Curvature-Preserving Regularization of Multi-Valued Images using PDE's

David Tschumperlé

GREYC Image (CNRS UMR 6072), 6 Bd du Maréchal Juin, 14050 Caen Cedex, France
David.Tschumperle@greyc.ensicaen.fr

Abstract. We are interested in diffusion PDE's for smoothing multi-valued images in an anisotropic manner. By pointing out the pros and cons of existing tensor-driven regularization methods, we introduce a new constrained diffusion PDE that regularizes image data while taking curvatures of image structures into account. Our method has a direct link with a continuous formulation of the Line Integral Convolutions, allowing us to design a very fast and stable algorithm for its implementation. Besides, our smoothing scheme numerically performs with a sub-pixel accuracy and is then able to preserves very thin image structures contrary to classical PDE discretizations based on finite difference approximations. We illustrate our method with different applications on color images.

1 Introduction

Computing regularized versions of corrupted images has always been a desirable goal in the field of computer vision. It is useful, either to restore degraded images, or - more indirectly - as a pre-processing step that eases further data analysis. Since the pioneering work of Perona-Malik [21], the framework of anisotropic diffusion PDE's has particularly raised a strong interest for such a task : it has the ability to smooth data in a *nonlinear* way, allowing the preservation of significant image discontinuities. PDE's are local formulations which are well adapted to deal with degraded images containing local or semi-local data corruption sources : Gaussian noise, scratches or compression artifacts are local degradations usually encountered in digital images. Important historical steps in PDE-based image regularization have been reached with the extension of the classical heat flow to deal with anisotropic smoothing [21, 17, 25, 36], the interpretation of diffusion PDE's as gradient descents of energy functionals [2, 8, 10, 13, 24], and the link between regularization PDE's and the concept of non-linear scale spaces [1, 18, 19]. Extensions of these techniques have been more recently tackled to deal with general multi-valued images (including colors) [26, 31, 32, 37], fields of unit vectors [14, 20, 29], orthonormal matrices [11, 31], positive-definite matrices [11, 31], or image data defined on implicit surfaces [3, 9, 30]. Despite this wide range of existing formalisms, all regularization methods have something in common : they locally *smooth* the image along one or several directions of the space that are different at each image point. Typically, the principal smoothing direction is chosen to be parallel to the image contours, resulting in an *anisotropic* regularization that do not destroy edges. As a requirement, defining a correct *smoothing behavior* is one of the first aim of a good regularization algorithm, the second being the precision of the smoothing process itself : it must respect

the pre-defined smoothing geometry as much as possible. This general principle has been recently adopted in [32, 36] with the proposal of regularization PDE's designed to fit a given (user-defined) underlying local smoothing geometry, modeled as a field of diffusion tensors. Separating the smoothing geometry from the regularization itself allows to unify a lot of previously proposed equations into generic formulations, and generally provides a *local geometric interpretation* of the corresponding processes.
Here, we first propose a quick analysis of these unifying methods, then introduce a comparable tensor-driven diffusion PDE that regularizes multi-valued images *while respecting specific curvature constraints*. We show that this general formalism, which is naturally positioned between the two previous ones, has interesting smoothing properties. Moreover, it is directly related to the framework of LIC's (Line Integral Convolutions, firstly proposed by Cabral & Leedom [6]). This analogy leads to the proposal of an efficient LIC-based scheme that implements our proposed method. It allows the preservation of thin image structures, thanks to its sub-pixel accuracy and runs up to three times faster than classical explicit scheme thanks to its high stability. Results are finally illustrated with applications on color images, including denoising, inpainting and non-linear resizing.

2 Anisotropic Smoothing of Images with PDE's : A Review

2.1 Local geometry and diffusion tensors

We consider a noisy multi-valued image $\mathbf{I} : \Omega \rightarrow \mathbb{R}^n$ ($n = 3$ for color images), defined on $\Omega \subset \mathbb{R}^2$. I_i denotes the particular vector channel i of $\mathbf{I}$: $\mathbf{I}_{(\mathbf{X})} = \left(I_{1\,(\mathbf{X})}, ..., I_{n\,(\mathbf{X})} \right)^T$.
PDE-based regularizations act as local smoothers of $\mathbf{I}$ along defined directions depending themselves on the local configuration of the pixel intensities : one wants to smooth $\mathbf{I}$ mostly along directions of the edges if there are any. Naturally, this means we need first to retrieve the *local geometry* of $\mathbf{I}$. As pointed out in [12, 36], it may be seen as the definition of these important features at each image point $\mathbf{X} = (x, y) \in \Omega$:
• Two orthogonal directions $\theta^{+}_{(\mathbf{X})}\ ,\ \theta^{-}_{(\mathbf{X})} \in \mathrm{S}^1$ (unit vectors of $\mathbb{R}^2$) directed along the local maximum and minimum variations of image intensities at $\mathbf{X}$. The direction θ^- corresponds to the edge direction, when there is one.
• Two corresponding positive values $\lambda^{+}_{(\mathbf{X})}\ ,\ \lambda^{-}_{(\mathbf{X})}$ that measure the effective variations of image intensities (local signal contrast) along $\theta^{+}_{(\mathbf{X})}$ and $\theta^{-}_{(\mathbf{X})}$ respectively.
This geometry can be retrieved by the field $\mathbf{G}$ of *structure tensors*, which is a natural tensor-valued extension of the gradient field for multi-valued images [12] :

$$\forall \mathbf{X} \in \Omega, \quad \mathbf{G}_{(\mathbf{X})} = \textstyle\sum_{i=1}^{n} \nabla I_{i(\mathbf{X})} \nabla I^T_{i(\mathbf{X})} \qquad \text{where} \qquad \nabla I_i = \left(\frac{\partial I_i}{\partial x}\ \ \frac{\partial I_i}{\partial y} \right)^T \tag{1}$$

A Gaussian-smoothed version $\mathbf{G}_\sigma = \mathbf{G} * G_\sigma$ is usually computed to retrieve a more coherent geometry (the standard deviation σ being proportional to the *noise scale* [36]). Then, the spectral elements of $\mathbf{G}_{\sigma(\mathbf{X})}$ give at the same time the contrast (eigenvalues λ^-, λ^+) and the orientations (eigenvectors $\theta^- \perp \theta^+$) of the local image structures.
Once the local geometry $\mathbf{G}_\sigma$ of the image $\mathbf{I}$ has been determined, authors of [32, 36] propose to design a particular field $\mathbf{T} : \Omega \rightarrow \mathrm{P}(2)$ of *diffusion tensors* which specifies

the local smoothing geometry that *should drive the regularization process*. $\mathbf{T}$ naturally depends on the spectral elements λ^-, λ^+ and θ^-, θ^+ of $\mathbf{G}_\sigma$:

$$\forall \mathbf{X} \in \Omega, \qquad \mathbf{T}_{(\mathbf{X})} = f^-_{(\lambda^+,\lambda^-)}\ \theta^-\theta^- + f^+_{(\lambda^+,\lambda^-)}\ \theta^+{\theta^+}^T \tag{2}$$

Basically, the functions $f^{+/-} : \mathbb{R}^2 \rightarrow \mathbb{R}$ set the strengths of the desired smoothing along the corresponding directions $\theta^{+/-}$. Several choices for $f^{+/-}$ are possible, depending on the considered application. For image denoising, a possible choice is (proposed in [10, 31, 32]) : $f^{+/-}_{(\lambda_+,\lambda_-)} = \frac{1}{(1+\lambda^+ + \lambda^-)^{p_\pm}}$, with $p_- < p_+$.
Intuitively, if a pixel $\mathbf{X}$ is located on an image contour ($\lambda^+_{(\mathbf{X})}$ is high) then the smoothing must be performed mostly along the contour direction $\theta^-_{(\mathbf{X})}$ with a strength inversely proportional to the local contrast. Conversely, if $\mathbf{X}$ is located on a flat region ($\lambda^+_{(\mathbf{X})}$ is low), the smoothing must be performed in all possible directions (isotropic behavior), leading then to $\mathbf{T} \simeq \mathbb{I}_d$ (identity matrix). Modeling the local smoothing geometry as a field $\mathbf{T}$ of diffusion tensors is the first stage proposed both in [32, 36]. The desired smoothing must be applied then, using a possible choice of diffusion PDE's, as detailed below. Most existing regularization PDE's [1–4, 8, 10, 13, 18, 19, 21, 24–26] may be seen as particular cases of such diffusion equations with different tensor fields $\mathbf{T}$.

2.2 The divergence-based PDE

A corrupted multi-valued image $\mathbf{I} : \Omega \rightarrow \mathbb{R}^n$ can be anisotropically regularized "along" a diffusion tensor field $\mathbf{T} : \Omega \rightarrow \mathrm{P}(2)$ by the following divergence PDE :

$$\forall i = 1, .., n, \qquad \frac{\partial I_i}{\partial t} = \text{div}\left(\mathbf{T}\nabla I_i\right) \tag{3}$$

This tensor-driven regularization equation has been introduced in [36], and adapted for color/multi-valued images in [37]. Note that $\mathbf{T}$ is the same for all image channels I_i, ensuring that the I_i are smoothed along a *common multi-valued geometry*, contrary to an uncorrelated channel-by-channel approach. Despite its popularity, the PDE (3) *does not strictly respect the geometry* $\mathbf{T}$, since the smoothing performed is not always the one that could be expected. Particularly, consider the case of choosing
$\mathbf{T}_{1(\mathbf{X})} = \left(\frac{\nabla I}{\|\nabla I\|}\right)\left(\frac{\nabla I}{\|\nabla I\|}\right)^T$ or $\mathbf{T}_{2(\mathbf{X})} = \mathbb{I}_d$ (identity matrix). For scalar images, these different fields both lead to the well known *heat flow equation* $\frac{\partial I}{\partial t} = \Delta I$ that is equivalent to the convolution of the image I by a normalized Gaussian kernel (isotropic smoothing [15]), despite the pure anisotropic form of $\mathbf{T}_{1(\mathbf{X})}$. The divergence is indeed a differential operator which makes the PDE (3) implicitly depending on the spatial variations of $\mathbf{T}$. It is actually not conceivable to easily define a pointwise smoothing behavior $\mathbf{T}$ with a divergence equation (3).

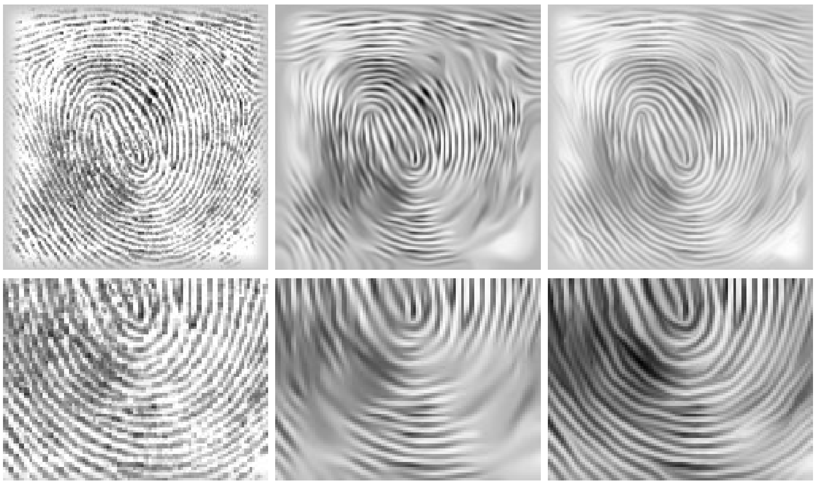
2.3 The trace-based PDE

In order to better respect the local smoothing geometry $\mathbf{T}$, we have proposed in [31, 32] a tensor-driven PDE, similar to (3), but based on a *trace* operator :

$$\forall i = 1, .., n, \qquad \frac{\partial I_i}{\partial t} = \text{trace}\left(\mathbf{T}\mathbf{H}_i\right) \tag{4}$$

$\mathbf{H}_i$ stands for the Hessian of I_i. As noticed in [31, 32], the evolution of (4) has an interesting geometric interpretation in terms of local image filtering with spatially varying short-time Gaussian kernels $G_{t\ (\mathbf{X})}^{\mathbf{T}} = \frac{1}{4\pi t}\exp\left(-\frac{\mathbf{X}^T\ \mathbf{T}^{-1}\ \mathbf{X}}{4t}\right)$, locally oriented by the tensor $\mathbf{T}_{(\mathbf{X})}$. It particularly ensures that the smoothing is truly done along the predefined smoothing geometry $\mathbf{T}$. As $trace()$ is not a differential operator, the spatial variation of $\mathbf{T}$ does not trouble the diffusion directions here and two differently shaped tensors necessarily lead to distinct smoothing behaviors.

Unfortunately, this analysis also points out one important drawback of the trace-based formulation. On curved structures (like corners), the Gaussian behavior of the smoothing is not desirable : when the local variation of the edge orientation θ^- is high, a Gaussian filter tends to *round* corners, since an oriented Gaussian kernel is not curved itself. This classical behavior is also best known as the "mean curvature flow" effect, characterized by the equation $\frac{\partial \mathbf{I}}{\partial t} = \frac{\partial^2 \mathbf{I}}{\partial \theta^{-2}}$. This is illustrated on Fig.1b where (4) has been applied on a real color image and $\mathbf{T}$ has been defined as (2) (then $f^- \neq 0$). Here, the mean curvature flow effect results in blending parallel thin curved structures. To avoid this over-smoothing, one usually try to vanish $f^{+/-}$ on curved structures (corners). But the detection of such structures on noisy images is a hard task. Conversely, image under-smoothing on edges may occur when one wants to limit the diffusion too much. There is a difficult trade-off between complete noise removal and preservation of curved structures, when using trace-based PDE's (4). This kind of regularization process does not care about the *curvature* of the smoothing directions, and by extension, of the curvature of the image contours. Taking this curvature into account is a very desirable goal and has motivated the work presented in this paper. For illustration purposes, results of our proposed curvature-preserving equation is shown on Fig.1c.



(a) Image of a fingerprint

(b) Applying trace-based PDE (4), with $p_1 = 0.5, p_2 = 1.2$.

(c) Applying our constrained PDE (11), with $p1 = 0.5, p_2 = 1.2$.

Fig. 1. Comparisons between trace PDE (4) and our proposed curvature-preserving PDE's (11).

3 The Framework of Curvature-Preserving PDE's

3.1 The single direction case

We introduce now the general idea of curvature-preserving PDE's, focusing first on image regularization along a *vector field* $\mathbf{w} : \Omega \rightarrow \mathbb{R}^2$ instead of a tensor field $\mathbf{T}$. We consider then a local smoothing everywhere along a single varying direction $\frac{\mathbf{w}}{\|\mathbf{w}\|}$, with a strength $\|\mathbf{w}\|$. The two spatial components of $\mathbf{w}$ are denoted by $\mathbf{w}_{(\mathbf{X})} = (u_{(\mathbf{X})} \;\; v_{(\mathbf{X})})^T$. We define the *curvature-preserving* regularization PDE that smoothes $\mathbf{I}$ along $\mathbf{w}$ by :

$$\forall i = 1, \ldots, n, \qquad \frac{\partial I_i}{\partial t} = \text{trace}\left(\mathbf{w}\mathbf{w}^T\ \mathbf{H}_i\right) + \nabla I_i^T \mathbf{J}_{\mathbf{w}} \mathbf{w} \tag{5}$$

where $\mathbf{J}_{\mathbf{w}}$ stands for the Jacobian of $\mathbf{w}$, and $\mathbf{H}_i$ the Hessian of I_i.
Let us study more closely how (5) is related to $\mathbf{w}$. We consider the curve $\mathcal{C}^{\mathbf{X}}_{(a)}$ defining the *integral curve* of $\mathbf{w}$, starting from $\mathbf{X}$ and parameterized by $a \in \mathbb{R}$ (Fig.2a) :

$$\mathcal{C}^{\mathbf{X}}_{(0)} = \mathbf{X} \qquad \text{and} \qquad \frac{\partial \mathcal{C}^{\mathbf{X}}_{(a)}}{\partial a} = \mathbf{w}(\mathcal{C}^{\mathbf{X}}_{(a)}) \tag{6}$$

We denote by $\mathcal{F}$ the family of integral curves of $\mathbf{w}$. A second-order Taylor development of $\mathcal{C}^{\mathbf{X}}_{(a)}$ around $a = 0$ is $\mathcal{C}^{\mathbf{X}}_{(h)} = \mathcal{C}^{\mathbf{X}}_{(0)} + h \frac{\partial \mathcal{C}^{\mathbf{X}}_{(a)}}{\partial a}_{|a=0} + \frac{h^2}{2} \frac{\partial^2 \mathcal{C}^{\mathbf{X}}_{(a)}}{\partial a^2}_{|a=0} + O(h^3)$, i.e :

$$\mathcal{C}^{\mathbf{X}}_{(h)} = \mathbf{X} + h\mathbf{w}_{(\mathbf{X})} + \frac{h^2}{2}\ \mathbf{J}_{\mathbf{w}_{(\mathbf{X})}} \mathbf{w}_{(\mathbf{X})} + O(h^3)$$

with $h \rightarrow 0$, and $O(h^n) = h^n\ \epsilon_n$. Then, we can compute a second-order Taylor development of $I_i(\mathcal{C}^{\mathbf{X}}_{(a)})$ around $a = 0$, which corresponds to the variations of the image intensity near $\mathbf{X}$ when following the integral curve $\mathcal{C}^{\mathbf{X}}$:

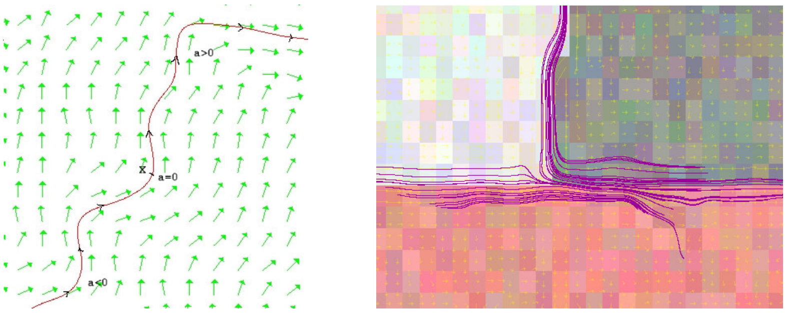
$$I_i(\mathcal{C}^{\mathbf{X}}_{(h)}) = I_{i(\mathbf{X})} + h \nabla I_{i(\mathbf{X})}^T\ (\mathbf{w}_{(\mathbf{X})} + \frac{h}{2}\ \mathbf{J}_{\mathbf{w}_{(\mathbf{X})}} \mathbf{w}_{(\mathbf{X})}) + \frac{h^2}{2} \text{trace}\left(\mathbf{w}_{(\mathbf{X})} \mathbf{w}_{(\mathbf{X})}^T \mathbf{H}_i\right) + O(h^3)$$

The second derivative of the function $a \rightarrow I_i(\mathcal{C}^{\mathbf{X}}_{(a)})$ at $a = 0$ is then :

$$\begin{aligned} \frac{\partial^2 I_i(\mathcal{C}^{\mathbf{X}}_{(a)})}{\partial a^2}_{|a=0} &= \lim_{h \rightarrow 0}\ \frac{1}{h^2} \left[I_i(\mathcal{C}^{\mathbf{X}}_{(h)}) + I_i(\mathcal{C}^{\mathbf{X}}_{(-h)}) - 2 I_i(\mathcal{C}^{\mathbf{X}}_{(0)}) \right] \\ &= \text{trace}\left(\mathbf{w}_{(\mathbf{X})} \mathbf{w}_{(\mathbf{X})}^T \mathbf{H}_{i(\mathbf{X})}\right) + \nabla I_i^T \mathbf{J}_{\mathbf{w}_{(\mathbf{X})}} \mathbf{w}_{(\mathbf{X})} \end{aligned} \tag{7}$$

This is exactly the right term in our curvature-preserving PDE (5). Actually, (5) can be seen individually for all integral curves of $\mathcal{F}$ instead of each point $\mathbf{X} \in \Omega$: consider another point $\mathbf{Y} \in \mathcal{C}^{\mathbf{X}}$. Then, there exist $\epsilon \in \mathbb{R}$ such that $\mathbf{Y} = \mathcal{C}^{\mathbf{X}}_{(\epsilon)}$. Indeed, $\mathcal{C}^{\mathbf{X}}$ and $\mathcal{C}^{\mathbf{Y}}$ describe the same curve (6) with different parameterizations : $\forall a \in \mathbb{R},\ \ \mathcal{C}^{\mathbf{Y}}_{(a)} = \mathcal{C}^{\mathbf{X}}_{(\epsilon+a)}$. As (5) is verified on $\mathbf{Y}$, then $\frac{\partial I_i(\mathcal{C}^{\mathbf{X}}_{(a)})}{\partial t}_{|a=\epsilon} = \frac{\partial^2 I_i(\mathcal{C}^{\mathbf{X}}_{(a)})}{\partial a^2}_{|a=\epsilon}$. This is obviously true for $\epsilon \in \mathbb{R}$ since (5) is verified for all points $\mathbf{Y}$ lying on the integral curve $\mathcal{C}^{\mathbf{X}}$. Then, the PDE (5) may be also written as : $\forall \mathcal{C} \in \mathcal{F},\ \ \forall a \in \mathbb{R},\ \ \frac{\partial I_i(\mathcal{C}_{(a)})}{\partial t} = \frac{\partial^2 I_i(\mathcal{C}_{(a)})}{\partial a^2}$.

We recognize thus a *one-dimensional heat flow constrained on* $\mathcal{C}$. This is very different from a heat-flow *oriented* by $\mathbf{w}$, as in $\frac{\partial I_i}{\partial t} = \frac{\partial^2 I_i}{\partial \mathbf{w}^2}$ since the curvatures of integral curves of $\mathbf{w}$ are now implicitly taken into account. In particular, the velocity of our constrained equation has the interesting property to vanish when image intensities are locally constant on $\mathcal{C}$, whatever the curvature of $\mathcal{C}$ is. In this context, defining a field $\mathbf{w}$ that is tangent everywhere to the image structures allows the preservation of these structures, even if they are curved (this concerns corners particularly, Fig.2b and Fig.1c). This is not the case with divergence (3) or trace-based PDE's (4).



(a) Integral curve of a vector field $\mathbf{w}$.

(b) Example of integral curves $\mathcal{C}^{\mathbf{X}}_{(a)}$ when $\mathbf{w}$ is the lowest eigenvector of the structure tensor $\mathbf{G}_\sigma$ of a color image $\mathbf{I}$.

Fig. 2. Integral curve $\mathcal{C}^{\mathbf{X}}$ of vector fields $\mathbf{w} : \Omega \rightarrow \mathbb{R}^2$.

The existence and unicity of the solutions of (5) are not directly approached in this article, although we show below that these solutions can be approximated by the line integral convolution technique, which is a well-posed analytical approach [6].

3.2 Curvature-preserving PDE's and line integral convolutions

Line Integral Convolutions (LIC) have been first introduced in [6] as a technique to create a textured image $\mathbf{I}^{LIC}$ that represents a vector field $\mathbf{w} : \Omega \rightarrow \mathbb{R}^2$. The idea consists in smoothing an image $\mathbf{I}^{noise}$ - containing only noise - by averaging its pixel values along the integral curves of $\mathbf{w}$. Actually, a continuous formulation of a LIC is :

$$\forall \mathbf{X} \in \Omega, \qquad \mathbf{I}^{LIC}_{(\mathbf{X})} = \frac{1}{N} \int_{-\infty}^{+\infty} f(p)\, \mathbf{I}^{noise}(\mathcal{C}^{\mathbf{X}}_{(p)})\, dp \tag{8}$$

where $f : \mathbb{R} \rightarrow \mathbb{R}$ is an even function (strictly decreasing to 0 on $\mathbb{R}^+$) and $\mathcal{C}^{\mathbf{X}}$ is defined as the *integral curve* (6) of $\mathbf{w}$ through $\mathbf{X}$. The normalization factor N allows the preservation of the average pixel value along $\mathcal{C}^{\mathbf{X}}$ and is equal to $N = \int_{-\infty}^{+\infty} f(p)\, dp$. As noticed in previous section, our curvature-preserving PDE (5) can be seen as the one-dimensional heat flow $\frac{\partial I_i(\mathcal{C}_{(a)})}{\partial t} = \frac{\partial^2 I_i(\mathcal{C}_{(a)})}{\partial a^2}$ constrained on the integral curve $\mathcal{C}^{\mathbf{X}} \in \mathcal{F}$. Using the variable substitution $\mathbf{L}_{(a)} = \mathbf{I}(\mathcal{C}^{\mathbf{X}}_{(a)})$, this PDE becomes $\frac{\partial \mathbf{L}}{\partial t}(a) = \mathbf{L}''_{(a)}$. The solution $\mathbf{L}^{[t]}$ at time t is known to be the convolution of $\mathbf{L}^{[t=0]}$ by a normalized Gaussian kernel G_t (see [15]) : $\mathbf{L}^{[t]}_{(a)} = \int_{-\infty}^{+\infty} \mathbf{L}^{[t=0]}_{(p)}\, G_{t(a-p)}\, dp$ with $G_{t(p)} = \frac{1}{\sqrt{4\pi t}} \exp\left(-\frac{p^2}{4t}\right)$.

Substituting $\mathbf{L}$ with $a = 0$, and remembering that $\mathcal{C}^{\mathbf{X}}_{(0)} = \mathbf{X}$ and $G_{t(-p)} = G_{t(p)}$:

$$\forall \mathbf{X} \in \Omega, \qquad \mathbf{I}^{[t]}_{(\mathbf{X})} = \int_{-\infty}^{+\infty} \mathbf{I}^{[t=0]}(\mathcal{C}^{\mathbf{X}}_{(p)})\; G_{t(p)}\; dp \tag{9}$$

Thus, the equation (9) is a particular form of the continuous LIC-based formulation (8) with a Gaussian weighting function $f = G_t$. Here, the normalization factor is $N = \int_{-\infty}^{+\infty} G_{t(p)}\; dp = 1$. Intuitively, the evolution of our curvature-preserving PDE (5) may be seen as the application of local convolutions by normalized one-dimensional Gaussian kernels *along integral curves* $\mathcal{C}$ of $\mathbf{w}$, which is a possibly *curved* filtering instead of an oriented one. Applying this setting on a multi-valued image $\mathbf{I}$, with $\mathbf{w}$ being the lowest eigenvector of the structure tensor field $\mathbf{G}$ (i.e. the contour direction) allows the preservation of curved image structures. This is illustrated on Fig.2b, where few integral lines $\mathcal{C}^{\mathbf{X}}$ are shown, around a typical T-junction structure. Note how the streamlines rotate when arriving at the junction.
Note that (9) is an analytical solution of (5) when $\mathbf{w}$ *does not evolve over time*. This is generally not true when dealing with general nonlinear regularization PDE's, where the smoothing geometry is re-evaluated at each time step. We can anyway perform several iterations of our LIC scheme (9), where the vector field $\mathbf{w}$ is updated at each iteration, exactly as it is done in explicit PDE implementations, where the smoothing geometry $\mathbf{w}$ is considered as constant between two successive time steps t and $t + dt$.

3.3 Between traces and divergence formulations

It is worth to notice than our curvature-preserving PDE (5) is naturally positioned between the trace and divergence formulations. We can express div $\left(\mathbf{w}\mathbf{w}^T\; \nabla I_i\right)$ as

$$\text{div}\left(\mathbf{w}\mathbf{w}^T\; \nabla I_i\right) = \text{trace}\left(\mathbf{w}\mathbf{w}^T \mathbf{H}_i\right) + \nabla I_i^T \mathbf{J}_{\mathbf{w}} \mathbf{w} + \text{div}(\mathbf{w}) \nabla I_i^T \mathbf{w}$$

The first term corresponds to the trace PDE (4) (that smoothes locally $\mathbf{I}$ along $\mathbf{w}$), the two first terms correspond to our *curvature-constrained* regularization PDE (5), (that smoothes locally $\mathbf{I}$ along $\mathbf{w}$ while taking the curvature of integral curves $\mathcal{C}$ of $\mathbf{w}$ into account), and the three terms together correspond to the classical divergence PDE (3) that performs local diffusions of $\mathbf{I}$ along $\mathbf{w}$. In our point of view, the last term div$(\mathbf{w})\nabla I_i^T \mathbf{w}$ is responsible for the perturbations of the effective smoothing direction (as described in section 2.2) and is not desirable for image regularization purposes. Our proposed curvature-constrained PDE (5) allows at the same time the full respect of the pre-defined smoothing directions $\mathbf{w}$, while preserving images structures which are curved along $\mathbf{w}$. Note also that we can also see our curvature-preserving PDE (5) as the corresponding divergence-based equation minus the term div$(\mathbf{w})\nabla I_i^T \mathbf{w}$. Thus, where $\mathbf{w}$ is a divergence-free field, the divergence and curvature-preserving approaches are strictly equivalent.

3.4 Extension to multi-directional smoothing

Here, we extend our single-direction smoothing PDE (5) so that it can deal with a tensor-valued geometry $\mathbf{T} : \Omega \rightarrow \mathrm{P}(2)$, instead of a single vector-valued geometry $\mathbf{w}$. Indeed, a diffusion tensor can describe much more complex smoothing behaviors (isotropic and anisotropic) than single directions (only anisotropic). This extension is not straightforward : curvatures and integral curves of tensor-valued fields

$\mathbf{T}$ are not easily defined. Instead, we propose to locally decompose a tensor-driven smoothing process by several vector-driven smoothing processes along different orientations. We first notice that $\int_{\alpha=0}^{\pi} a_\alpha a_\alpha^T \, d\alpha = \frac{\pi}{2} \, \mathbb{I}_d$ with $a_\alpha = \left(\cos\alpha \quad \sin\alpha\right)^T$. Then, any 2×2 tensor $\mathbf{T}$ may be written as : $\mathbf{T} = \frac{2}{\pi} \sqrt{\mathbf{T}} \left(\int_{\alpha=0}^{\pi} a_\alpha a_\alpha^T \, d\alpha \right) \sqrt{\mathbf{T}}$ where $\sqrt{\mathbf{T}} = \sqrt{f^+}\mathbf{u}\mathbf{u}^T + \sqrt{f^-}\mathbf{v}\mathbf{v}^T$ stands for the square root of $\mathbf{T}$. One can easily verify that $(\sqrt{\mathbf{T}})^2 = \mathbf{T}$ and $(\sqrt{\mathbf{T}})^T = \sqrt{\mathbf{T}}$. Thus, the tensor $\mathbf{T}$ may be written as :

$$\mathbf{T} = \frac{2}{\pi} \int_{\alpha=0}^{\pi} (\sqrt{\mathbf{T}} a_\alpha)(\sqrt{\mathbf{T}} a_\alpha)^T \, d\alpha \tag{10}$$

We have split the tensor $\mathbf{T}$ into a sum of *atomic* tensors $(\sqrt{\mathbf{T}} a_\alpha)(\sqrt{\mathbf{T}} a_\alpha)^T$, each being purely anisotropic and directed only along the vector $\sqrt{\mathbf{T}} a_\alpha \in \mathbb{R}^2$. The equation (10) naturally suggests to decompose any tensor-driven regularization PDE into a sum of single direction smoothing processes, each of them respecting the overall geometry $\mathbf{T}$. For instance, if $\mathbf{T} = \mathbb{I}_d$ (identity matrix), the tensor is isotropic and $\forall \alpha \in [0, \pi], \ \sqrt{\mathbf{T}} a_\alpha = a_\alpha$. The resulting smoothing will be then performed in all directions a_α of the plane with the same strength, while if $\mathbf{T} = \mathbf{u}\mathbf{u}^T$ (where $\mathbf{u} \in \mathrm{S}^1$), the tensor is purely anisotropic and : $\forall \alpha \in [0, \pi], \ \sqrt{\mathbf{T}} a_\alpha = (\mathbf{u}^T a_\alpha)\mathbf{u}$. The resulting smoothing will be then performed only along the direction $\mathbf{u}$ of the tensor $\mathbf{T}$.
Then, considering that each single direction smoothing must be done with a curvature-preserving PDE (5), we define the following constrained regularization PDE, acting on a multi-valued image $\mathbf{I}$ and driven by a tensor-valued smoothing geometry $\mathbf{T}$:
$\frac{\partial I_i}{\partial t} = \frac{2}{\pi} \int_{\alpha=0}^{\pi} \mathrm{trace}\left((\sqrt{\mathbf{T}} a_\alpha)(\sqrt{\mathbf{T}} a_\alpha)^T \mathbf{H}_i \right) + \nabla I_i^T \, \mathbf{J}_{\sqrt{\mathbf{T}} a_\alpha} \sqrt{\mathbf{T}} a_\alpha \, d\alpha,$
which can be simplified as :

$$\forall i = 1, \ldots, n, \qquad \frac{\partial I_i}{\partial t} = \mathrm{trace}(\mathbf{T}\mathbf{H}_i) + \frac{2}{\pi} \nabla I_i^T \int_{\alpha=0}^{\pi} \mathbf{J}_{\sqrt{\mathbf{T}} a_\alpha} \sqrt{\mathbf{T}} a_\alpha \, d\alpha \tag{11}$$

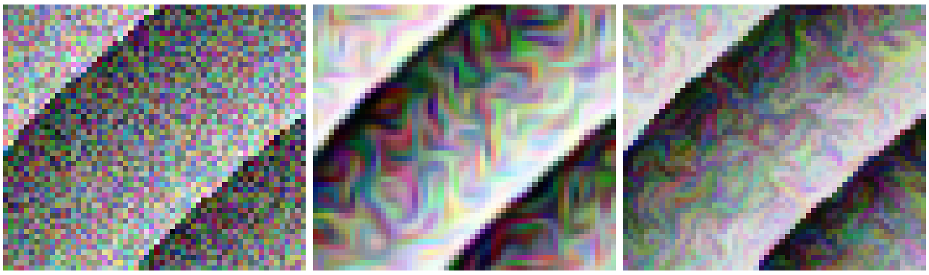
where $a_\alpha = (\cos\alpha \quad \sin\alpha)^T$, and $\mathbf{J}_{\sqrt{\mathbf{T}} a_\alpha}$ stands for the Jacobian of the vector field $\Omega \to \sqrt{\mathbf{T}} a_\alpha$. A similar idea of smoothing decomposition along all orientations of the plane can be also found in [35]. As in the single direction case, (11) may be seen as a trace-based equation (4), where an extra term has been added in order to respect the curvature of all integral lines passing through the tensor-valued geometry $\mathbf{T}$.

4 Implementation and Applications

The implementation of our regularization method (11) benefits from the LIC-based interpretation of curvature-preserving PDE's presented in section 3.2. Indeed, we can explicitly discretize (11) by the Euler scheme : $\mathbf{I}^{[t+dt]} = \mathbf{I}^{[t]} + \frac{2dt}{N} \left(\sum_{k=0}^{N-1} \mathcal{R}(\sqrt{\mathbf{T}} a_\alpha) \right)$ where $\alpha = k\pi/N$ (in the interval $[0, \pi]$), dt is the usual temporal discretization step and $\mathcal{R}(\mathbf{w})$ represents a discretization of the curvature-preserving PDE (5) that preserves curvatures along the single direction $\mathbf{w}$. If we write this expression as : $\mathbf{I}^{[t+dt]} = \frac{1}{N} \left(\sum_{k=0}^{N-1} \mathbf{I}^{[t]} + 2dt \, \mathcal{R}(\sqrt{\mathbf{T}} a_\alpha) \right)$, we can express it as the averaging of different LIC's along vector fields $\sqrt{\mathbf{T}} a_\alpha$:

$$\mathbf{I}^{[t+dt]} = \frac{1}{N} \left(\sum_{k=0}^{N-1} \mathbf{I}^{[t]}_{LIC(\sqrt{\mathbf{T}} a_\alpha)} \right)$$

The only difficult part here is the LIC implementation which needs the computation of integral curves. A classical second-order Runge-Kutta integration [23] has been used with success for our implementation. On one hand, our scheme allows the preservation of thin image structures from a numerical point of view : the smoothing is performed along integral curves of $\mathbf{w}$, with a sub-pixel accuracy (see comparisons with a classical finite difference discretization, Fig.3). On the other hand, this scheme is unconditionally stable and allows to choose very large time steps dt, without visible artifacts in the obtained regularization ($dt \simeq 50$ in our experiments). As a result, the algorithm performs very fast ($\simeq \times 3$) compared to traditional diffusion PDE implementations.


(a) Noisy color image.

(b) Regularization using a finite-difference scheme (stopped at $t = 100$).

(c) Regularization using our LIC-based scheme (stopped at $t = 100$).

Fig. 3. Comparison between traditional and LIC-based implementations of our PDE (11).

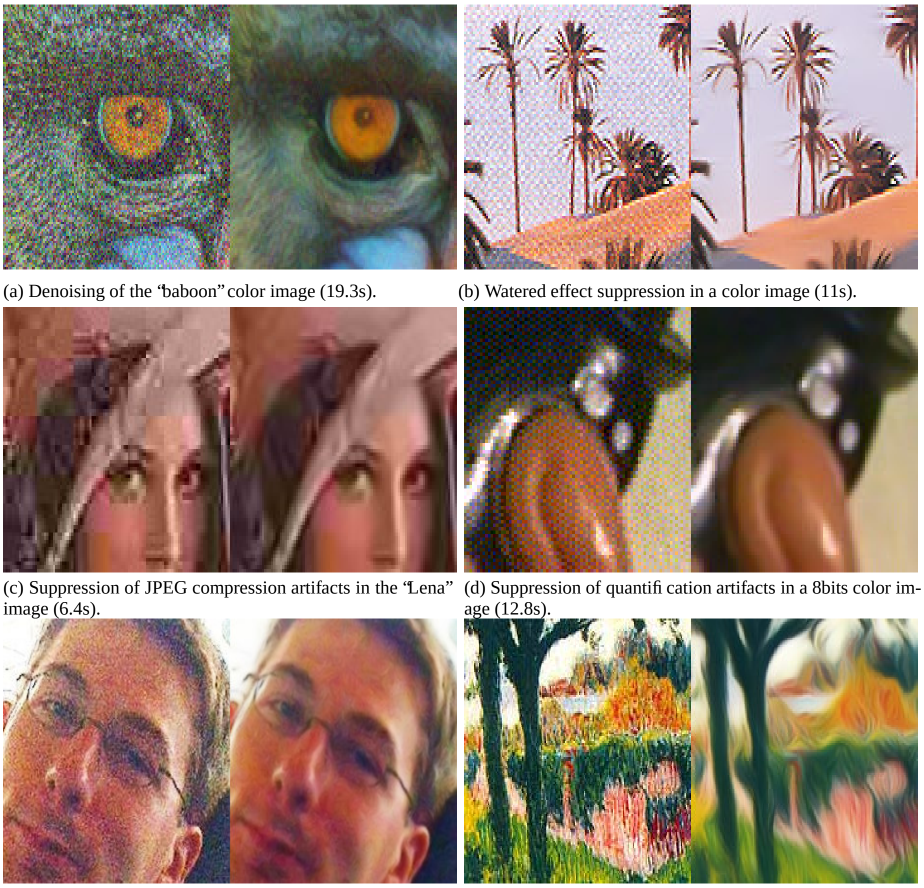
Fig.4 and 5 present different application results of our curvature-preserving PDE (11), implemented by the LIC-based scheme and applied on 24bits RGB color images $\mathbf{I} : \Omega \rightarrow [0, 255]^3$. All experiments have been performed on a single-CPU PC 2.8 Ghz running Linux. Possible application range covers color image regularization (PDE is applied on the entire image), inpainting (PDE is applied only inside regions to inpaint), and non-linear resizing (similar to inpainting with a sparse mask). See [31, 32] for more precisions on how diffusion PDE's are used in these contexts. Processing time is displayed for each example. Other results, as well as the C++ source code of the proposed algorithm can be found at : `http://www.greyc.ensicaen.fr/˜dtschump/greycstoration/`.

References

1. L. Alvarez, F. Guichard, P.L. Lions, J.M. Morel. *Axioms and fundamental equations of image processing.* Arch. for Rational Mechanics and Analysis, Vol.123, No.3, pp.199–257, 1993.
2. G. Aubert and P. Kornprobst. *Mathematical Problems in Image Processing: PDE's and the Calculus of Variations*, Applied Math. Sciences, Vol.147, Springer-Verlag, January 2002.
3. M. Bertalmio, L.T. Cheng, S. Osher, and G. Sapiro. *Variational Problems and PDE's on Implicit Surfaces.* Comput. and Visual. in Science, Vol.174, No.2, pp.759–780, 2001.
4. M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester. *Image inpainting.* ACM SIGGRAPH, Int. Conf. on Computer Graphics and Interactive Techniques, pp.417–424, 2000.

5. M.J. Black, G. Sapiro, D.H. Marimont, and D. Heeger. *Robust anisotropic diffusion.* IEEE Transaction on Image Processing, Vol.7, No.3, pp.421–432, 1998.
6. B. Cabral and L.C. Leedom. *Imaging vector fields using line integral convolution.* SIGGRAPH'93, in Computer Graphics Vol.27, pp.263–272, 1993.
7. R. Carmona and S. Zhong. *Adaptive Smoothing Respecting Feature Directions.* IEEE Transactions on Image Processing, Vol.7, No.3, pp.353–358, 1998.
8. A. Chambolle and P.L. Lions. *Image recovery via total variation minimization and related problems.* Nümerische Mathematik, Vol.76, No.2, pp.167–188, 1997.
9. T. Chan and J. Shen. *Variational restoration of non-flat image features : Models and algorithms.* SIAM Journal of Applied Mathematics, Vol.61, No.4, pp.1338–1361, 2000.
10. P. Charbonnier, L. Blanc-Féraud, G. Aubert, M. Barlaud. *Deterministic edge-preserving regularization in computed imaging.* IEEE Trans. on Image Proc., Vol.6 (2), 1997.
11. C. Chefd'hotel, D. Tschumperlé, R. Deriche, and O. Faugeras. *Regularizing Flows for Constrained Matrix-Valued Images.* Journ. of Math. Imaging and Vision, Vol.20 (2), Jan. 2004.
12. S. Di Zenzo *A note on the gradient of a multi-image.* Computer Vision, Graphics and Image Processing, Vol.33, pp.116-125, 1986.
13. R. Kimmel, R. Malladi, and N. Sochen. *Images as embedded maps and minimal surfaces: movies, color, texture, and volumetric medical images.* IJCV, Vol.39 (2), Sept. 2000.
14. R. Kimmel and N. Sochen. *Orientation diffusion or how to comb a porcupine.* Journal of Visual Communication and Image Representation, Vol.13, pp.238–248, 2002.
15. J.J. Koenderink. *The structure of images.* Biological Cybernetics, Vol.50, pp.363–370, 1984.
16. P. Kornprobst, R. Deriche, and G. Aubert. *Nonlinear operators in image restoration* IEEE Conference on Computer Vision and Pattern Recognition, pp 325-331, June 1997.
17. K. Krissian. *Multiscale Analysis : Application to Medical Imaging and 3D Vessel Detection.* Ph.D. Thesis, INRIA-Sophia Antipolis/France, 2000.
18. T. Lindeberg. *Scale-Space Theory in Computer Vision.* Kluwer Academic Publishers, 1994.
19. M. Nielsen, L. Florack, and R. Deriche. *Regularization, scale-space and edge detection filters.* Journal of Mathematical Imaging and Vision, Vol.7, No.4, pp.291–308, 1997.
20. P. Perona. *Orientation diffusions.* IEEE Trans. on Image Proc., Vol.7 (3), pp.457–467, 1998.
21. P. Perona and J. Malik. *Scale-space and edge detection using anisotropic diffusion.* IEEE Trans. on Pattern Anal. and Machine Intell., Vol.12 (7), pp.629–639, 1990.
22. T. Preusser and M. Rumpf. *Anisotropic nonlinear diffusion in flow visualization.* IEEE Visualization Conference, 1999.
23. W.H. Press, B.P. Flannery, S.A. Teukolsky, and W.T. Vetterling. *"Runge-Kutta Method"* in Numerical Recipes, Cambridge University Press, pp. 704-716, 1992.
24. L. Rudin, S. Osher, and E. Fatemi. *Nonlinear total variation based noise removal algorithms.* Physica D, Vol.60, pp.259–268, 1992.
25. G. Sapiro. *Geometric Partial Differential Equations and Image Analysis*. Cambridge University Press, 2001.
26. G. Sapiro and D.L. Ringach. *Anisotropic diffusion of multi-valued images with applications to color filtering.* IEEE Transactions on Image Processing, Vol.5, No.11, pp.1582–1585, 1996.
27. N. Sochen, R. Kimmel, and A.M. Bruckstein. *Diffusions and confusions in signal and image processing.* Journal of Mathematical Imaging and Vision, Vol.14, No.3, pp.195–209, 2001.
28. D. Stalling and H.C. Hege. *Fast and Resolution Independent Line Integral Convolution.* SIGGRAPH, 22nd Ann. Conf. on Computer Graphics, pp.249–256, 1995.
29. B. Tang, G. Sapiro, and V. Caselles. *Direction diffusion.* IEEE International Conference on Computer Vision, pp.1245, 1999.
30. B. Tang, G. Sapiro, and V. Caselles. *Diffusion of general data on non-flat manifolds via harmonic maps theory : The direction diffusion case.* IJCV, Vol.36 (2), pp.149–161, 2000.
31. D. Tschumperlé *PDE's Based Regularization of Multi-valued Images and Applications.* PhD Thesis, Université de Nice-Sophia Antipolis/France, December 2002.

32. D. Tschumperlé and R. Deriche *Vector-Valued Image Regularization with PDE's : A Common Framework for Different Applications.* IEEE Trans. on PAMI, Vol.27, No.4, April 2005.
33. D. Tschumperlé. *The CImg Library :* `http://cimg.sourceforge.net`. The C++ Template Image Processing Library.
34. B. Vemuri, Y. Chen, M. Rao, T. McGraw, T. Mareci, and Z. Wang. *Fiber tract mapping from diffusion tensor MRI.* IEEE Workshop on Variat. and Level Set Methods, July 2001.
35. J. Weickert. *Anisotropic Diffusion Filters for Image Processing Based Quality Control.* 7th European Conference on Mathematics in Industry, pp.355–362, 1994.
36. J. Weickert. *Anisotropic Diffusion in Image Processing*. Teubner-Verlag, Stuttgart, 1998.
37. J. Weickert. *Coherence-Enhancing Diffusion of Colour Images*. Image and Vision Computing, Vol.17, pp.199–210, 1999.
38. J. Weickert and T. Brox. *Diffusion and Regularization of Vector and Matrix-valued Images.* Inverse Problems, Image Analysis, and Medical Imaging, Vol.313 of Contemporary Mathematics, pp.251–268, 2002.



(a) Denoising of the "baboon" color image (19.3s).

(b) Watered effect suppression in a color image (11s).

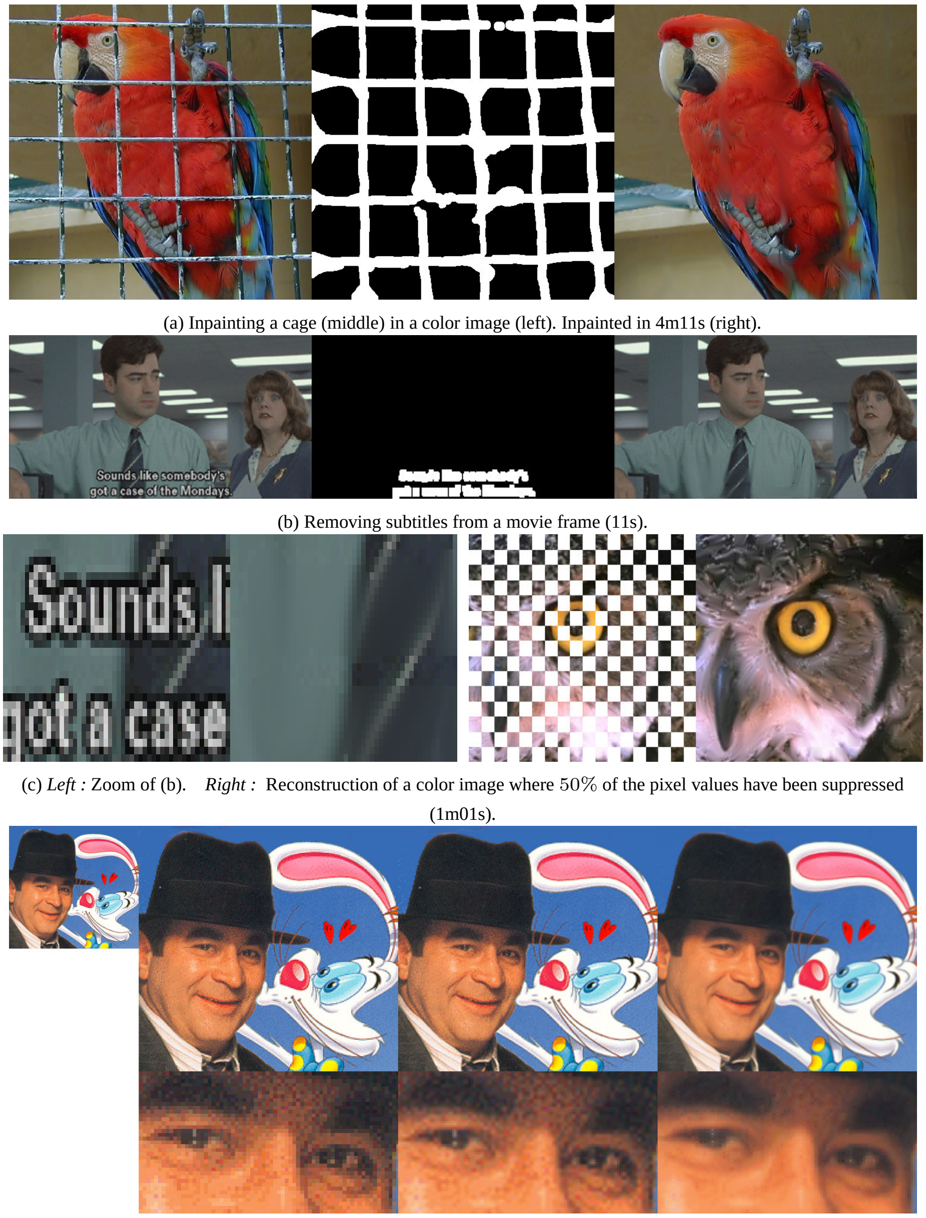
(c) Suppression of JPEG compression artifacts in the "Lena" image (6.4s).

(d) Suppression of quantification artifacts in a 8bits color image (12.8s).

(e) Denoising of a digital photograph with digital noise (5.6s).

(f) Creating painting effects with over-smoothing procedures (26s).

Fig. 4. Results of color image regularization using our curvature-preserving PDE's (11).



(a) Inpainting a cage (middle) in a color image (left). Inpainted in 4m11s (right).

(b) Removing subtitles from a movie frame (11s).

(c) *Left :* Zoom of (b). *Right :* Reconstruction of a color image where 50% of the pixel values have been suppressed (1m01s).

From left to right : First row : Original color image, bloc-resizing, bicubic resizing, PDE-based resizing.

Fig. 5. Result of our curvature-preserving PDE (11) for interpolation of color images. (More results at *http://www.greyc.ensicaen.fr/˜dtschump/greycstoration/*)