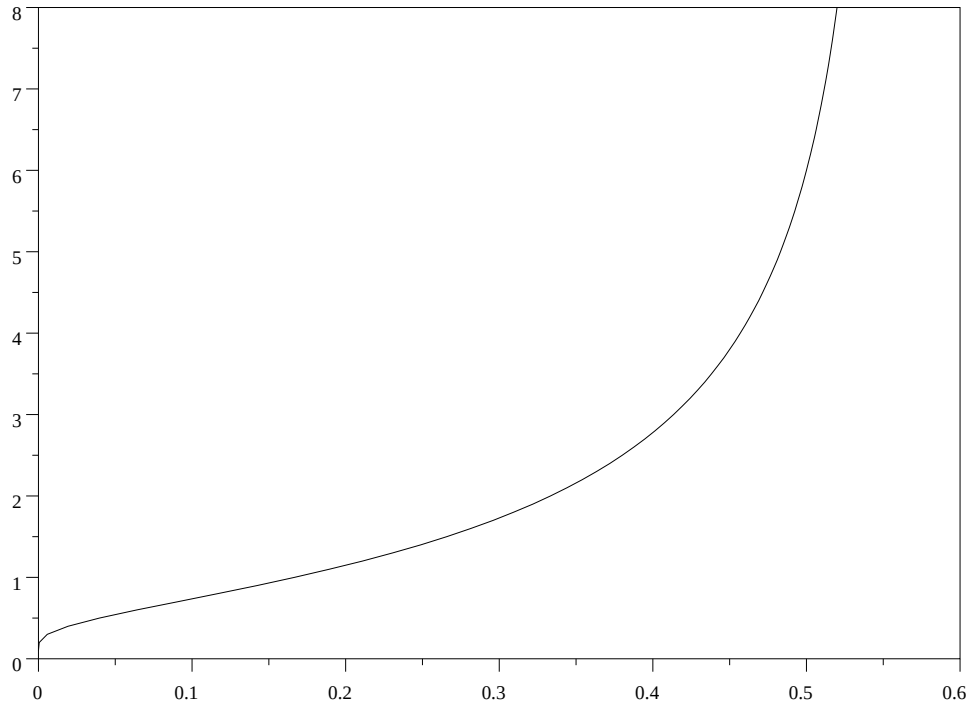

Figure 9: Graph of F^{-1} (given by (4.12))

Proof of (4.7):

$$E(X^2) = \int_0^{+\infty} \frac{A}{2} \frac{y}{\sqrt{y}} \exp - \left(\frac{\sqrt{y}}{\alpha} \right)^{\beta} dy \tag{4.15}$$

We set $x = \sqrt{y}$:

$$E(X^2) = \frac{A}{2} \int_0^{+\infty} 2x^2 \exp - \left(\frac{x}{\alpha} \right)^{\beta} dx \tag{4.16}$$

Then $v = \frac{x}{\alpha}$:

$$E(X^2) = \alpha A \int_0^{+\infty} \alpha^2 v^2 \exp - (v)^{\beta} dv \tag{4.17}$$

Next, $v = u^{\frac{1}{\beta}}$:

$$\begin{aligned} E(X^2) &= \frac{\alpha^3 A}{\beta} \int_0^{+\infty} u^{\frac{2}{\beta}} e^{-u} u^{\frac{1}{\beta}-1} du \\ &= \frac{\alpha^3 A}{\beta} \int_0^{+\infty} u^{\frac{3}{\beta}-1} e^{-u} du \end{aligned}$$

Hence (with (4.5)):

$$E(X^2) = \frac{\alpha^3 A}{\beta} \Gamma \left(\frac{3}{\beta} \right) = E(Y) \tag{4.18}$$

i.e. the proof of (4.7).

Proof of (4.8) :

$$E(Y^2) = \int_0^{+\infty} \frac{A}{2} \frac{y^2}{\sqrt{y}} \exp - \left(\frac{\sqrt{y}}{\alpha} \right)^{\beta} dy \tag{4.19}$$

We set $x = \sqrt{y}$:

$$E(Y^2) = \frac{A}{2} \int_0^{+\infty} 2x^4 \exp - \left(\frac{x}{\alpha} \right)^{\beta} dx \tag{4.20}$$

Then $v = \frac{x}{\alpha}$:

$$E(Y^2) = \alpha A \int_0^{+\infty} \alpha^4 v^4 \exp - (v)^\beta dv \tag{4.21}$$

Next, $v = u^{\frac{1}{\beta}}$:

$$\begin{aligned} E(Y^2) &= \frac{\alpha^5 A}{\beta} \int_0^{+\infty} u^{\frac{4}{\beta}} e^{-u} u^{\frac{1}{\beta}-1} du \\ &= \frac{\alpha^5 A}{\beta} \int_0^{+\infty} u^{\frac{5}{\beta}-1} e^{-u} du \end{aligned}$$

which implies (with (4.5)):

$$E(Y^2) = \frac{\alpha^5 A}{\beta} \Gamma\left(\frac{5}{\beta}\right) = E(X^4) \tag{4.22}$$

And we thus get (4.8).

5. Complete functional

5.1 Hypotheses

From the modelization we made in the previous section, we now deduce the data term we will use in our algorithm.

We compute a packet wavelet decomposition of the image (up to the second order in practice): we get I channels ($I = 16$ in practice).

Définition 5.1. In what follows, we will call the energy at pixel s the vector $u(s) = (u_1(s), \ldots, u_I(s))$, where $u_i(s)$ is the square of the wavelet coefficient in the sub-band i at pixel s.

$U(s)$ will be the texture's feature at pixel s.

Hypotheses:

(H **1)** We assume that, for each texture $k = 1 \ldots K$, in each channel $i = 1 \ldots I$, the square of the wavelet coefficients follows a law of the type (4.4) of mean $M_1^{k,i}$ and of second order moment $M_2^{k,i}$.

(H**2**) We consider that the different channels are independent.

The index k stands for the class, and i for the channel (sub-band). As we mentioned before, our goal is to perform supervised classification. The first and second order moments $M_1^{k,i}$ and $M_2^{k,i}$ are assumed to be known by the user.

5.2 Data term

The goal is to find for each pixel the class which makes the observed energy U the most likely (U corresponds to the energy in the wavelet decomposition which enables us to characterize textures).

We begin by using discrete variables. We discretize the set Ω: we consider that it is composed of the set of the image pixels. We denote S the set of the pixels. We want to maximize $P(U|Cl)$, where Cl is the assumed class.

Remark: $P(U(s) = u|Cl(s) = k)$ is the probability that the energy U at pixel s takes the value u knowing that the pixel s lies in the class k. So maximizing $P(U(s)|Cl(s))$ amounts to finding the class which makes the energy the most likely for the pixel s when its energy is $U(s)$. And maximizing $P(U|Cl)$ amounts in fact in seeking the class which makes the energy the most probable for each pixel depending on their energy.

Computation of the maximum likelihood (ML): According to our hypotheses (H 1) and (H 2), the probability density is:

$$f(u|Cl = k) = \prod_{i=1}^{I} \frac{A_k^i}{2\sqrt{u_i}} \exp\left(-\left(\frac{\sqrt{u_i}}{\alpha_k^i}\right)^{\beta_k^i}\right) \tag{5.1}$$

The parameters A_k^i, α_k^i and β_k^i are computed from the first and second order moments thanks to formulae (4.6), (4.9) and (4.11).

Let us recall the notations:

- K is the number of classes.
- I is the number of channels (sub-bands) (in practice, $I = 16$).

- α_k^i and β_i^k are the parameters of the energy probability law (i.e. the density probability of the square of the wavelet coefficients, see (4.4)) of the class k in the channel i.
- $u_i(s)$ is the square of the wavelet coefficient in the channel i (at s).
- $u(s)$ is the energy value at s: $u(s) = (u_1(s), \ldots, u_I(s))$ (see definition 5.1).

In fact, we are going to maximize the log-likelihood, which amounts to minimize:

$$\begin{aligned} -\ln(P(U|Cl) &= -\sum_{s\in S} \ln(P(U_s|Cl = s) \\ &= -\sum_{s\in S}\sum_{k=1}^{K} \ln(P(U_s|Cl = s)\delta_k(s) \\ &= -\sum_{k=1}^{K}\sum_{s\in S} \ln(P(U_s|Cl = s)\delta_k(s) \end{aligned}$$

where $\delta_k(s) = 1$ if s belongs to the class k, 0 otherwise.
By using (5.1), it can be deduced:

$$-\ln(P(U|Cl) = -\sum_{k=1}^{K}\sum_{s\in S}\sum_{i=1}^{I}\left(\ln A_k^i - \ln 2 - \frac{1}{2}\ln u_i(s) - \left(\frac{\sqrt{u_i(s)}}{\alpha_k^i}\right)^{\beta_k^i}\right)\delta_k(Cl_s) \tag{5.2}$$

We rewrite now (5.2) in a continuous framework. Here $\delta_k(Cl_s) = 1$ means $x \in \Omega_k$.

$$-\ln(P(U|Cl) = \sum_{k=1}^{K}\sum_{i=1}^{I}\int_{\Omega_k}\left(-\ln A_k^i + \ln 2 + \frac{1}{2}\ln u_i(x) + \left(\frac{\sqrt{u_i(x)}}{\alpha_k^i}\right)^{\beta_k^i}\right)dx \tag{5.3}$$

Finally, our fitting term to the data is:

$$F^C(\Phi_1,\dots,\Phi_K) = \sum_{k=1}^{K}\sum_{i=1}^{I}\int_{\Omega_k}\left(-\ln A_k^i + \ln 2 + \frac{1}{2}\ln u_i(x) + \left(\frac{\sqrt{u_i(x)}}{\alpha_k^i}\right)^{\beta_k^i}\right)dx \tag{5.4}$$

And we approximate 5.4 by:

$$F_\alpha^C(\Phi_1,\dots,\Phi_K) = \sum_{k=1}^{K}\sum_{i=1}^{I}\int_{\Omega} H_\alpha(\Phi_k)\left(-\ln A_k^i + \ln 2 + \frac{1}{2}\ln u_i(x) + \left(\frac{\sqrt{u_i(x)}}{\alpha_k^i}\right)^{\beta_k^i}\right)dx \tag{5.5}$$

5.3 The functional

We are now able to completely write the functional which models the classification problem for textured images:

$$\begin{aligned} F(\Phi_1,\dots,\Phi_K) &= \frac{\lambda}{2}\int_\Omega\left(\sum_{k=1}^{K}H_\alpha(\Phi_k)-1\right)^2 + \sum_{k=1}^{K}\gamma_k\int_\Omega \delta_\alpha(\Phi_k)\,|\nabla\Phi_k| \\ &\quad + \sum_{k=1}^{K} e_k \sum_{i=1}^{16}\int_\Omega H_\alpha(\Phi_k)\left(-\ln A_k^i + \ln 2 + \frac{1}{2}\ln u_i(x) + \left(\frac{\sqrt{u_i(x)}}{\alpha_k^i}\right)^{\beta_k^i}\right)dx \end{aligned} \tag{5.6}$$

We have introduced some weights in front of each energy term of our functional. They allow us, if needed, to use more a priori about the data. But in practice, we take $\gamma_1 = \cdots = \gamma_K = \gamma$ and $e_1 = \cdots = e_K = 1.0$, so that it remains just two parameters to set: λ and γ).

The classical mathematical minimization results do not apply to our functional. Its mathematical study is under investigation (existence and uniqueness of functions $\Phi_1,\dots,\Phi_K$ minimizing (5.6)).

5.4 Remarks

Use of a window In order to get more regular results, we have sometimes used a window to compute the data term (such as in [53, 56]). In this case, the associated energy to a pixel s in the channel i is in fact a weighted average of the energies in this channel of the pixels contained in a window around the pixel s considered. We have seen that the use of an arithmetic mean induced a loss of the smallest structures. In practice, when we use a window, we use a Gaussian mask. Figures 10 and 11 show the influence of the use of a window on the computation of the data term. The classifications that are shown are done according to the data term only.

Nevertheless, most of the time, we have not used any averaging window. Indeed, the wavelets coefficients at a point already depend on the information in a neighbourhood of this point.

Other criteria We have tried other criteria than energy, especially entropy. But in our preliminary trials, energy has proved to be the most interesting criterion.

Moreover, in [29], the authors show experimentally that energy is more adapted than entropy as texture feature. That is why we have considered energy.

5.5 Euler-Lagrange Equations

Let us recall here the functional we want to minimize (for solving the classification problem):

$$\begin{aligned} F\left(\Phi_1,\ldots,\Phi_K\right) &= \frac{\lambda}{2}\int_\Omega \left(\sum_{k=1}^{K} H_\alpha\left(\Phi_k\right)-1\right)^2 + \sum_{k=1}^{K}\gamma_k\int_\Omega \delta_\alpha\left(\Phi_k\right)\left|\nabla\Phi_k\right| \\ &\quad + \sum_{k=1}^{K} e_k \sum_{i=1}^{16}\int_\Omega H_\alpha(\Phi_k)\left(-\ln A_k^i + \ln 2 + \frac{1}{2}\ln u_i(x) + \left(\frac{\sqrt{u_i(x)}}{\alpha_k^i}\right)^{\beta_k^i}\right)dx \end{aligned} \tag{5.7}$$

If $\Phi_1,\ldots,\Phi_K$ minimize the functional F, then necessarily (if furthermore the $\Phi_1,\ldots,\Phi_K$ verify some regularity conditions (see [31] for instance)), we

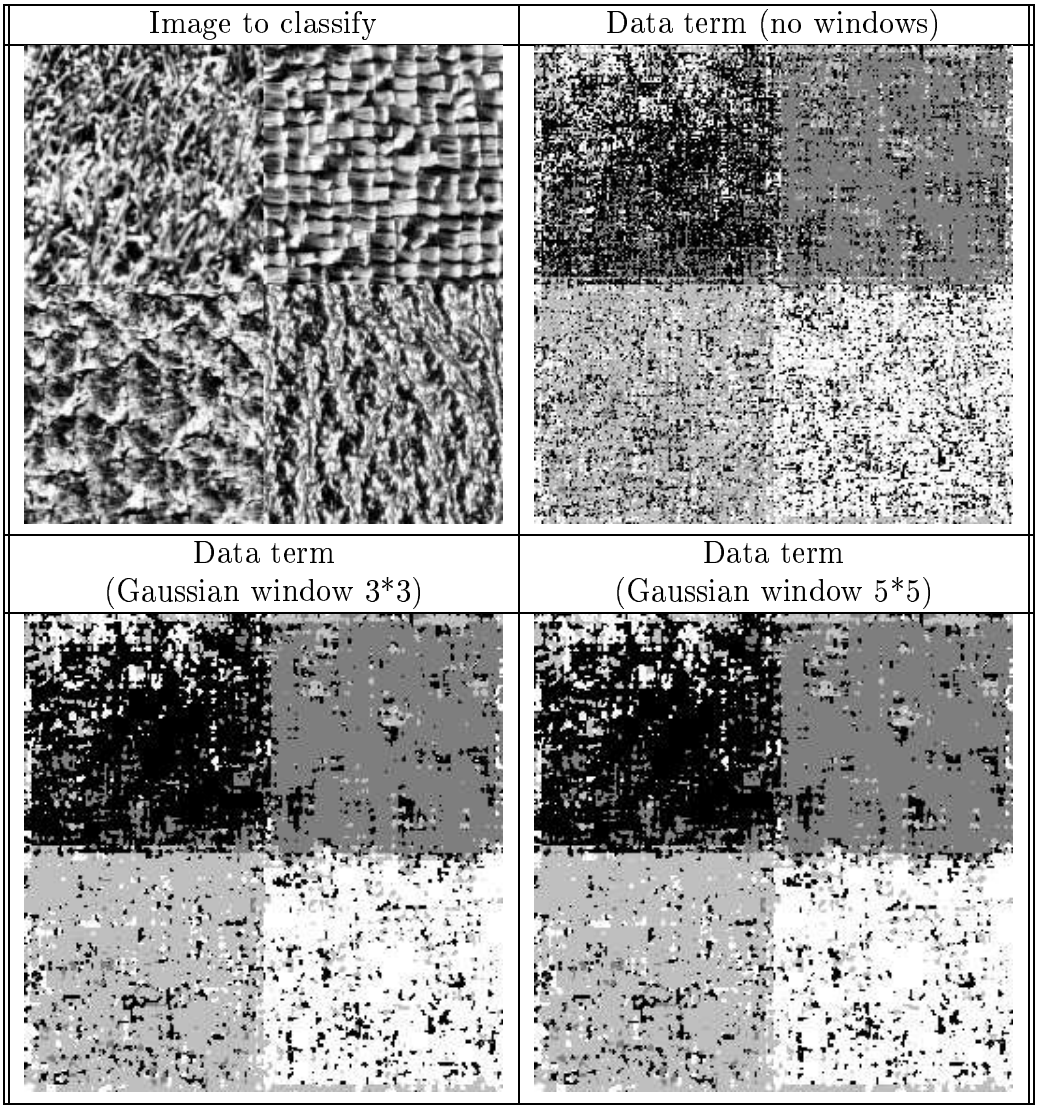

Figure 10: Use of a Gaussian window

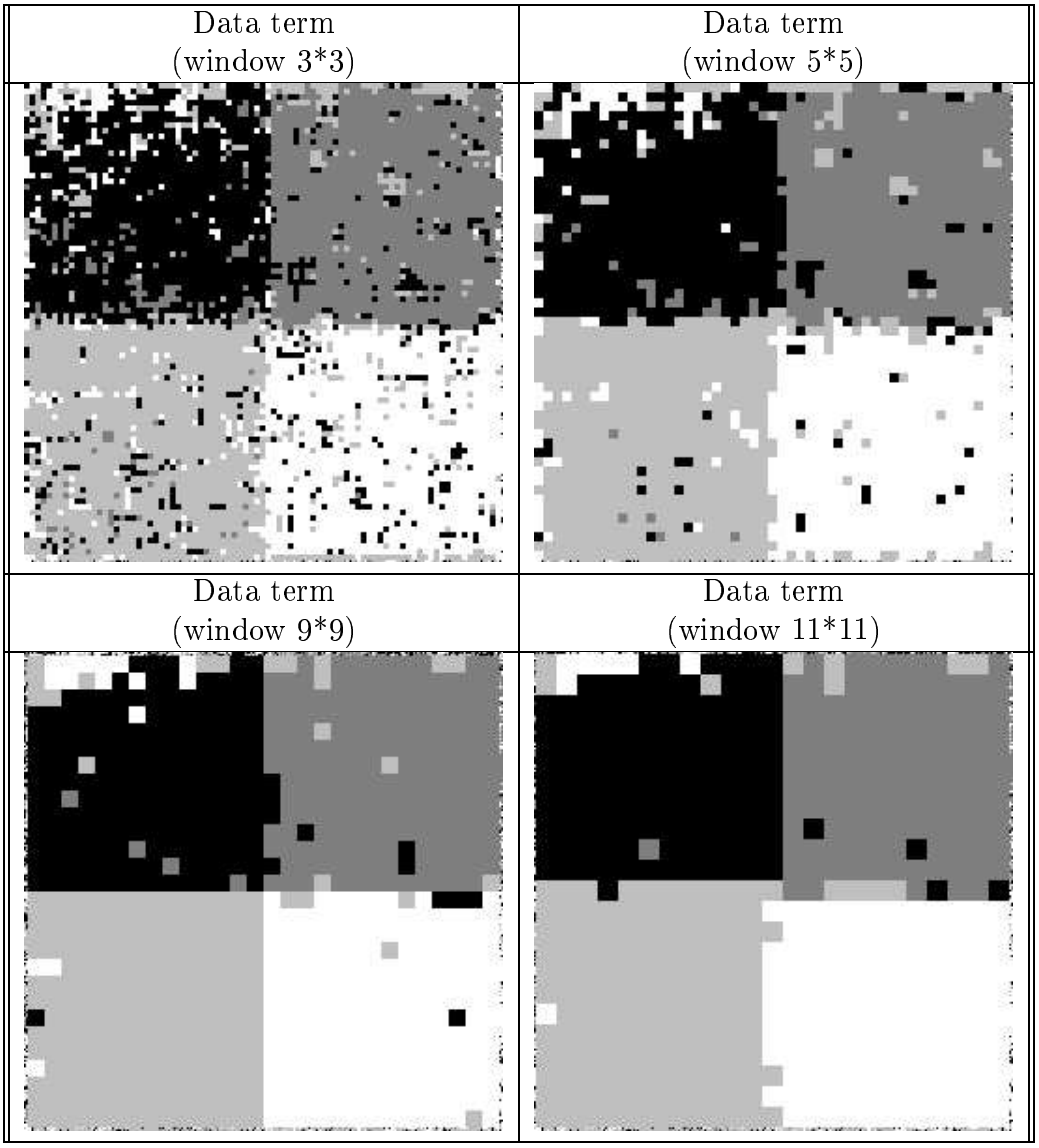

Figure 11: Use of an arithmetic window

have for $k = 1 \dots K$:

$$\frac{\partial F}{\partial \Phi_k} = 0 \tag{5.8}$$

Assuming that Neumann conditions are verified $\left(\frac{\partial \Phi_k}{\partial \vec{n}}(x) = 0 \ , \forall x \in \partial\Omega\right)$, the associated Euler-Lagrange equations to F (see [31]) give the following system composed of K-coupled PDE's: we have for $k = 1 \dots K$

$$\begin{aligned} 0 &= \lambda \delta_\alpha(\Phi_k) \left(\sum_{q=1}^{K} H_\alpha(\Phi_q) - 1 \right) - \gamma_k \delta_\alpha(\Phi_k) div \left(\frac{\nabla \Phi_k}{|\nabla \Phi_k|} \right) \\ &\quad + e_k \delta_\alpha(\Phi_k) \left(\sum_{i=1}^{16} \left(-\ln A_k^i + \ln 2 + \frac{1}{2} \ln u_k^i + \left(\frac{\sqrt{u_k^i}}{\alpha_k^i} \right)^{\beta_k^i} \right) \right) \end{aligned} \tag{5.9}$$

The term $div \left(\frac{\nabla \Phi_k(x)}{|\nabla \Phi_k(x)|} \right)$ can be interpreted as the mean curvature of the level set Φ_k at point x.

5.6 Dynamical scheme

To solve the PDE's system (5.9), we embed it in the following dynamical scheme ($k = 1 \dots K$):

$$\begin{aligned} \frac{\partial \Phi_k}{\partial t} &= -\delta_\alpha(\Phi_k) \left[2\lambda \left(\sum_{q=1}^{K} H_\alpha(\Phi_q) - 1 \right) - \gamma_k div \left(\frac{\nabla \Phi_k}{|\nabla \Phi_k|} \right) \right. \\ &\quad \left. + e_k \left(\sum_{i=1}^{16} \left(-\ln A_k^i + \ln 2 + \frac{1}{2} \ln u_i(x) + \left(\frac{\sqrt{u_i(x)}}{\alpha_k^i} \right)^{\beta_k^i} \right) \right) \right] \end{aligned} \tag{5.10}$$

with the initial condition $\Phi_k(0, x)$ is the Euclidean signed distance function to the zero level set Φ_k.

We discretize this system with finite differences (see [48]). We arbitrarily set $\alpha \in \mathbb{R}^+$ (in our experiments, we have used $\alpha = 3.0$). We need to set α small enough so that the approximations of the Dirac and Heaviside distributions (δ_α and H_α) remain reasonable. We also prefer α to be small since the bandwidth on which the PDE's (5.10) are not empty depends on it (if $\alpha \in \mathbb{N}$, it contains

$2\alpha - 1$ pixels width when ϕ_k is the signed distance to its zero level set). On the contrary, we need α not to be too small so that the band contains enough pixels to compute the different terms of (5.10).

5.7 Reinitialization

The Φ_k are initialized as Euclidean signed distance functions. Nevertheless, as in the classical active contours method [4]), the evolution of the Φ_k with respect to (5.10) does not keep them as Euclidean signed distance functions to their zero level sets. This prevents the convergence of (5.10) towards some Φ_k minimizing F.

That is why it is necessary to periodically reinitialize the Φ_k functions into Euclidean signed distance functions (in our experiments, we perform reinitialization for the Φ_k each three iterations of (5.10)). This is a general problem of the active contours methods with level sets. For speed of convergence and also for numerical reasons ([26, 51, 46, 1]), it is interesting that the Φ_k verify:

$$0 < A < |\nabla \Phi_k| < B < \infty$$

and that is precisely the case when the Φ_k are Euclidean signed distance functions (we then have $|\nabla \Phi_k|(x) = 1$ almost everywhere in x).

To reinitialize Φ_k into the Euclidean signed distance function, we use the PDE:

$$\frac{\partial \Phi}{\partial t} + \text{sign}(\Phi_k)(|\nabla \Phi| - 1) = 0 \tag{5.11}$$

A theoretical study of this PDE, as well as an extension to more general Hamiltonians can be found in [2]. The authors of [2] show that, if $\Phi(0, x) = \Phi_k(x)$, the viscosity solution of (5.11) converges towards the Euclidean signed distance function to the zero level set of Φ_k.

In practice, the Φ_k are reinitialized into Euclidean signed distance functions to their zero level set in a neighbourhood of their zero level set (it is not necessary to compute the Φ_k in the whole image, which considerably increases the speed of the algorithm). The width of these neighbourhood being given, [2] give then the number of iterations to perform with PDE (5.11) to reinitialize exactly the Φ_k in these neighbourhoods.

6. Numerical results

6.1 Choice of parameters

Wavelets: As we explained before, we have chosen in our experiments to use wavelet packet decompositions of the second order to compute the data term. We have tested different kinds of wavelets, and we have chosen to use the Daubechies wavelet with ten vanishing moments (see [18, 38]), this choice appearing to give the best results.

Information given by the user: As the classification is here supervised, the user has to give the number of classes (textures), as well as the parameters of each class (the first and second order moments of the energy distribution in each sub-band of the packet wavelet decomposition).

Parameters: In our experiments, we always choose $e_1 = \cdots = e_K = 1.0$ and $\gamma_1 = \cdots = \gamma_K = \gamma$ ($\gamma \in \mathbb{R}$). There remains only two parameters to set: the partition term coefficient λ, and the common value of the contour regularization terms γ.

Initialization: In a first time, we have proceeded to a manual initialization with circles or squares as in [48]. Each circle represents then the zero level set of one of the class (see figure 12).

To get an automatic initialization, and to make it independent of the user, we have then used "seeds": we split the initial image into small sub-images (in practice 5*5 images). In each sub-image, for each class k, we compute the data term by assuming that all the pixels of the sub-image belong to the same class k. We set all the pixels in the sub-image to the class k for which the whole sub-image's energy is the smallest (see figure 13). We have used this initialization in the examples presented here-after.

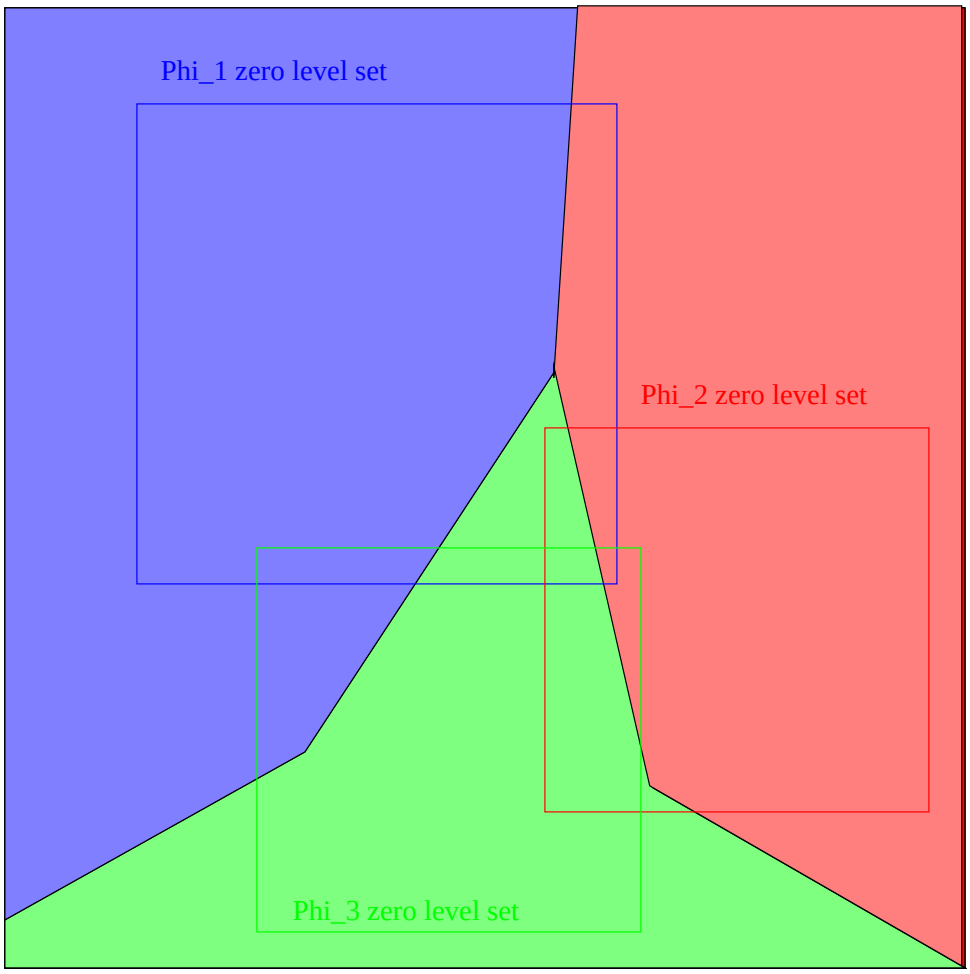

Figure 12: Manual initialization

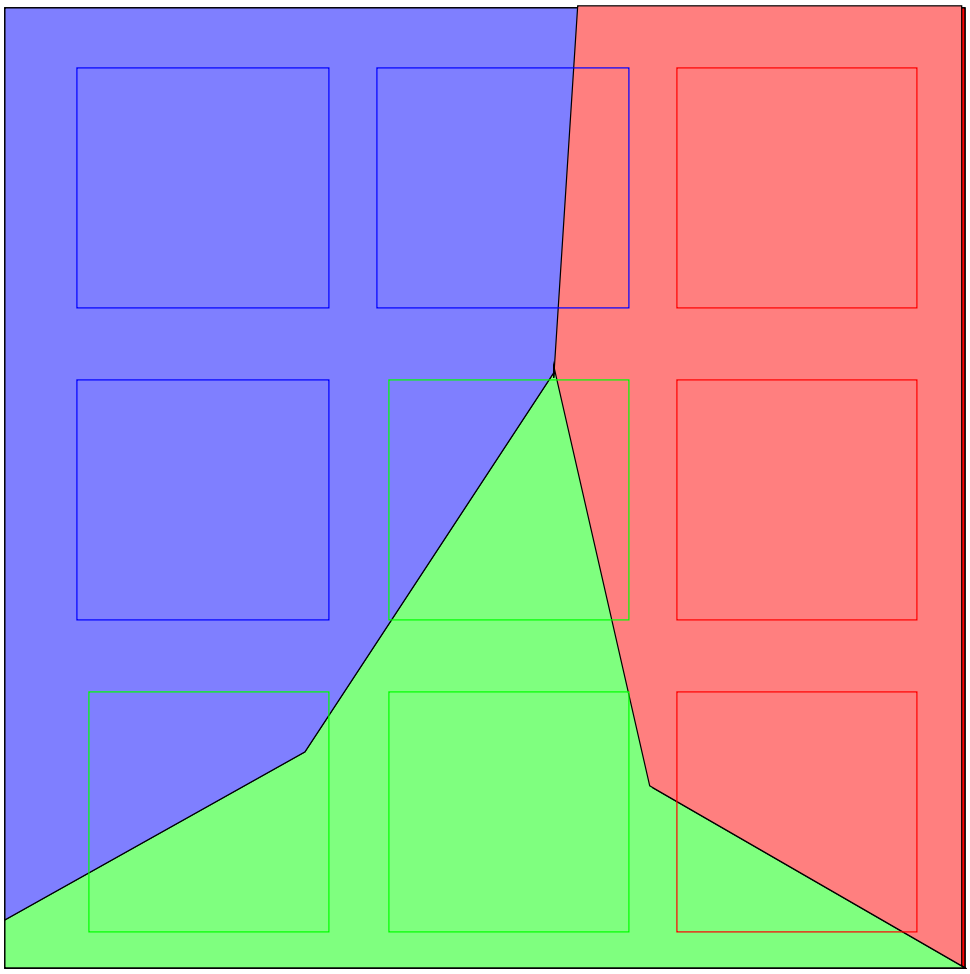

Figure 13: Seeds initialization

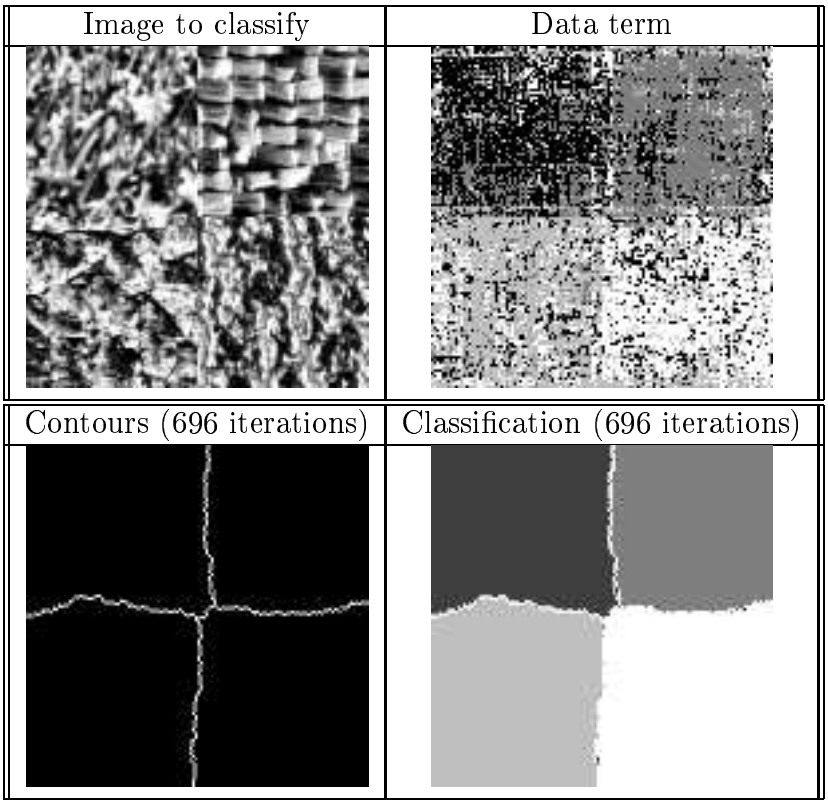

Figure 14: Classification of a synthetic image composed of four textures

6.2 Examples:

6.2.1 Synthetic image with four textures

In this example (see figure 14), one sees clearly that our model can handle with triple junctions. On the contrary, as in the classical approach of the Mumford-Shah functional [42], the junction of four textures give two triple junctions in the classified image (at 120 degrees).

It is also worth noticing that our algorithm can differentiate these four textures although they are visually very closed.

In the figures 14 to 17, we represent the data term by giving to each pixel of the image the class for which its energy is the smallest. This is the classification that we would get if we only use the data term(Maximum Likelihood criterion).

6.2.2 Synthetic image with two textures

This example (voir figure 15) shows that our model can handle with any kind of geometrical shape (contrary to the model proposed in [52] for instance).

6.2.3 Synthetic image with six textures

This example (see figure 16) shows that our model can handle complex textured images. Here, some of the textures are visually very close, and the geometrical shape of the contours are yet quite well detected. To get more homogeneous classes, we have applied here a Gaussian mask to the data term.

6.2.4 Case of a real image with a zebra

This example (see figure 17) also shows that our model can handle with any geometrical shape.

7. Conclusion

We have presented a variational model based on a level set formulation to classify textured images. We analyze the textures by using a wavelet transform: we consider that textures are caracterized by their energy distribution in each

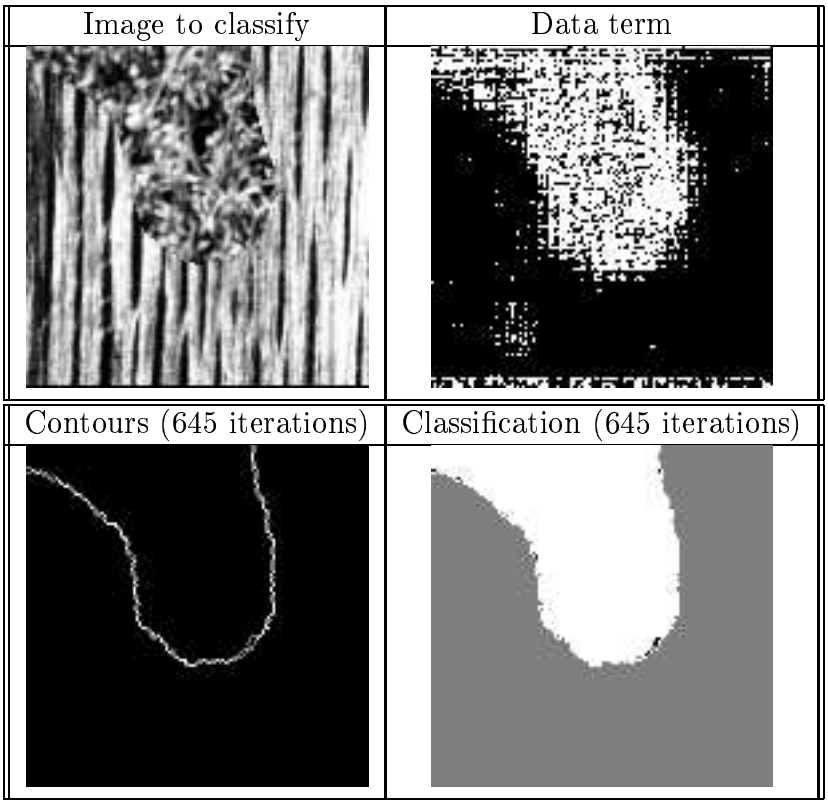

Figure 15: Classification of a synthetic image composed of two textures

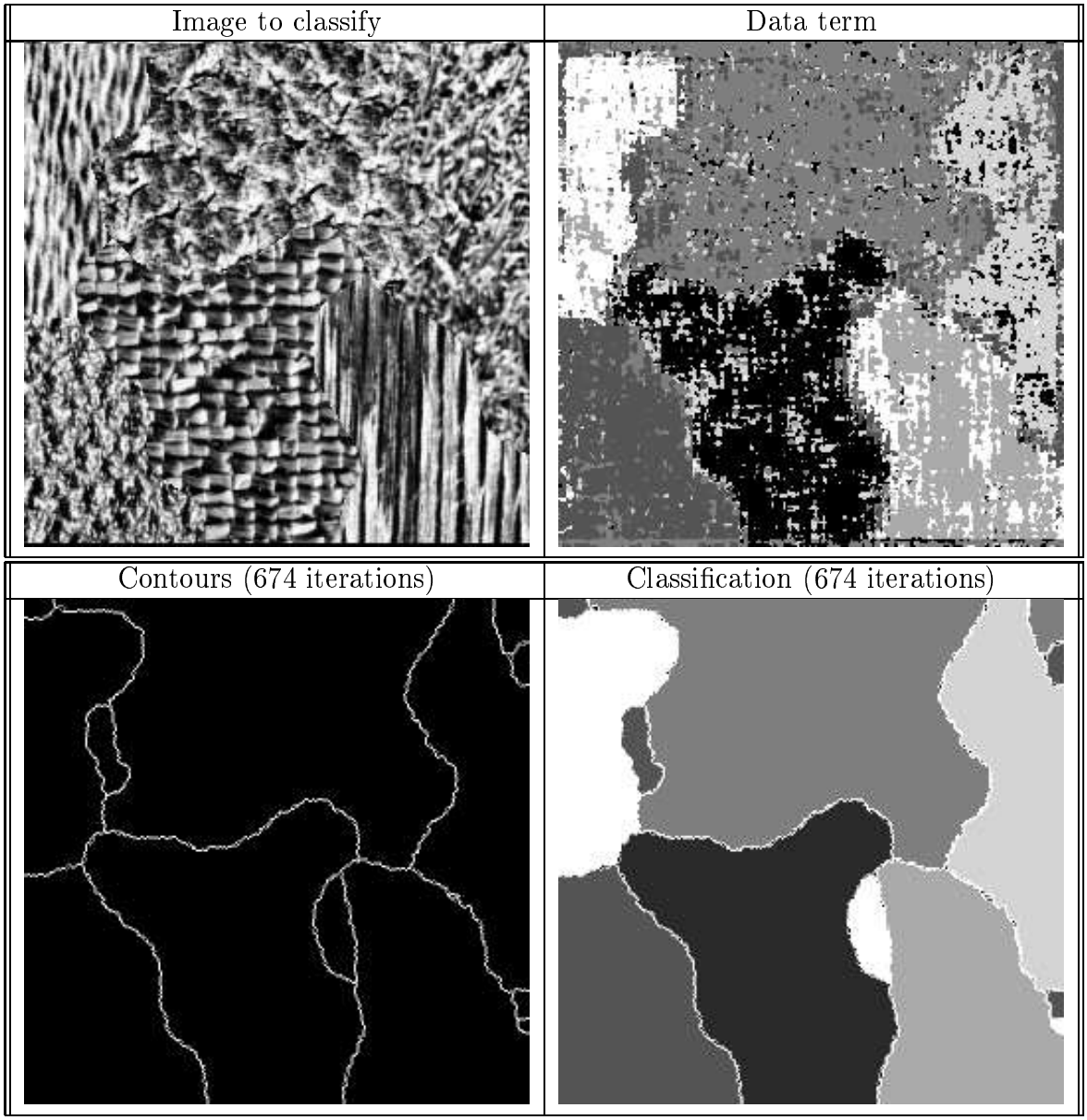

Figure 16: Classification of a synthetic image with six textures

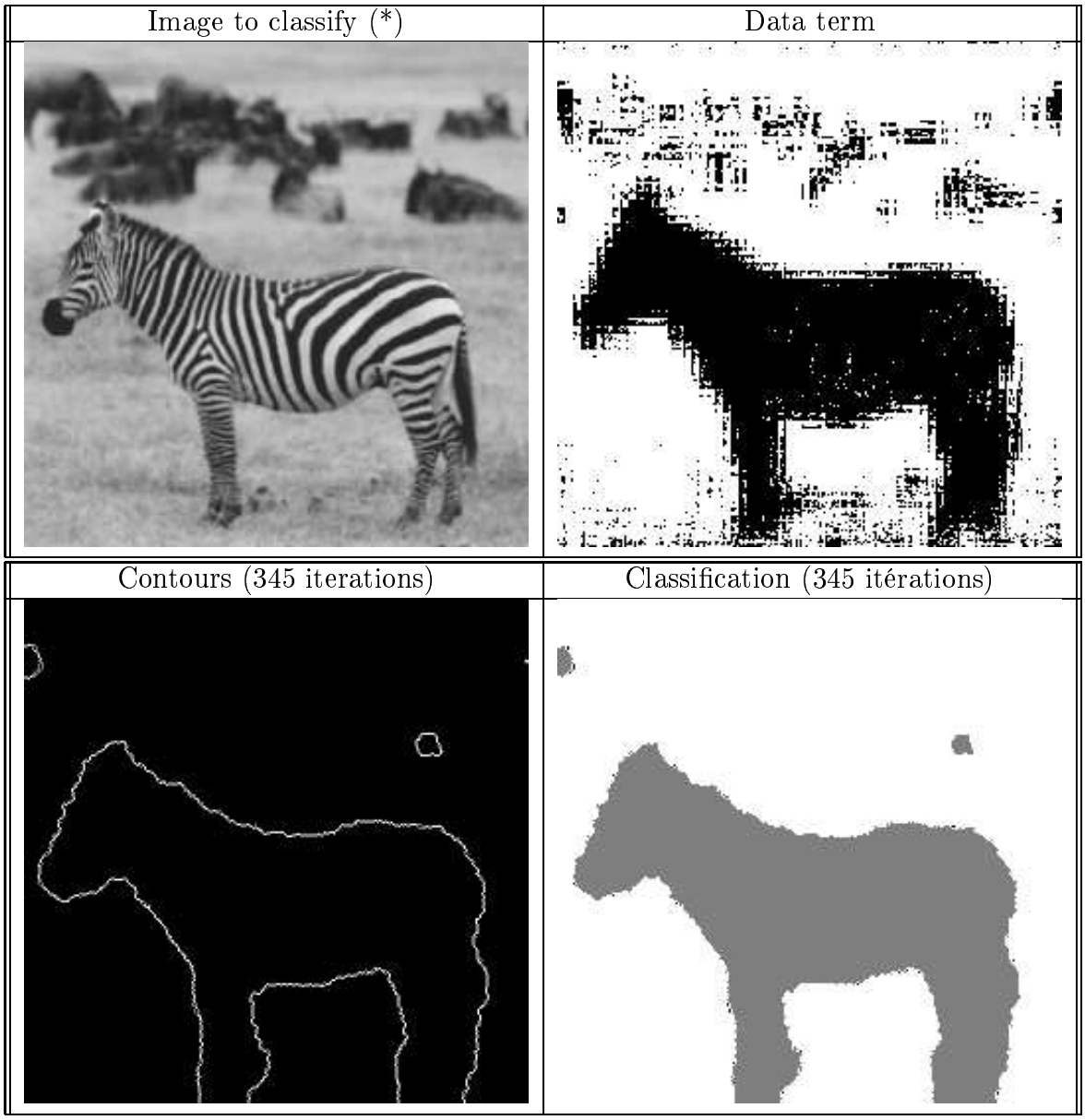

Figure 17: Classification of a zebra on a background.
(*) Copyright © Corel. All rights reserved.

sub-band of the decomposition. The level set formulation is a way to represent regions and interfaces with a continuous function defined on the whole image support. The functional minimization leads to a coupled PDE's system, which is embedded in a dynamical scheme. Each PDE guides the level set of a function by taking into account external forces (contours regularization) and internal forces (data term and partition). We give results for both synthetic and real images. In a future work, we plan to develop an unsupervised classification algorithm from our supervised classification algorithm: to do this, we will have to estimate in an efficient way the number of classes, as well as the parameters of each class. We also plan to make a complete theoretical study of our model.

References

[1] D. Adalsteinsson and J.A. Sethian. The fast constructions of extension velocities in level set methods. *Journal of Computational Physics*, 148:2–22, 1999.

[2] G. Aubert and J.F. Aujol. Signed distance functions and viscosity solutions of discontinuous hamilton-jacobi equations. Technical Report 4507, INRIA, July 2002. submitted.

[3] G. Aubert and L. Blanc-Feraud. Some remarks on the equivalence between 2d and 3d classical snakes and geodesic active contours. *IJCV*, 34(1):19–28, 1999.

[4] G. Aubert and P. Kornprobst. *Mathematical Problems in Image Processing*, volume 147 of *Applied Mathematical Sciences*. Springer-Verlag, 2002.

[5] M. Berthod, Z. Kato, S. Yu, and J. Zerubia. Bayesian image classification using markov random fields. *Image and Vision Computing*, 14(4):285–293, 1996.

[6] C. Bouman. Multiple resolution segmentation of textured images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(2):99–113, July 1995.

[7] C.A. Bouman and M. Shapiro. A multiscale random field model for bayesian image segmentation. *IEEE Trans. on Image Precessing*, 3:162–177, 1994.

[8] V. Caselles, F. Catte, T. Coll, and F. Dibos. A geometric model for active contours. *Numerische Mathematik*, 66:1–31, 1993.

[9] V. Caselles, R. Kimmel, and G. Sapiro. Geodesic active contours. *IJCV*, 22(1):61–79, 1997.

[10] T. Chang and C.C.J. Kuo. Texture analysis and classification with tree-structured wavelet transform. *IEEE Transactions on Image Processing*, 2(4):429–441, October 1993.

[11] D. Charalampidis and T. Kasparis. Wavelet-based rotational invariant roughness features for texture classification and segmentation. *IEEE Transcations on Image Processing*, 11(8):825–837, August 2002.

[12] J.L. Chen and A. Kundu. Rotation and gray scale transform invariant texture identification using wavelet decomposition and hidden markov model. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(2):208–214, February 1994.

[13] H. Choi and R. Baraniuk. Interpolation and denoising of nonumiformly sampled data using wavelet-domain processing. In *SPIE*, volume 3, pages 1645–8, 1999.

[14] H. Choi and R. Baraniuk. Multiple basis wavelet denoising using besov projections. In *ICIP*, volume 1, pages 595–9, 1999.

[15] H. Choi and R. Baraniuk. Information-theoretic interpretation of besov spaces. In *SPIE*, August 2000.

[16] H. Choi and R.G. Baraniuk. Multiscale image segmentation using wavelet-domain hidden markov models. *IEEE Transactions on Image Processing*, 10(9):1309–1321, September 2001.

[17] M.S. Crouse, R.D. Nowak, and R.G. Baraniuk. Wavelet-based statistical signal processing using hidden markov models. *IEEE Transcations on Signal Processing*, 46(4):886–902, April 1998.

[18] I. Daubechies. *Ten lectures on wavelets*. SIAM, 1992.

[19] G. Van de Wouwer, P. Scheunders, and D. Van Dyck. Statistical texture characterisation from discrete wavelet representation. *IEEE IP*, 8(4):592–598, 1999.

[20] X. Descombes, R. Morris, and J. Zerubia. Some improvements to bayesian image segementation. part one: modelling. (in french). *Traitement du Signal*, 14(4):373–382, 1997.

[21] X. Descombes, R. Morris, and J. Zerubia. Some improvements to bayesian image segementation. part two: classification. (in french). *Traitement du Signal*, 14(4):383–395, 1997.

[22] D.L. Donoho and M. Johnstone. Adapting to unknown smoothness via wavelet shrinkage. *Journal of the American Statistical Association*, 90(432):1200–1224, December 1995.

[23] D. Dunn and W.E. Higgins. Optimal gabor filters for texture segmentation. *IEEE Transactions on Image Processing*, 4(7):947–964, July 1995.

[24] N. Fatemi-Ghomi, P.L. Palmer, and M. Petrou. Performance evaluation of texture segmentation algorithms based on wavelets. In *Workshop on Performance Characteristics of Vision Algorithms*, April 1996. Cambridge.

[25] R. Fjortoft, A. Lopes, P. Marthon, and E. Cubero-Castan. An optimal multiedge detector for sar image segmentation. *IEE Transactions on Geoscience and Remote Sensing*, 36(3):793–802, May 1998.

[26] J.Gomes and O.Faugeras. Reconciling distance functions and level sets. *Journal of Visual Communication and Image Representation*, 11:209–223, Avril 1999.

[27] Z. Kato. *Modélisation markoviennes multirésolutions en vision par ordinateur. Application à la segmentation d'images SPOT.* PhD thesis, Université de Nice Sophia Antipolis, 1994. (in French and English).

[28] S. Kichenassamy, A. Kumar, P. Olver, A. Tannenbaum, and A. Yezzi. Conformal curvature flows: From phase transitions to active vision. *Arch. Rational Mech. Anal.*, 134:275–301, 1996.

[29] A. Laine and J. Fan. Texture classification by wavelet packet signatures. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 11:1186–1190, 1993.

[30] S. Lakshmanan and H. Derin. Simultaneous parameter estimation and segmentation of gibbs random fields using simulated annealing. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 11:799–813, August 1989.

[31] L.C.Evans. *Partial Differential Equations*, volume 19 of *Graduate Studies in Mathematics*. American Mathematical Society, 1991.

[32] C. Lemarechal, R. Fjortoft, P. Marthon, E. Cubero-Castan, and A. Lopes. Sar image segmentation by morphlogical methods. In *Proc. SAR Image Analysis, Modelling, and Technique III*. SPIE, September 1998.

[33] D. Leporini and J.C. Pesquet. Bayesian wavelet denoising: Besov priors and non-gaussian noises. *Signal Processing*, 81:55–67, 2001.

[34] J. Liu and P. Moulin. Information-theoretic analysis of interscale and intrascale dependencies between image wavelet coefficients. *IEEE Transactions on Image Processing*, 10(11):1647–1658, November 2001.

[35] R. Malladi, J.A. Sethian, and B.C. Vemuri. Shape modeling with front propagation: A level set approach. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(2), February 1995.

[36] S.G. Mallat. Multiresolution approximations and wavelet orthonormal bases of $l^2(r)$. *Transactions of the American Mathematical Society*, 315(1):69–87, September 1989.

[37] S.G. Mallat. A theory for multiresolutionsignal decomposition: The wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(7):674–693, July 1989.

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

[39] B. Manjunath and R. Chellappa. Unsupervised texture segmentation using markov random fields models. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 13:478–482, May 1991.

[40] C. Menegaz, A. Rivoldini, and J.Ph. Thiran. Continuous directional dyadic wavelets as texture descriptors. In *SPIE*, pages 263–73, 2000.

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

[42] D. Mumford and J. Shah. Boundary detection by minimizing functionals. In *Proc. IEEE Conf. on Computer Vison and Pattern Recognition*. SPIE, 1985. San Francisco.

[43] S. Osher and R.P. Fedkiw. Level set methods: An overview and some recent results. *Journal of Computational Physics*, 169:463–502, 2001.

[44] N. Paragios and R. Deriche. Geodesic active regions and level set methods for supervised texture segmentation. *International Journal of Computer Vision*, 46(3), 2002.

[45] N. K. Paragios. *Geodesic Active Regions and Level Set methods: Contributions and Applications in Artificial Vision*. PhD thesis, University of Nice Sophia Antipolis, January 2000.

[46] D. Peng, B. Merriman, S. Osher, H. Zhao, and M. Kang. A pde based fast local level set method. *Journal of Computational Physics*, 155(2), November 1998.

[47] C. Samson. *Contribution à la classification d'images satellitaires par approche variationnelle et équations aux dérivées partielles*. PhD thesis, Université de Nice Sophia Antipolis, September 2000.

[48] C. Samson, L. Blanc-Feraud, G. Aubert, and J. Zerubia. A level set method for image classification. *IJCV*, 40(3):187–197, 2000.

[49] C. Samson, L. Blanc-Feraud, G. Aubert, and J. Zerubia. A variational model for image classification and restoration. *IEE Transactions on Pattern Analyis and Machine Intelligence*, 22(5):460–472, May 2000.

[50] P. Scheunders, S. Liven, G. Van de Wouwer, P. Vautrot, and D. Van Dyck. Wavelet-based texture analysis. *International Journal Computer Science and Information management*, December 1997.

[51] M. Sussman, P. Smereka, and S. Osher. A level set approach for computing solutions to incompressible two-phase flow. *Journal of Computational Physics*, 114:146–159, 1994.

[52] G. Unal, A. Yezzi, and H. Krim. Information-theoretic active polygons for unsupervised texture segmentation, June 2002. preprint available upon request, submitted.

[53] M. Unser. Texture classification and segmentation using wavelet frames. *IEEE Transactions on Image Processing*, 4(11):1549–1560, November 1995.

[54] T.Y. Young and K.S. Fu, editors. *Handbook of Pattern Recognition and Image Processing*. Norman G. Einspruch, 1986.

[55] Hong-Kai Zhao, T. Chan, B. Merriman, and S. Osher. A variational level set approach to multiphase motion. *Journal of Computational Physics*, 127:179–195, July 1996.

[56] Song Chu Zhu and A.Yuille. Region competition: unifying snakes, region growing, and bayes/mdl for multi-band image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(9), 1996.

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

Unité de recherche INRIA Lorraine : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex (France)
Unité de recherche INRIA Rennes : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)
Unité de recherche INRIA Rhône-Alpes : 655, avenue de l'Europe - 38330 Montbonnot-St-Martin (France)
Unité de recherche INRIA Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex (France)

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