Reconstructing an image from its edge representation

Suman Kumar Maji, Hussein Yahia, Hicham Badri

Reconstructing an image from its edge representation

Suman Kumar Maji[*], Hussein M. Yahia and Hicham Badri

Geostat Team (Geometry and Statistics in Acquisition Data) of INRIA,
200 rue de la Vieille Tour, 33405 Talence Cedex, FRANCE
Email: sumankumar.maji, hussein.yahia, hicham.badri@inria.fr

Abstract

In this paper, we show that a new edge detection scheme developed from the notion of transition in nonlinear physics, associated with the precise computation of its quantitative parameters (most notably singularity exponents) provide enhanced performances in terms of reconstruction of the whole image from its edge representation; moreover it is naturally robust to noise. The study of biological vision in mammals state the fact that major information in an image is encoded in its edges, the idea further supported by neurophysics. The first conclusion that can be drawn from this stated fact is that of being able to reconstruct accurately an image from the compact representation of its edge pixels. The paper focuses on how the idea of edge completion can be assessed quantitatively from the framework of reconstructible systems when evaluated in a microcanonical formulation; and how it redefines the adequation of edge as candidates for compact representation. In the process of doing so, we also propose an algorithm for image reconstruction from its edge feature and show that this new algorithm outperforms the well-known 'state-of-the-art' techniques, in terms of compact representation, in majority of the cases.

Keywords: Edge detection, statistical physics, critical exponents, compact representation, nonlinear signal processing, multifractals, wavelets, image reconstruction.

---

[*]Corresponding author. Tel.: (+33)(0)5 24 57 40 49.

1. Introduction

Edges are the primary features of natural images. These features are naturally associated to scale invariant properties, for example in the case of turbulent signals (flows) where symmetry is restored only in the statistical sense [22, 23]. Previous methods dedicated to the reconstructability of signals from their edge pixels lacked completeness in terms of compact representation and failed to provide a near estimate of the original signal [11]. The first theoretical instance of reconstructing a one-dimensional signal from its zero-crossing was found in the Logan theorem [2]. Logan proved that for a signal which is one-dimensional and strictly band-limited to a single octave, the time of the zero-crossings can form a complete representation of the signal and are sufficient candidates to reconstruct the signal. The findings of Logan were used in [3] to further investigate the possibility of complete representation of an image from the zero-crossings and gradient magnitudes of the image, convolved with a Laplacian of Gaussian (LoG) filter, at multiple scales. The proof of the theoretical completeness of the zero-crossing was further extended to the case of one-dimensional finite polynomial signal [5] and for a restricted class of band-limited two-dimensional signals [4].

It was acknowledged, however, in [2] that "the problem of actually recovering functions from their zero-crossing appears to be difficult (to say the least), under the most general conditions of uniqueness". The completeness of zero-crossings in representing an image feature and its ability to produce stable reconstructions, in practice, was further argued in [27]. In [24] an algorithm was introduced that computes an approximate reconstruction of an image from information coded at the image edges. The edges were computed in a manner quite similar to [3], but were further thresholded based on outputs from gradient based filters [11]. The method however lacked completeness to a large extent and was reintroduced in [25] by modifying the edge representation from a sub-sampled low-pass residual image [11]. In [26], it was proposed that images are well represented by the partial information confined to zero-crossings and a new reconstruction technique was proposed. The results, although better than the previous techniques [24, 25], lacked completion. A method based on minimizing equation error for stable reconstruction of image, from the restriction of its gradient measure over edges, was proposed in [27]. A new contour based image reconstruction technique was proposed in [28], by taking the edge as a Gaussian-blurred step discontinuity and considering a number of parameters, like luminance, brightness,

contrast, blur and 'contour width' [11], in the process of calculating them. In a seminal paper, Mallat and Zhong [29] described an algorithm to reconstruct images from their multiscale edges; the edges were detected by applying local maxima on a wavelet transformed version of the image. While this representation is far more compact in the case of 1-D signals, it misses certain details and some defocused structures are not recovered [11] in the case of images.

When an intensity image $I(x, y)$ is considered as a mathematical surface [9], edges can be detected as irregular distribution of intensity values over that surface. The idea of constructing complete surface specifications from the information contained in the zero-crossings was illustrated in [12]. In fact, a common surface reconstruction technique of regularization has been exploited in an attempt to reconstruct from contour line information [13], and in [14], a regularized fusion approach was applied to the problem of reconstruction from color edge maps.

In this work, the ideas of criticality and exponents, well defined in Statistical Physics and intimately related to predictability in complex systems [21], are associated to a computable notion of transition. We will show that these exponents, when evaluated in a microcanonical formulation, give rise to a notion of edge whose quantitative performance evaluation can be tested through the framework of *reconstructible systems*. We show that the new notion of edge can be used to produce near approximations of the original image. We show that 'state-of-the-art' surface reconstruction techniques, like fast Poisson solver [6], M-estimator, regularization [8] and diffusion [7], can be successfully applied to reconstruct images from their edge representation. We also propose an algorithm for image reconstruction from their edges and show that this new algorithm provides superior results, in terms of compact representation, over the well-known classical techniques.

The paper is organized as follows: In Section 2, we introduce the concept of singularity exponents and edge detection, in Section 3 we discuss the framework of reconstructible systems, where a brief discussion on the existing surface reconstruction techniques have been presented in Section 3.2. Quantitative results are shown and discussed in Section 4, followed by conclusion in Section 5.

2. Singularity Analysis and Edge detection

Natural images are characterized by their singularities. Singularities in an image correspond to a particular set of pixels where very high change in

the graylevel values are well recorded. In this approach, we try to estimate these singular values by a hierarchical classification, based on the strength of transition of the pixels around them. The first step concerns the definition of an appropriate multiscale measure.

Let I be a scalar image defined over a compact subset of $\mathbb{R}^2$ and $\|\nabla I\|$ is the norm of its gradient. We work with a recorded field of normalized luminance intensities i.e., $I(\vec{x}) - \langle I \rangle$, where $\langle I \rangle$ is the average of luminance intensities over the signal's domain. We then define a measure μ through its density $\mathrm{d}\mu(\vec{x})$, so that the measure of a ball $\mathcal{B}_r(\vec{x})$ of radius r centered around the point $\vec{x}$ corresponds to summing the norm of the gradient over $\mathcal{B}_r(\vec{x})$:

$$\mu(\mathcal{B}_r(\vec{x})) = \int_{\mathcal{B}_r(\vec{x})} \mathrm{d}(\vec{y}) \|\nabla I\|(\vec{y}) \tag{1}$$

A measure μ as the one defined above is a multiscale measure, in a microcanonical sense, if for any point $\vec{x} \in \Omega$ the following equality holds [17]:

$$\mu(\mathcal{B}_r(\vec{x})) = \alpha(\vec{x}) r^{h(\vec{x})} + o(r^{h(\vec{x})}) \; (\vec{r} \to 0) \tag{2}$$

where $\alpha(\vec{x})$ is a signal-dependent amplitude prefactor. The word "microcanonical" is used here w.r.t the relationship that can be contemplated between multifractal formalism and equilibrium studied in statistical physics [15]. The analogy can be considered from two approaches: The derivation of the singularity spectrum using the ensemble averages and moments [20], corresponding to the canonical approach in Statistical Physics, or using geometrically localized exponents corresponding to a microcanonical description [16]. A singularity spectrum, characteristic of a multiscale organization, can be computed in a coherent way by the two approaches [17]. We will discuss later in this article how localized singularity exponents can be computed in an efficient and robust manner. It should be noted that in the microcanonical formulation, to compute the measure $\mu(\mathcal{B}_r(\vec{x}))$, we do not make use of computationally large and physically inaccessible ensemble average of signals, as in the case of canonical formulations [17]. Instead, the average over different points $\vec{x}$ in the same realization $\mathcal{B}_r(\vec{x})$ are taken, making the approach computationally more efficient and less data demanding.

The exponent $h(\vec{x})$, which is a function of the point $\vec{x}$, is called the singularity exponent at point $\vec{x}$ [18], and quantifies the multiscale behavior of the measure μ [17]. In fact, decomposing a multiscale signal results in partitioning the signal domain into components $\mathcal{F}_h$, which are in general of fractal

4

nature. In other words, each point $\vec{x}$ in the signal is characterized by a singularity exponent $h(\vec{x})$ which is typical to one component $\mathcal{F}_h$. The fractal components are level sets of the function $h(\vec{x})$ [17] and are defined as follows:

$$\mathcal{F}_h = \{\vec{x} : h(\vec{x}) = h\} \tag{3}$$

This family of sets is naturally associated to the multiscale hierarchy in a signal and in the case of natural images, there exists a distinguished set of points, called the Most Singular Manifold (MSM), where the features of the system are well recorded [17]. The MSM points are the singularity components associated with the smallest possible value h_∞ and can be interpreted as the most informative set, from which the most relevant dynamical properties of the system can be extracted [17]. We will denote this set by $\mathcal{F}_\infty$ and it can be expressed as:

$$\mathcal{F}_\infty = \{\vec{x} : h(\vec{x}) = h_\infty = \min(h(\vec{x}))\} \tag{4}$$

The MSM plays a fundamental role in the multiscale geometrical hierarchy of natural images. Visual inspection of this set reveals a structure which is characterized by the presence of 'edges' or contours in natural images [10]. An example of the MSM set and its reconstruction using our proposed technique (see section 3.3) is shown in Fig 1.

The second step of the approach concerns the computation of singularity exponents with high numerical precision, since poor approximations of the exponents will lead to poor reconstruction results. In the next section, we give a brief overview of the different techniques that can be used for computing the singularity exponents.

2.1. Singularity Analysis

The singularity exponents for experimental, discretized data can be calculated using different methods. A very fast, but crude version of computing the singularity exponents is known as the Gradient histogram method [18]. It starts by taking into account the multiscale measure $\mu(\mathcal{B}_r(\vec{x}))$, defined by equation (1), as the gradient measurement for a discretized signal. Let us consider equation (2) at a minimal resolution r_0, where r_0 is chosen such that the whole image corresponds to size 1; in otherwords, if the image is an array of discretized values of size $m \times n$, one chooses $r_0 = \frac{1}{\sqrt{m \times n}}$ [33]. Following [18], equation (2) is then rewritten in the form:

$$\|\nabla I\|(\vec{x}) = \alpha_\mu(\vec{x}) r_0^{h(\vec{x})} \tag{5}$$
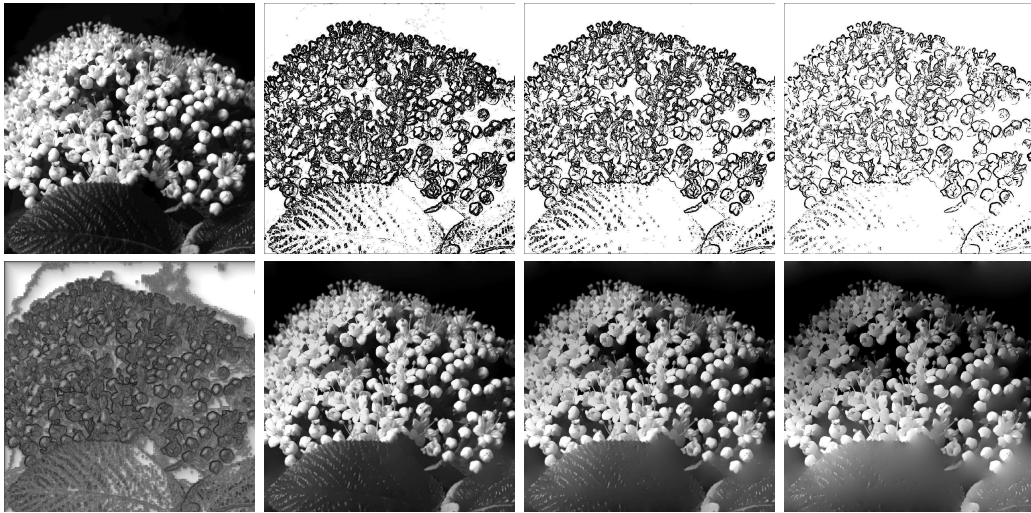
5

Figure 1: Flowers - image imk01305 of van Hateren database. **Top row**, from left to right: original image, compact representation of MSM points corresponding to 35 % pixel density, MSM points corresponding to 25 % pixel density, MSM points corresponding to 15 % pixel density. **Bottom row** (from left to right): visualisation of the singularity exponents, reconstruction with equation (23) using top rows respective MSM sets. The psnr's are 19.80 dB, 13.64 dB and 8.03 dB respectively.

Approximating the variables by statistical translational invariance [18] and by direct log-log regression, an estimate of $h(\vec{x})$ is obtained as:

$$\tilde{h}(\vec{x}) \equiv \frac{\log(\|\nabla I\|(\vec{x})/\langle \|\nabla I\| \rangle)}{\log r_0} \tag{6}$$

with the choice of α_μ as the average of the norm of the gradients. Although simple, this method is unstable and leads to poor reconstructions. For the experiments involved in this paper, we rely on a more precise and cost effective computation of the singularity exponents based on wavelet analysis of the measure. The method [33] makes good point estimates of the singularity exponent:

$$h(\vec{x}) = \frac{\log(\tau_\Psi \mu(\vec{x}, r_0))/\langle \tau_\Psi \mu(., r_0) \rangle}{\log r_0} + o(\frac{1}{\log r_0}) \tag{7}$$

where $\mathcal{T}_\psi \mu(\vec{x}, r_0)$ is a projection of measure μ evaluated at point $\vec{x}$ and scale r_0 with admissible wavelet ψ, $\langle \mathcal{T}_\psi \mu(\cdot, r_0) \rangle$ is the average value of the wavelet

projection over the whole signal. The scale r_0 is chosen such that the relative amplitude of the correction term $o(\frac{1}{\log r_0})$ is diminished. Sufficiently large values of I, at least of the order of 100×100 pixels, are therefore required to make the first term on the right hand side of the equation (7) a good approximation of the singularity exponent [33]. The procedure for calculating the MSM points, corresponding to the edge pixels of an image, is presented in algorithm 1.

Algorithm 1 Edge detection: Finding the MSM points

Step 1: Normalize an input image I as: $I(\vec{x}) - \langle I \rangle$.
Step 2: Compute the singularity exponents $h(\vec{x})$, at each point $\vec{x}$ on the image I, from equation (7).
Step 3: Determine the value of the most singular exponent h_∞ from the distribution of the singularity exponents $h(\vec{x})$. h_∞ is generally calculated as the average of the exponents corresponding to 1% and 5% quantiles.
Step 4: Define the density function $\delta_{\mathcal{F}_\infty}$ as: $\delta_{\mathcal{F}_\infty}(\vec{x}) = 1$ if $h(\vec{x}) \approx h_\infty$; $\delta_{\mathcal{F}_\infty}(\vec{x}) = 0$ otherwise.
Step 5: $\delta_{\mathcal{F}_\infty}(\vec{x})$ is a binary mask that locates the MSM points.

---

3. Framework of Reconstructible Systems

3.1. Problem Formulation

Let f_x, f_y denote a given non-integrable[1] gradient field over a $L \times B$ rectangular grid of image pixels. Given f_x, f_y, the goal is to obtain an image I, such that ∇I is "as close as" possible to the vector field (f_x, f_y). Let I_x, I_y denote the gradient field of I. Note that in our experiments f_x, f_y represents the density of the gradient measure corresponding to the MSM i.e., $f_x = I_x \delta_{\mathcal{F}_\infty}$ and $f_y = I_y \delta_{\mathcal{F}_\infty}$, where $\delta_{\mathcal{F}_\infty}$ stands for the standard density measure restricted to the set $\mathcal{F}_\infty$ i.e., $\delta_{\mathcal{F}_\infty}$ denotes the Dirac measures associated to the set $\mathcal{F}_\infty$. A common approach is to minimize the least square error function such that:

$$\underset{\text{I}}{\text{argmin}} \int \int ((I_x - f_x)^2 + (I_y - f_y)^2) dx dy \qquad (8)$$

---

[1]In the sense that the differential form $f_x dx + f_y dy$ is not supposed to be exact.

The associated Euler-Lagrange equation gives the Poisson equation:

$$\text{div}(I_x, I_y) = \text{div}(f_x, f_y) \qquad (9)$$

where 'div' refers to the divergence operator and is defined as $\text{div}(f_x, f_y) = \frac{\partial f_x}{\partial x} + \frac{\partial f_y}{\partial y}$. The aim of all the reconstruction algorithms is to find a solution for (I_x, I_y) of equation (9) which minimizes the error function in equation (8).

3.2. Linear systems

In this section, we try to summarize some existing techniques for reconstruction from a given gradient field. Readers are referred to [6, 8, 1] for a more detailed explanation of the algorithms.

3.2.1. Fast Poisson Solver

A well-known approach to solving the Poisson equation was proposed in [6]. The idea is to project the non-integrable gradient field on to a set of integrable slopes using discrete cosine functions. The Poisson equation can be written as $\frac{\partial^2 I}{\partial^2 x} + \frac{\partial^2 I}{\partial^2 y} = f(x, y)$, where $f(x, y) = \text{div}(f_x, f_y)$ is the divergence of the gradient field. An associated finite difference equation reads:

$$I_{j+1,l} + I_{j-1,l} + I_{j,l+1} + I_{j,l-1} - 4I_{j,l} = f_{j,l} \qquad (10)$$

where every coefficient can be expanded with the 2D discrete cosine transform as:

$$I_{j,l} = \frac{4}{JL} \sum_{m=0}^{J-1} \sum_{n=0}^{L-1} \hat{I}_{m,n} cos\frac{\pi jm}{J} cos\frac{\pi ln}{L} \qquad (11)$$

Substituting the values of the expansion coefficients in equation (10), we get the final solution as

$$\hat{I}_{m,n} = \frac{\hat{f}_{m,n}}{2(cos\frac{\pi m}{J} + cos\frac{\pi n}{L} - 2)} \qquad (12)$$

3.2.2. M-estimator

M-estimators is viewed as an iterative re-weighted least square solution [8] and can be written as

$$\underset{I}{\text{argmin}} \int \int (w(\epsilon_x^{k-1})(I_x - f_x)^2 + w(\epsilon_y^{k-1})(I_y - f_y)^2)dxdy \qquad (13)$$

8

where the weights $w_x = w(\epsilon_x^{k-1})$, $w_y = w(\epsilon_y^{k-1})$ at iteration k depend on the residual at iteration $k-1$. Applying Euler-Lagrange over equation (13) we get $\text{div}(w_x I_x, w_y I_y) = \text{div}(w_x f_x, w_y f_y)$. I can then be recovered by solving the linear equation $L_w I = f_w$, where $f_w = \text{div}(w_x f_x, w_y f_y)$ and L_w is the sparse Laplacian matrix of size $LB \times LB$.

3.2.3. Regularization

The L^2 regularization can be written as [8]:

$$\underset{\text{I}}{\text{argmin}} \int \int ((I_x - f_x)^2 + (I_y - f_y)^2 + \alpha(\phi(I_x) + \phi(I_y)))dxdy \qquad (14)$$

where α is called the regularization parameter using function ϕ. Applying Euler-Lagrange to equation (14) and after simplification, the solution can be achieved by iterative minimization [8].

3.2.4. Anisotropic diffusion

The anisotropic diffusion equation $I_t = \text{div}(c(x, y, t)\nabla I)$ of [1] was generalized in [7] as $I_t = \text{div}(c\nabla I)$, where c is the diffusion tensor and can be defined as a 2×2 symmetric, positive-definite matrix at each pixel. A generalized Poisson equation using c was proposed in [8] as

$$\text{div}(c \begin{bmatrix} I_x \\ I_y \end{bmatrix}) = \text{div}(c \begin{bmatrix} f_x \\ f_y \end{bmatrix}) \qquad (15)$$

Several methods have been proposed for obtaining the diffusion tensor c. The method followed in [8] suggests an edge preserving diffusion tensor at each pixel, by convolving component wise $\begin{bmatrix} f_x^2 & f_x \times f_y \\ f_x \times f_y & f_y^2 \end{bmatrix}$ with a Gaussian kernel. The final solution is given by $L_c I = f_c$, where $f_c = \text{div}(c \begin{bmatrix} f_x \\ f_y \end{bmatrix})$ and L_c is the Laplacian matrix.

3.3. Proposed approach: Reconstruction from MSM (R_{msm})

In this section, we propose a propagator that reconstructs the image $I(\vec{x})$ at every point $\vec{x}$ of the scene, given the gradient on MSM. We consider the gradient measure of the signal $\nabla I(\vec{x})$ and integrate it over the multifractal

set of most unpredictable points $\mathcal{F}_\infty$. A practical expression for the reconstruction formula is given by [10]:

$$I(\vec{x}) = \int \langle \vec{g}(\vec{x} - \vec{y}) | \nabla_\infty I(\vec{y}) \rangle \mathrm{d}\vec{y} = \vec{g} * \nabla_\infty I(\vec{x}) \tag{16}$$

where $\int_{\mathcal{F}_\infty} \mathrm{d}\vec{y}$ means integration over the MSM, $\vec{g}$ represents the desired propagator and $\langle \cdot | \cdot \rangle$ denotes the standard Hermitian product on $\mathbb{C}^2$. The essential gradient of the signal $\nabla_\infty I(\vec{x})$ is defined as the following distribution:

$$\nabla_\infty I(\vec{x}) = \nabla I(\vec{x}) \delta_{\mathcal{F}_\infty}(\vec{x}) \tag{17}$$

where $\delta_{\mathcal{F}_\infty}(\vec{x})$ is the density of the gradient measure restricted to the MSM. Accordingly, equation (16) can be expressed in the Fourier domain as:

$$\hat{I}(\vec{\omega}) = \langle \hat{\vec{g}}(\vec{\omega}) | \widehat{\nabla_\infty I}(\vec{\omega}) \rangle \tag{18}$$

where $\widehat{}$ represents the Fourier transform.

To derive an exact formulation for the reconstruction kernel $\hat{\vec{g}}$, we use an alternate approach to [10] and follow the derivation presented in [8] in the case of differentiable fields. In that context, one seeks a smooth vector field $\vec{f}$ defined over the whole image and which minimizes the L^2 distance with the original gradient $\nabla I(\vec{x})$ and can be expressed as

$$\underset{\mathrm{I}}{\mathrm{argmin}} \int \int \| \nabla I(\vec{x}) - \vec{f}(\vec{x}) \|_2^2 \, \mathrm{d}\vec{x} \tag{19}$$

We then follow the derivation explained in [8] to get a version of the reconstruction kernel $\hat{\vec{g}}$ in the smooth case. Taking the Euler-Lagrange variational formulation of equation (19) we get

$$\mathrm{div}(\nabla I)(\vec{x}) = \mathrm{div}(\vec{f})(\vec{x}) \tag{20}$$

Taking the Fourier transform of equation (20), we get as in [8]

$$\hat{I}(\vec{\omega}) = -i \frac{\omega_x \hat{f}_x(\vec{\omega}) + \omega_y \hat{f}_y(\vec{\omega})}{\omega_x^2 + \omega_y^2} \tag{21}$$

where the vector field $\vec{f}(\vec{x})$, after Fourier transformation gives rise to a complex vector field $\hat{\vec{f}} = (\hat{f}_x(\vec{\omega}), \hat{f}_y(\vec{\omega}))$. Comparing with equation (18), this suggest the kernel as

$$\hat{\vec{g}}(\vec{\omega}) = \frac{\vec{\omega}}{i \| \vec{\omega} \|^2} \tag{22}$$

10

and we have the final expression of the reconstruction formula over the MSM $\mathcal{F}_\infty$ i.e., R$_{\mathrm{msm}}$, in the Fourier domain, as:

$$\hat{I}(\vec{\omega}) = \frac{\langle \vec{\omega} | \widehat{\nabla_\infty I}(\vec{\omega}) \rangle}{i \|\vec{\omega}\|^2} \tag{23}$$

Fourier inversion of this formula gives the reconstruction of the image from the restriction of the gradient field to the MSM. In fact, $\delta_{\mathcal{F}_\infty}$ holds true for edges detected by any edge detector and can be incorporated likewise to create f_x, f_y and subsequently perform reconstruction. The algorithmic formulation of R$_{\mathrm{msm}}$ is presented in algorithm 2:

Algorithm 2 Reconstructing the signal from the MSM

Step 1: Calculate the singularity exponents $h(\vec{x})$ for every point $\vec{x}$ in the image.
Step 2: Determine h_∞.
Step 3: Define the density function $\delta_{\mathcal{F}_\infty}$ as the mask relative to the set $\mathcal{F}_\infty$.
Step 4: Calculate the field $\nabla_\infty I(\vec{x}) = \nabla I(\vec{x})\delta_{\mathcal{F}_\infty}(\vec{x})$ i.e., the values of the gradient over the MSM.
Step 5: Go to Fourier domain to obtain $\widehat{\nabla_\infty I}(\vec{\omega}) = (\widehat{\nabla_\infty I_x}(\vec{\omega}), \widehat{\nabla_\infty I_y}(\vec{\omega}))$.
Step 6: Calculate the scalar product $\langle \vec{\omega} | \widehat{\nabla_\infty I}(\vec{\omega}) \rangle = \omega_x \widehat{\nabla_\infty I_x}(\vec{\omega}) + \omega_y \widehat{\nabla_\infty I_y}(\vec{\omega})$.
Step 7: Calculate $\hat{I}(\vec{\omega}) = \frac{\langle \vec{\omega} | \widehat{\nabla_\infty I}(\vec{\omega}) \rangle}{i \|\vec{\omega}\|^2}$ with $\omega^2 = \omega_x^2 + \omega_y^2$.
Step 8: Do an inverse Fourier transform of $\hat{I}(\vec{\omega})$ to obtain $I(\vec{x})$.

---

4. Results

In this section, we discuss about the experiments performed. We perform two sets of experiments on two different sets of data.

- **Experiment 1**: We take the edge representation of different natural images (shown in Fig 2) given by their MSM points and reconstruct from them an approximation of the original image, using equation (23). We compare our reconstruction with the results of the reconstruction, over the MSM points, obtained from standard reconstruction techniques discussed in Section 3.2. The pixel density of the edges, for the experimental images, are kept the same (30%) while performing reconstruction. The results are shown in table 1. Similar experiment is

11

Figure 2: **Images used for experiment 1**. **Top row** (from left to right): Brick wall (SIPI image database), House (SIPI image database), imk03322 (van Hateren image database), Aerial view of a truck (SIPI image database), Julia Roberts (internet download), Sea Surface Temperature (SST) image of the Agulhas current below the coast of South Africa (SST data acquired by MODIS satellite on August 2, 2007). **Middle row**: visualisation of the singularity exponents. **Bottom row**: compact representation of MSM points corresponding to 30 % pixel density.

repeated over the images by adding different proportions of Gaussian white noise with respect to the standard deviation of the input images (results are shown in table 3).

- **Experiment 2**: We compare our edge detection technique with the classical edge detectors in terms of their reconstructibility. We choose R_{msm} and other reconstructors, already discussed in this article, for reconstructing test images (shown in Fig 3) from their edge pixels, and in the process check the performance of the reconstructors as well. The pixel density of the edges, calculated using different edge detectors, are kept within a close range (between 25 and 35%), with the pixel density of MSM points kept minimum. The classical edge algorithms tested are Matlab$^{©}$ implementations. Results are shown in table 4.

12
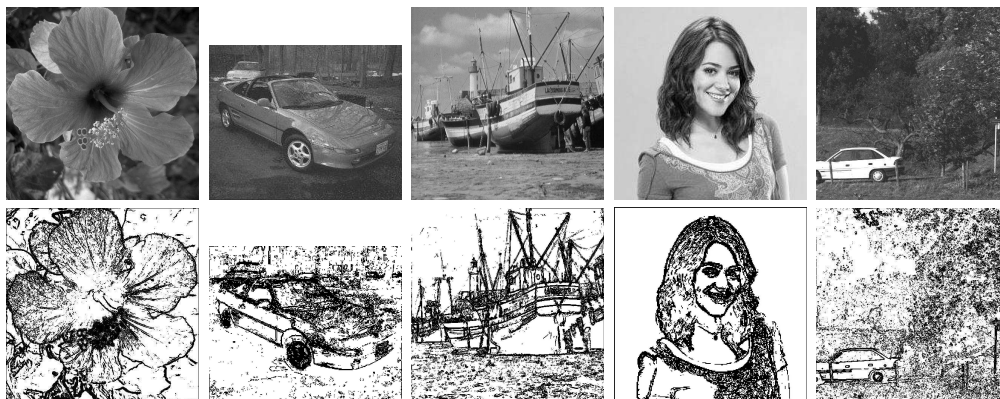
Figure 3: **Images used for experiment 2**. **Row 1** (from left to right): Hibiscus (SIPI image database), Car (CMU image database), Boat (SIPI image database), Camille (internet download), imk03324 (van Hateren image database). **Row 2**: MSM points corresponding to 30% pixel density.

4.1. Choice of images

Images of different entities are chosen trying to cover a broad spectrum of natural images starting from an object (brick wall, house, car) to landscape (imk03322, imk03324), aerial view (truck), face (Julia Roberts, Camille) and turbulent acquisitions of signals (sea surface temperature) as is shown in Fig 2 and Fig 3. These experimental images are chosen from standard databases like SIPI image database (brick wall, house, truck, boat, Hibiscus) [30], CMU image database (car) [31] and the van Hateren database (imk03322, imk03324) [19]. The sea surface temperature (SST) image is an acquisition by MODIS satellite of the Agulhas current below the coast of South Africa while the Julia Roberts image is a still image from a Hollywood movie. The edges in a turbulent signal, like the SST image, are difficult to determine as it is poorly characterized by the conventional Gaussian pre-filtered edge operators. These operators fail to study the transitions and the underlying multiscale dynamics of the system and henceforth leads to poor reconstruction results. The singularity exponents can well characterize the dynamical properties of a turbulent signal and the high-order transitions of such signals are recorded well in the MSM.

4.2. Quantitative analysis of the results

Visual quality of the reconstructed images, shown in table 1 and table 4, are evaluated based on the structural similarity index metric (SSIM) [32].

Table 1: **Performance of different reconstruction algorithms (Experiment 1)**. **Row 1**: Brick wall **Row 2**: House **Row 3**: imk03322 **Row 4**: Aerial view of a truck **Row 5**: Julia Roberts **Row 6**: SST image.

Original	R_{msm}	Poisson solver	Regularization	M-estimators	Diffusion
SSIM = 1	SSIM = 0.9922	SSIM = 0.9509	SSIM = 0.9182	SSIM = 0.9403	SSIM = 0.8895
SSIM = 1	SSIM = 0.9987	SSIM = 0.9954	SSIM = 0.9922	SSIM = 0.9951	SSIM = 0.9945
SSIM = 1	SSIM = 0.9986	SSIM = 0.9972	SSIM = 0.9964	SSIM = 0.9966	SSIM = 0.9962
SSIM = 1	SSIM = 0.9813	SSIM = 0.9519	SSIM = 0.9456	SSIM = 0.9530	SSIM = 0.9536
SSIM = 1	SSIM = 0.9901	SSIM = 0.9809	SSIM = 0.9707	SSIM = 0.9742	SSIM = 0.9783
SSIM = 1	SSIM = 0.9991	SSIM = 0.9989	SSIM = 0.9909	SSIM = 0.9990	SSIM = 0.9989

Table 2: **Quantitative analysis (Experiment 1):** Evaluation of the reconstruction algorithms, corresponding to Table 1, in terms of PSNR (in dB) and MSE.

Image		R_{msm}	Poisson solver	Regularization	M-estimators	Diffusion
Brick wall	PSNR	24.33	16.84	14.39	15.98	12.67
(512×512 pixels)	MSE	0.0909	0.5102	0.8975	0.6204	1.3284
House	PSNR	24.90	20.77	19.09	20.48	20.62
(256×256 pixels)	MSE	0.0183	0.0478	0.0733	0.0543	0.0493
imk03322	PSNR	30.63	27.72	26.75	27.05	26.75
(512×512 pixels)	MSE	0.0170	0.0329	0.0409	0.0382	0.0415
Aerial	PSNR	21.01	17.06	16.41	17.25	17.19
(512×512 pixels)	MSE	0.2209	0.4948	0.5682	0.4780	0.4661
Julia	PSNR	18.31	15.47	13.64	14.33	14.79
(256×256 pixels)	MSE	0.0934	0.1805	0.2758	0.2364	0.2068
Ocean	PSNR	25.10	24.82	16.25	24.61	24.58
(512×512 pixels)	MSE	0.0088	0.0110	0.0820	0.0114	0.0121

SSIM's for the reconstructed images show that R_{msm} outperforms the classical reconstruction algorithms in majority of the cases. Performance of the reconstruction under different levels of noise (SNR = 40 dB, 26 dB, 20 dB, 14 dB and 6 dB) is shown in table 3. We also illustrate the pertinence of the framework of reconstructible systems for evaluating an edge operator's compact representation effectiveness. In this work, the quality of a reconstruction is evaluated using the *mean square error* (MSE) and *peak signal to noise ratio* (PSNR, expressed in decibels dB). Table 2 and table 5 shows the performance evaluation of the reconstructed images using PSNR and MSE metrics and table 3 show the performance of the reconstruction under different levels of input SNR using the same metrics.

4.3. Computational complexity

The reconstruction technique used in algorithm 2 (R_{msm}) is essentially based on Fourier transform computation. We use fast Fourier transform (FFT) where the computational complexity is $N \log N$, N being the total number of pixels in the image. We use FFT only once in our operation and very fast implementations of FFT already exists e.g. Fastest Fourier Transform in the West (FFTW). Poisson solver is using the discrete cosine transform (DCT) instead of FFT. The other solvers are using one or more linear systems (e.g. M-estimator is using $7-8$ iterations) and the complexity

Table 3: **Quantitative analysis for noisy environment (Experiment 1)**: Evaluation of the reconstruction algorithms, in terms of PSNR and MSE, under different levels of noise.

Image	Algorithm	SNR = 40 dB MSE, PSNR	SNR = 26 dB MSE, PSNR	SNR = 20 dB MSE, PSNR	SNR = 14 dB MSE, PSNR	SNR = 6 dB MSE, PSNR
Brick wall	R_{msm}	0.0912, 24.26	0.0938, 24.19	0.0952, 23.98	0.1180, 23.28	0.2357, 21.46
	Poisson solver	0.5119, 16.79	0.5169, 16.77	0.5187, 16.69	0.5264, 16.54	0.5952, 16.27
	Regularization	0.8978, 14.31	0.8979, 14.26	0.8992, 14.18	0.9556, 13.22	0.9891, 12.84
	M-estimator	0.6213, 15.92	0.6254, 15.85	0.6343, 15.76	0.6408, 15.70	0.8261, 14.91
	Diffusion	1.3296, 12.64	1.3304, 12.62	1.3342, 12.59	1.4321, 12.48	1.4565, 12.35
House	R_{msm}	0.0183, 24.88	0.0207, 24.53	0.0269, 23.37	0.0475, 21.33	0.1723, 18.16
	Poisson solver	0.0472, 21.20	0.0513, 20.45	0.0607, 19.88	0.0805, 19.41	0.2057, 16.87
	Regularization	0.0745, 18.91	0.0767, 18.90	0.0775, 18.88	0.0889, 18.26	0.2057, 16.92
	M-estimator	0.0548, 20.32	0.0560, 20.05	0.0629, 19.68	0.0820, 19.50	0.2009, 17.68
	Diffusion	0.0497, 20.40	0.0517, 20.32	0.0590, 20.15	0.1106, 18.34	0.4236, 14.34
imk03322	R_{msm}	0.0171, 30.59	0.0196, 29.92	0.0248, 28.69	0.0478, 25.60	0.1780, 20.93
	Poisson solver	0.0331, 27.67	0.0337, 27.54	0.0425, 26.40	0.0687, 23.65	0.2023, 18.76
	Regularization	0.0411, 26.72	0.0442, 26.50	0.0452, 26.32	0.0674, 24.14	0.1918, 19.12
	M-estimator	0.0386, 26.98	0.0394, 26.87	0.0449, 26.28	0.0638, 24.74	0.1972, 17.57
	Diffusion	0.0419, 26.63	0.0445, 26.29	0.0557, 25.47	0.0871, 23.18	0.3066, 15.93
Aerial	R_{msm}	0.2210, 21.00	0.2278, 20.93	0.2311, 20.67	0.2577, 20.02	0.3862, 18.87
	Poisson solver	0.4960, 17.03	0.4962, 16.99	0.4966, 16.61	0.5029, 16.52	0.6099, 14.71
	Regularization	0.5684, 16.39	0.5696, 16.15	0.6011, 16.11	0.6607, 16.06	0.7021, 14.63
	M-estimator	0.4783, 17.03	0.4801, 16.98	0.4854, 16.84	0.4993, 16.38	0.5889, 15.77
	Diffusion	0.4669, 17.08	0.4774, 16.99	0.5046, 16.79	0.6174, 15.66	0.8075, 14.00
Julia	R_{msm}	0.0936, 18.18	0.0954, 18.02	0.1343, 17.93	0.1699, 17.10	0.2412, 16.34
	Poisson solver	0.1776, 15.37	0.1862, 15.30	0.2098, 15.11	0.2393, 14.98	0.4882, 13.76
	Regularization	0.2736, 13.54	0.2788, 13.37	0.2812, 13.11	0.3119, 12.43	0.3908, 11.94
	M-estimator	0.2367, 14.26	0.2578, 14.19	0.2619, 13.99	0.3323, 13.98	0.4940, 13.51
	Diffusion	0.2256, 14.38	0.2721, 14.21	0.3718, 13.54	0.3864, 12.92	0.7331, 12.65
Ocean	R_{msm}	0.0091, 25.08	0.0110, 24.86	0.0169, 23.23	0.0384, 21.42	0.1699, 17.54
	Poisson solver	0.0112, 24.68	0.0138, 23.84	0.0182, 23.26	0.0386, 20.68	0.1874, 17.61
	Regularization	0.0853, 16.09	0.0874, 15.83	0.0889, 15.30	0.1756, 14.05	0.1820, 12.39
	M-estimator	0.0115, 24.55	0.0128, 24.18	0.0182, 23.14	0.0499, 20.13	0.1701, 17.40
	Diffusion	0.0118, 24.49	0.0136, 24.13	0.0216, 22.38	0.0489, 20.37	0.3598, 14.21

depends on the solvers used. Multigrid solvers are the fastest solvers available till now, with the complexity $o(N)$, but still their fastest implementations are slow compared to the fastest implementations of FFT.

5. Conclusion

The ability to reconstruct an image from its edge representation lies in the efficiency of the edge detection algorithm as well as in the ability of the reconstruction algorithm to estimate an accurate approximation of the

Table 4: **Experiment 2:** Performance of different reconstructors over different edge detectors. Each row indicates the performance of the reconstructors for a given edge detection algorithm. Every column shows the edge detection algorithm that gives the best result, in terms of reconstruction, for a particular reconstructor.

	R_{msm}	Poisson solver	Regularization	M-estimator	Diffusion
MSM Density=25.00%	SSIM = 0.9942	SSIM = 0.9935	SSIM = 0.9904	SSIM = 0.9936	SSIM = 0.9908
NLFS [34] Density=25.14%	SSIM = 0.9938	SSIM = 0.9915	SSIM = 0.9863	SSIM = 0.9921	SSIM = 0.9577
Canny Density=28.29%	SSIM = 0.9638	SSIM = 0.9578	SSIM = 0.9567	SSIM = 0.9526	SSIM = 0.9445
LoG Density=31.26%	SSIM = 0.9750	SSIM = 0.8121	SSIM = 0.8134	SSIM = 0.8182	SSIM = 0.8093
Sobel Density=25.31%	SSIM = 0.9895	SSIM = 0.9808	SSIM = 0.9819	SSIM = 0.9871	SSIM = 0.9633
Prewitt Density=25.29%	SSIM = 0.9896	SSIM = 0.9681	SSIM = 0.9753	SSIM = 0.9859	SSIM = 0.9545

17

Table 5: **Quantitative analysis (Experiment 2)**: Data marked in blue indicates the reconstructor that performs the best, over other reconstruction techniques, for a particular edge detection algorithm. Data underlined indicates the edge detection algorithm that gives the best result, in terms of reconstruction, over all the other edge detectors, for a particular reconstructor.

Image	Algorithm	MSM	NLFS [34]	Canny	LoG	Sobel	Prewitt
		MSE, PSNR	MSE, PSNR	MSE, PSNR	MSE, PSNR	MSE, PSNR	MSE, PSNR
	R_{msm}	0.0906, 22.08	0.0944, 21.92	0.4005, 16.04	0.2657, 17.76	0.1690, 17.72	0.1622, 19.80
Hibiscus	Poisson solver	0.1042, 21.96	1.4791, 10.41	0.4889, 15.38	1.6979, 9.83	0.1571, 20.19	0.1443, 20.49
	Regularization	0.1530, 20.37	1.3846, 10.75	0.6270, 14.20	1.6545, 10.08	0.2285, 18.56	0.2270, 18.45
	M-estimator	0.1163, 21.42	1.2795, 11.12	0.7614, 13.42	1.5900, 10.09	0.1826, 19.52	0.1795, 19.61
	Diffusion	0.1168, 21.43	1.4393, 10.46	1.0162, 12.08	1.7405, 9.76	0.5453, 14.79	0.5062, 15.13
	R_{msm}	0.0794, 25.59	0.1321, 23.37	0.7350, 15.84	0.4038, 18.53	0.2150, 21.26	0.2262, 21.04
Car	Poisson solver	0.1033, 24.45	0.2330, 20.91	1.0848, 14.14	1.5238, 12.75	0.7679, 15.73	0.9510, 14.80
	Regularization	0.1617, 22.50	0.2547, 20.52	1.1642, 13.83	1.5087, 12.80	0.8179, 15.46	0.9629, 14.75
	M-estimator	0.1242, 23.65	0.2931, 19.92	1.2375, 13.57	1.4763, 12.89	0.4536, 18.02	0.5347, 17.30
	Diffusion	0.1095, 24.19	0.4530, 18.03	1.1917, 13.73	1.5260, 12.76	0.6453, 16.49	0.7295, 15.95
	R_{msm}	0.0969, 18.71	0.0997, 18.58	0.9636, 8.72	0.4145, 12.40	0.3296, 13.39	0.3708, 12.88
Boat	Poisson solver	0.1567, 16.62	0.2271, 15.01	0.7153, 10.01	1.2415, 7.64	0.7788, 9.66	0.7365, 9.90
	Regularization	0.2961, 13.86	0.3671, 12.93	0.9995, 8.56	1.2389, 7.65	0.9993, 8.58	0.9803, 8.66
	M-estimator	0.1466, 16.91	0.4923, 11.65	0.9272, 8.88	1.2606, 7.57	0.8205, 9.44	0.7814, 9.65
	Diffusion	0.1970, 15.63	0.4648, 11.90	0.9067, 8.98	1.3018, 7.43	0.4706, 11.85	0.5439, 11.22
	R_{msm}	0.0521, 15.60	0.0529, 15.53	0.3550, 7.26	0.2639, 8.56	0.1114, 12.19	0.1133, 12.23
Camille	Poisson solver	0.0743, 14.06	0.0792, 13.78	0.3927, 6.82	0.8539, 5.09	0.1808, 10.20	0.2888, 8.16
	Regularization	0.0852, 13.47	0.1412, 11.28	0.4094, 6.64	0.8301, 5.15	0.1771, 10.29	0.2388, 9.09
	M-estimator	0.0723, 14.18	0.0834, 13.56	0.4665, 6.07	0.7235, 5.41	0.1432, 11.21	0.1541, 10.89
	Diffusion	0.0910, 13.18	0.3865, 6.90	0.5704, 5.19	0.8237, 5.16	0.3831, 6.94	0.4681, 6.07
	R_{msm}	0.0854, 23.08	0.0960, 22.57	0.7808, 13.42	0.3168, 17.38	0.1634, 20.26	0.1705, 20.07
imk03324	Poisson solver	0.1456, 20.76	0.1887, 19.63	1.4146, 10.85	2.3025, 8.77	1.3966, 10.94	1.4172, 10.88
	Regularization	0.1848, 19.72	0.2285, 18.80	1.4535, 10.72	2.3019, 8.79	1.4782, 10.69	1.4859, 10.67
	M-estimator	0.2604, 18.24	0.3536, 17.35	1.4446, 10.75	2.3256, 8.73	1.4603, 10.75	1.4926, 10.65
	Diffusion	0.5899, 14.69	1.1471, 11.79	1.5041, 10.58	1.4650, 10.73	1.5532, 10.48	1.4189, 10.87

original image from the information coded in its edges. The efficiency of an edge detector lies not only in extracting features of real information from an image, but at the same time discarding redundant or perceptually irrelevant information. We show that critical exponents defined in Statistical Physics lead to a much more coherent definition of edges whose performance can be reviewed, in terms of compact representation, through the framework of reconstructible systems. The R_{msm} permits a quantitative evaluation of the

compactness of a representation, leading to highly accurate approximation of an original image from its edge pixels. Comparison with latest reconstruction techniques show that R_{msm} outperforms them on natural images and geophysical variables. Also, in Table 5, we can see that the performance of the reconstructors over the MSM points are the best (data underlined) compared to any other edge detector. For certain edge detectors, the combination with Poisson solver gives better results: for example, Hibiscus (with Sobel edge detector) and Boat (with Canny edge detector). However, they are always outperformed by the combination of MSM and R_{msm}.

Acknowledgment

[1] P. Perona, J. Malik: Scale-Space and Edge Detection Using Anisotropic Diffusion, *IEEE Trans. Pattern Anal. and Mach. Intelligence (PAMI)*, 12, 629-639, 1990.

[2] B. F. Logan: Information in the zero-crossings of bandpass signals, *Bell Syst. Tech. J.*, 56:487-510.

[3] D. Marr, E. Hildreth: Theory of edge detection, *Proc. R. Soc. Lond. B*, 207:187-217, 1980.

[4] S. Curtis, S. Shitz, A. Oppenheim: Reconstruction of nonperiodic two-dimensional signals from zero crossings, *IEEE Trans. Acoust., Speech, Signal Processing*, 35:890-893, 1987.

[5] A. Yuille, T. Poggio: Fingerprints theorems for zero crossings. *J. Opt. Soc. Am. A*, 2(5):683-692, 1985.

[6] A. S. Georghiades, P. N. Belhumeur, D. J. Kriegman: From few to many: Illumination cone models for face recognition under variable lighting and pose, *IEEE Trans. Pattern Anal. and Mach. Intelligence (PAMI)*, 23, 643-660, 2001.

[7] J. Weickert: Anisotropic Diffusion in Image Processing. PhD thesis, University of Kaiserslautern, Germany (2006).

[8] A. Agrawal, R. Raskar, R. Chellappa: What is the Range of Surface Reconstructions from a Gradient Field?, *European Conference on Computer Vision (ECCV)*, 2006.

[9] S. J. Wang, T. O. Binford: Generic, Model-Based Estimation and Detection of Discontinuities in Image Surfaces, *Proceedings of 1994 ARPA Image Understanding Workshop*, pp.1443-1449, 1994.

[10] A. Turiel, A. del. Pozo: Reconstructing images from their most singular fractal manifold, *IEEE Transactions Image Processing*, 2002, 11, 345-50.

[11] J. H. Elder: Are Edges Incomplete ?, *Int. J. Comput. Vis.*, 34, 97-122, 1999.

[12] W. E. L. Grimson: From Images to Surfaces: A Computational Study of the Human Early Visual System. *The MIT Press, Cambridge, Massachusetts*, 1981.

[13] S. Muraki, N. Yokoya, K. Yamamoto: Surface reconstruction from a contour line image by regularization, *Systems and Computers in Japan*, Volume 23, Issue 2, pages 81-91, 1992.

[14] R. A. Salinas, C. Richardson, M. A. Abidi, R. C. Gonzalez, Data Fusion: Color Edge Detection and Surface Reconstruction Through Regularization, *IEEE Trans. on Industrial Electronics*, vol. 43, no. 3, pp. 355-363, 1996.

[15] A. B. Chhabra, C. Meneveau, R. V. Jensen, K. R. Sreenivasan: Direct determination of the f(alpha) singularity spectrum and its application to fully developed turbulence, *Phys Rev A.*, 40(9), 5284-5294, 1989.

[16] A. Arneodo, E. Bacry, J. F. Muzy: The thermodynamics of fractals revisited with wavelets, *Physica A: Statistical and Theoretical Physics*, 213(1-2):232-275, 1995.

[17] A. Turiel, H. Yahia and C. Perez-Vicente: Microcanonical Multifractal Formalism: a geometrical approach to multifractal systems. Part I: singularity analysis. *Journal of Physics A: Mathematical and Theoretical, vol 41*, doi: 10.1088/1751-8113/41/1/015501.

[18] A. Turiel, C. Perez-Vicente, J. Grazzini: Numerical methods for the estimation of multifractal singularity spectra on sampled data: A comparative study. *Journal of Computational Physics*, Elsevier, 216, 2006, 362-390.

[19] J. H. van Hateren, A. van der Schaaf: Independent component filters of natural images compared with simple cells in primary visual cortex, *Proc.R. Soc. Lond., B265*, 359-366, 1998.

[20] A. Arneodo, F. Argoul, E.Bacry, J. Elezgaray, and J. F. Muzy: Ondelettes, multifractales et turbulence, *Diderot Editeur*, Paris, France, 1995.

[21] G. Bofetta, M. Cencini, M. Falcioni, A. Vulpiani: Predictability: a way to characterize Complexity, *Physics Reports 356*, 367-474, 2001.

[22] A. Turiel, N. Parga: The Multifractal Structure of Contrast Changes in Natural Images: From Sharp Edges to Textures, *Neural Computation*, 12, 763-793, 2000.

[23] U. Frisch: Turbulence, *Cambridge Univ. Press*, 1995, Cambridge MA.

[24] S. Carlsson: Sketch based image coding, *In Proc. of Premier Colloque Image*, Biarritz, France, pp. 71-77, 1984.

[25] S. Carlsson: Sketch based coding of grey level images, *Signal Processing*, 15:57-83, 1988.

[26] Y. Zeevi, D. Rotem: Image reconstruction from zero crossings, *IEEE Trans. Acoust., Speech, Signal Processing*, 34:1269-1277, 1986.

[27] R. Hummel, R. Moniot: Reconstructions from zero crossings in scale space, *IEEE Trans. on Acoustics, Speech, and Signal Processing*, 37(12):2111-2130, 1989.

[28] P. Grattoni, A. Guiducci: Contour Coding for Image Description, *Pattern Recognition Letters*, 11:95-105, 1990.

[29] S. Mallat, S. Zhong: Characterization of signals from multiscale edges, *IEEE Trans. Pattern Anal. Machine Intell.*, 14:710-732, 1992.

[30] USC-SIPI image database. http://sipi.usc.edu/services/database (1981).

[31] Carnegie Mellon image database. http://vasc.ri.cmu.edu/idb/.

[32] Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli: Image quality assessment: From error visibility to structural similarity, *IEEE Transactios on Image Processing*, vol. 13, no. 4, pp. 600-612, Apr. 2004.

[33] A. Turiel: Method and system for the singularity analysis of digital signals, patent registered under number pct/es2008/070195, 2008.

[34] O. Laligant, F. Truchetet: A Nonlinear Derivative Scheme Applied to Edge Detection, *IEEE PAMI*, 32, 242-257, 2010.