Topology preserving linear filtering applied to medical imaging

A. Buades * A. Chien † J.M. Morel ‡ S. Osher §

Abstract

One of the central problems of medical imaging is the 3D visualization of body parts. The 3D volume can be viewed in slices, but the extraction of a part requires a segmentation process. Inasmuch as body parts are distinguishable by their various densities, a widely accepted method to extract an organ is to extract iso-density surfaces by a simple threshold. Unfortunately the density of organs, arteries, etc. varies spatially due to morphology and no unique threshold allows one to extract the organs boundaries. The snake or active contours methods have attempted to capture these boundaries as smooth and overall contrasted surfaces. The snake method suffers, however, from severe drawbacks. The contour has to be initialized near the boundary. In addition many body parts have a too complex topology. In this paper we focus on another idea, which is to preprocess the image before thresholding. The preprocessing aims at the homogeneity of the different parts while preserving small features. Starting from a seminal recent work by Grady et al. several linear heat equations on images will be compared. They stem from non linear PDEs or from their associated nonlinear filters. By linearizing these processes one obtains more accurate topology preserving methods. These linear filters will be tested comparatively to visualize challenging angiography images of arteries. A salient fact of the method will emerge. By a concentration phenomenon, peaks in the image histogram become much more concentrated under the linear heat equations, thus permitting to fix the thresholds defining the surfaces without supervision. Automatic extraction can be in this way performed for angiography images taken at one year delay or more.

1 Introduction

Advances in 3D medical imaging have taken a prominent role in clinical evaluation and treatment planning [39, 28, 36, 9]. 3D images allow reconstruction of organs or vessels, and allow physicians to visualize morphology and monitor pathological changes [39, 36, 9, 31]. However, these detailed images generate large data sets. To efficiently analyze this 3D data has been technically challenging. Researchers have proposed methods for segmentation

*University of Balearic Islands, Ctra. Valldemossa Km. 7.5, 07122 Palma de Mallorca, Spain (e-mail: toni.buades@uib.es)

†Division of Interventional Neuroradiology, David Geffen School of Medicine at UCLA, 10833 LeConte Ave, Los Angeles, CA 90095. (email: aichi.seas@gmail.com).

‡CMLA, ENS Cachan 61, Av du President Wilson 94235 Cachan, France (e-mail: morel@cmla.ens-cachan.fr)

§IPAM, University of California Los Angeles, 460 Portola Plaza, Los Angeles, CA 90095-7121. (email: sjo@math.ucla.edu).

0Work supported by NGIA grant HM-1582-07-C-010, NIH grant P20 MH65166, NSF grant DMS-0714087, ONR grant N00014-97-1-0839 and MEC grant MTM-2005-08567.

and reconstruction [23, 43, 38]. However, in addition to general imaging processing issues such as noise and artifacts, the greatest challenges presented by medical images are the presence of inhomogeneous contrast, and irregular contouring [31, 43, 10, 7]. In fact, many important pathological abnormalities are also indicated by inhomogeneous contrast or irregular contouring in an image. Therefore, an algorithm that distinguishes noise without losing topology is needed.

Image processing on medical images usually favors one particular imaging modality, and using prior information in the reconstruction process is common [23, 7, 1]. However, the usage of prior information can easily omit the small changes in pathology which are important in clinical follow-up and essential to detect motion. In this paper, we utilized the information provided by images without relying on prior information or methods specific to a particular image acquisition technique. Therefore, in the future, the presented algorithm can be easily applied to dynamic imaging techniques (dynamic MR or 4D CT images) to detect small pulsation and derive tissue functional data.

The mathematical tools used to solve this problem go across partial differential equations (PDEs) and the calculus of variations. We can distinguish two kinds of methods. Some define an iterative process which can be implemented directly [21]. Other more ambitious ones fix a segmentation energy to be minimized for the ideal segmentation [29]. Such methods are called energy methods, or variational methods. In a wide extent, however, both approaches are linked. Indeed most of the considered PDEs minimize some energy and conversely variational methods are almost necessarily implemented as a PDE evolution. Thus, we'll concentrate on PDEs and we'll go back later to the variational principles. In image processing, PDEs are nothing but formulas describing local interactions between neighboring pixels. In the case of smoothing PDEs, this interaction is described as an evolution starting from the initial image $u_0(\mathbf{x}) = u(0, \mathbf{x})$ and leading to a smoother version $u(t, \mathbf{x})$ depending on a scale t.

Since many image processing algorithms are described in a discrete setting we'll jump from the continuous to the discrete by writing $u(\mathbf{x})$ for the grey level image defined on the continuous domain Ω and $u(i)$ for i belonging to a discrete set of pixels on a regular grid of Ω.

Our analysis will mainly deal with linearizations of classic and less classic PDEs. Probably the main technical outcome of the present study is to demonstrate that the linearized versions of several classical and less classical nonlinear heat equations work better for image segmentation than their nonlinear counterparts. This is *a posteriori* understandable: while the solution $u(t, \mathbf{x})$ to the nonlinear equations goes away from its initial data u_0, the linearized version maintains the direct knowledge of u_0 and this presence acts as a fidelity term to the initial datum. The heat equations we consider are non local on the image domain, but can be interpreted as local heat equations in a similarity graph associated with the initial image u_0. In this we shall follow and extend recent works where grey level similarity has been used to define the weights between nodes of a graph whose nodes are all image pixels [3, 4, 24, 17, 18].

1.1 All filters in a table

On the technical side the main lines of our discussion on image filters with increasing image fidelity are summarized in Table 1. Each row represents an image processing method. The first column is the name of the process, the second one gives the associated discrete filter name, the third column gives the PDE form and the fourth column the form of the linearized version. Finally the fifth column specifies the initial data. It can be the image or, in the case where a segmentation is sought, a set of seed points with value 1 and others with zero value, manually indicated by an expert. In this latter case the result of the

evolution is not a smoothed version of u_0 but a probability for each pixel to belong to the same region as the seed [18, 16]. In all cases the steady state for the evolution is a "harmonic function" with respect to a graph Laplacian associated with the initial image. This harmonic version is faithful to the original u_0, but with more homogeneous regions.

PDE	Filter	PDE	Linearized PDE	initial
Heat	Gaussian	$\frac{\partial u}{\partial t} = \Delta u$	$\frac{\partial u}{\partial t} = \Delta u$	u_0
Mean Curv.	median	$\frac{\partial u}{\partial t} = \Delta_{Du^\perp} u$	$\frac{\partial u}{\partial t} = \Delta_{Du_0^\perp} u$	u_0
Per.-Malik	sigma filter	$\frac{\partial u}{\partial t} = \Delta_{NF(u)} u$	$\frac{\partial u}{\partial t} = \Delta_{NF(u_0)} u$	u_0
Grady	sigma-filter	$\frac{\partial u}{\partial t} = \Delta_{NF(u)} u$	$\frac{\partial u}{\partial t} = \Delta_{NF(u_0)} u$	Seeds
Non Local	NL-means	$\frac{\partial u}{\partial t} = \Delta_{NL(u)} u$	$\frac{\partial u}{\partial t} = \Delta_{NL(u_0)} u$	u_0

Table 1: **Table of main PDEs and their linearized versions, which are all heat equations.** Each row represents an image processing method. The first column is the name of the PDE, the second one gives the associated discrete filter name, the third column gives the nonlinear PDE form and the fourth column the linearized version. The fifth column specifies the kind of initial data. The initial data can be the image or, in the case where a segmentation is sought, some seed points with value 1 et others with zero value are indicated by the expert. In this later case the result of the evolution is not a smoothed version of u_0 but a probability for each pixel to belong to the same region as the seed. In all cases the steady state for the evolution is a "harmonic function" with respect to a graph Laplacian associated with the initial image u_0.

The four main *neighborhood filters* we consider in the second column are:

- the Gaussian mean, a Gaussian average of neighboring pixels,

$$G_\rho u(\mathbf{x}) = \frac{1}{C(\mathbf{x})} \int_\Omega e^{-\frac{|\mathbf{x}-\mathbf{y}|^2}{\rho^2}} u(\mathbf{y}) d\mathbf{y};$$

- the anisotropic or median filter, Gaussian average of pixels the level line direction,

$$AF_h u(\mathbf{x}) = G_h * u_{|L(\mathbf{x})} = \int_{\mathbb{R}} G_h(t) u(\mathbf{x} + t\frac{Du^\perp}{|Du|}) dt$$

  where $L(\mathbf{x})$ is the line passing by $\mathbf{x}$ and orthogonal to the gradient of u at $\mathbf{x}$;

- The sigma filter, a local average of pixels with similar grey level,

$$NF_{\rho,h} u(\mathbf{x}) = \frac{1}{C(\mathbf{x})} \int_\Omega e^{-\frac{|\mathbf{x}-\mathbf{y}|^2}{\rho^2}} e^{-\frac{|u(\mathbf{x})-u(\mathbf{y})|^2}{h^2}} \; u(\mathbf{y}) \; d\mathbf{y};$$

  (In some versions the spatial distance weights are replaced by a distance threshold.)

- the non-local means filter, an average of all pixels with similar local configuration

$$NL_{h,a} u(\mathbf{x}) = \frac{1}{C(\mathbf{x})} \int_\Omega e^{-\frac{1}{h^2}\int_{\mathbb{R}^2} G_a(t)|u(\mathbf{x}+t)-u(\mathbf{y}+t)|^2 dt} \; u(\mathbf{y}) \; d\mathbf{y},$$

  where G_a is a Gaussian kernel of standard deviation a and h acts as a filtering parameter. In all cases $C(\mathbf{x})$ is a normalization constant so that the integral of coefficients is 1.

The sigma filter due to Lee [25] fits into the general theory of neighborhood filters developed by the Soviet school [41]. Several variants, discrete and continuous make it also known as SUSAN [37] and as the bilateral filter [40]. An iterative version of this filter is applied to color imaging in [11]. The authors of this paper also show by a kernel based density

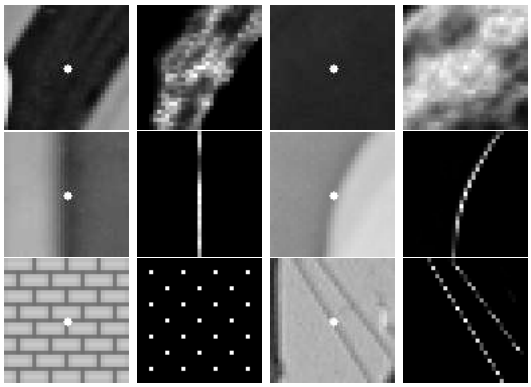

Figure 1: On the right-hand side of each pair, we display the weight distribution used to estimate the central pixel of the left image by the NL-means algorithm. Compare with Figure 2.

estimation point of view that clusters of points converge to its mean, making such an approach useful for segmentation purposes.

In the NL-means filter the resemblance between pixels is evaluated by comparing a whole window around each pixel, not just the color of the pixel itself. This idea which originates in [14] leads to a generalized sigma filter whose denoising properties were studied in [5]. NL-means is more robust to noise than the sigma filter. The window comparison also makes the weight distribution of NL-means adapt to the local geometry of the image as displayed in Figure 1 which we take from [5].

The simple form of the PDEs in the table of Figure 1 is slightly deceptive. It could suggest that we consider heat equations, that is local, linear and isotropic image processes. In fact, with the exception of the very first equation, all considered equations in the third column are nonlinear local or nonlocal diffusion processes. However, from the numerical view point they all compute a neighborhood for each pixel and then perform an average on this adaptive neighborhood. Passing from the third to the fourth column only means on the numerical side that the neighborhood of each pixel is fixed at the beginning of the iterations. Thus, it is computed with the initial image u_0 and not computed again at each time iteration. So the equations of the fourth column which are the linearized versions of the ones in the third column turn out to be iterative averages on fixed neighborhoods. They are considerably simpler, being linear. One of the main outcomes of the present paper is to demonstrate that these linear equations are more adapted to a faithful image processing.

The main differences between the five rows of the table is the shape of the neighborhood of these equations viewed as implementations of an iterative average on an adaptive neighborhood. The shape of these neighborhoods is shown (in red) in Figure 2. The central point is assumed to be grey in the left hand image. The second image shows the Gaussian radial neighborhood associated of this point, which corresponds to the first row of the table. The third image shows the neighborhood given by the median filter (this filter turns out to be equivalent to an average in the direction tangent to the level line [21]). The fourth and fifth row of our table correspond to the classical sigma-filter, which select pixels which are both close in space and in grey level. The last row of the table corresponds to a more sophisticated neighborhood filter, the non-local means filter, which selects the pixels for which a surrounding window looks like the corresponding window of the current pixel. In all cases the red points depict a criterion for pixel similarity, with growing efficiency.

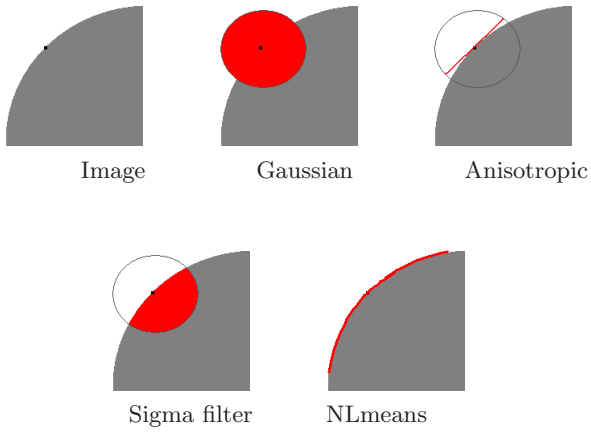

Figure 2: In red: the shape of the neighborhoods involved in 2D image filters. From left to right: a point is picked on the disk's boundary; the heat equation corresponds to a Gaussian average (red disk centered at the point). The median corresponds to an anisotropic filtering in the direction tangent to the disk boundary. The sigma filter selects points which are close to the current point in both space and grey level value. NL-means selects as neighbors the points for which a small image window around them is similar to the corresponding image window around the current pixel. This results in the selection of a piece of the grey disk's boundary. See Figure 1 for examples of NL-means neighborhoods on real images.

1.2 Plan of the paper

Our plan follows from the above table. Section 2 discusses the comparative properties of the PDEs, with emphasis on their topology preserving properties. Section 3 gives the linearizations of all nonlinear heat equations and nonlinear filters and shows experimentally that they are more faithful to the original. They also are much easier and faster to implement. Section 4 contains an application to medical image segmentation. One of the main outcomes of the linear filters is illustrated in Section 4.3 on the *histogram concentration phenomenon*.

2 Classic PDEs first

2.1 The heat equation as simplest neighborhood filter

The first process we shall consider is the heat equation as a PDE and the Gaussian convolution as a filter. This filter stems from the remark that the difference between the original image u_0 and a blurred version image of it, $k * u_0$, is roughly proportional to its Laplacian. More precisely assume that k is spatially concentrated and scale k as $k_h(\mathbf{x}) = \frac{1}{h}k(\frac{\mathbf{x}}{h^{\frac{1}{N}}})$, where N is the image domain dimension and $h \to 0$. Denote by $\mathbf{x}$ a point of the image domain and assume that u_0 is C^3 around $\mathbf{x}$. Assume further that k is a positive radial kernel satisfying $\int(1 + |\mathbf{x}|^2 + |\mathbf{x}|^3)k(\mathbf{x})d\mathbf{x} < \infty$ and $\int x_1^2 k(\mathbf{x})d\mathbf{x} = 2$. Under these assumptions and using a Taylor expansion of u_0 around $\mathbf{x}$, it is easy to prove that [21]:

$$\frac{k_h * u_0(\mathbf{x}) - u_0(\mathbf{x})}{h} \to \Delta u_0(\mathbf{x})$$

as $h \to 0$. We can rewrite this relation as

$$k_h * u_0(\mathbf{x}) - u_0(\mathbf{x}) = h\Delta u_0(\mathbf{x}) + o(h). \tag{1}$$

Let $u(t, \mathbf{x})$ denote the solution of the heat equation

$$\frac{\partial u}{\partial t} = \Delta u, \quad u(0, \mathbf{x}) = u_0(\mathbf{x}). \tag{2}$$

If u_0 is C^2 and bounded, then we deduce that

$$u(t, \mathbf{x}) - u(0, \mathbf{x}) = t\Delta u_0(\mathbf{x}) + o(t). \tag{3}$$

The comparison of equations (1) and (3) shows that blurring u_0 with a kernel k_h is for small h equivalent to applying the heat equation to u_0 at scale h.

2.2 The Perona-Malik equation

Our second main equation was proposed by Perona and Malik [30] to avoid the blur effect of the Gaussian convolution. The idea is to smooth what needs to be smoothed, namely the homogeneous regions, and to enhance the boundaries of u_0. With this in mind, the diffusion should look like the heat equation when $|Du|$ is small, but it should act like the inverse heat equation when $|Du|$ is large. This leads to an equation in divergence form:

$$\frac{\partial u}{\partial t} = \mathrm{div}(g(|Du|)Du), \tag{4}$$

where (e.g.) $g(s) = 1/(1 + \lambda^2 s^2)$. It is easily checked that we have a diffusion equation when $\lambda|Du| \leq 1$ and an inverse diffusion equation when $\lambda|Du| > 1$. To see this, consider the second derivative of u in the direction of Du,

$$u_{\eta\eta} = D^2 u\left(\frac{Du}{|Du|}, \frac{Du}{|Du|}\right),$$

and the second derivative of u in the orthogonal direction,

$$u_{\xi\xi} = D^2 u\left(\frac{Du^\perp}{|Du|}, \frac{Du^\perp}{|Du|}\right),$$

where $Du = (u_x, u_y)$ and $Du^\perp = (-u_y, u_x)$. The Laplacian can be rewritten in the intrinsic coordinates (ξ, η) as $\Delta u = u_{\xi\xi} + u_{\eta\eta}$. We display an example of these intrinsic coordinates in Figure 3. The Perona–Malik equation writes in the intrinsic coordinates

$$\frac{\partial u}{\partial t} = \frac{1}{1 + \lambda^2|Du|^2} u_{\xi\xi} + \frac{1 - \lambda^2|Du|^2}{(1 + \lambda^2|Du|^2)^2} u_{\eta\eta}.$$

The first term in this representation always appears as a one-dimensional diffusion in the direction orthogonal to the gradient, tuned by the size of the gradient. The nature of the second term depends on the value of the gradient; it can be either a diffusion in the direction Du or an inverse diffusion in the same direction.

Figure 3: The intrinsic coordinates ξ and η computed at a point of an image. The coordinate η is normal to the level lines and therefore normal to the image structure while the coordinate ξ is the tangent to the level line. The image Laplacian $\Delta u = u_{\eta\eta} + u_{\xi\xi} = D^2u(\frac{Du}{|Du|}, \frac{Du}{|Du|}) + D^2u(\frac{Du^\perp}{|Du|}, \frac{Du^\perp}{|Du|})$ can be decomposed into the sum of the second derivatives in both directions.

2.3 Perona-Malik as a sigma filter

Local average filters can be asymptotically related to the above PDE formulations. It was suggested in [2] and proved in [6] that the sigma filters are asymptotically equivalent to a Perona-Malik equation. This result was proven with the slightly simpler (but in practice equivalent) form

$$NF_{h,\rho}u(\mathbf{x}) = \frac{1}{C(\mathbf{x})}\int_{B_\rho(\mathbf{x})} u(\mathbf{y})e^{-\frac{|u(\mathbf{y})-u(\mathbf{x})|^2}{h^2}}d\mathbf{y}, \tag{5}$$

where $B_\rho(\mathbf{x})$ is a ball of center $\mathbf{x}$ and radius ρ, h is the filtering parameter and $C(\mathbf{x}) = \int_{B_\rho(\mathbf{x})} e^{-\frac{|u(\mathbf{y})-u(\mathbf{x})|^2}{h^2}}d\mathbf{y}$ is the normalization factor. The Perona-Malik equation writes in general form

$$u_t = \mathrm{div}(g(|Du|^2)Du),$$

where $g : [0, +\infty) \to [0, +\infty)$ satisfying $g(0) = 1$, $\lim_{s\to+\infty} g(s) = 0$ is a smooth decreasing function. This equation rewrites as

$$u_t = g(|Du|^2)u_{\xi\xi} + h(|Du|^2)u_{\eta\eta}. \tag{6}$$

It is good to get the shape of g and h as illustrated in Figure 4. Theorem 1 below and Figures 4 and 5 explain why sigma filters are efficient implementations of a Perona-Malik equation.

Theorem 1 *[6] If h and ρ are of the same order of magnitude,*

$$NF_{h,\rho}u(\boldsymbol{x}) - u(\boldsymbol{x}) \simeq \left[\hat{g}(\frac{\rho}{h}\,|Du(\boldsymbol{x})|)\;u_{\xi\xi}(\boldsymbol{x}) + \hat{h}(\frac{\rho}{h}\,|Du(\boldsymbol{x})|)\;u_{\eta\eta}(\boldsymbol{x}).\right]\rho^2 \tag{7}$$

First, the form of the PDE (7) is the same as for the Perona-Malik form (6) and second the functions $\hat{g}$ and $\hat{h}$ weighting the diffusion tangent to the level line and the diffusion orthogonal to the level line have the same shape as the corresponding functions g and h in (6). This confirms the experimental evidence that the sigma filter is an implementation of a Perona-Malik equation. This implementation as a local adaptive average filter is notably simple. It yields a maximum decreasing, minimum increasing numerical scheme.

2.4 Mean curvature motion, the median and the geometry of surfaces

The mean curvature motion was first introduced in shape analysis by Tannenbaum et al. [22] and a clever implementation, the "threshold dynamics" can be found in [27]. It

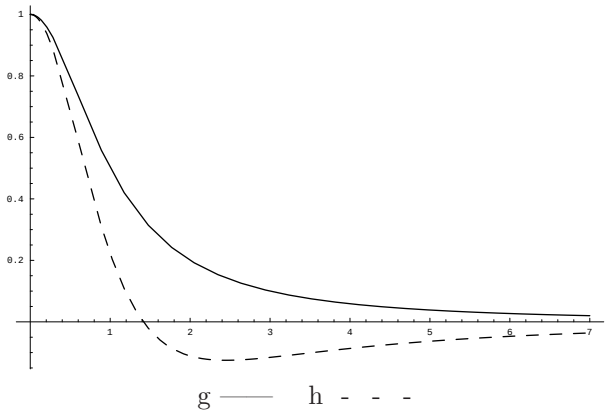

Figure 4: The diffusion and inverse diffusion weights in the Perona-Malik equation. The diffusion in the direction of level line term coefficient is $g(|Du|^2)$. See its graph above in continuous line. This term remains positive, which means that there is a diffusion in the direction of the level line, like in the mean curvature motion. This diffusion is slower when the gradient is large. The coefficient of the diffusion in the direction normal to the level line is $h(|Du|^2)$. This term performs like an inverse heat equation (an enhancement) in the direction of the gradient when the gradient is above a critical value. This enhancement effect also decays when the gradient is large. Compare with the shapes of g and h for the sigma filter.

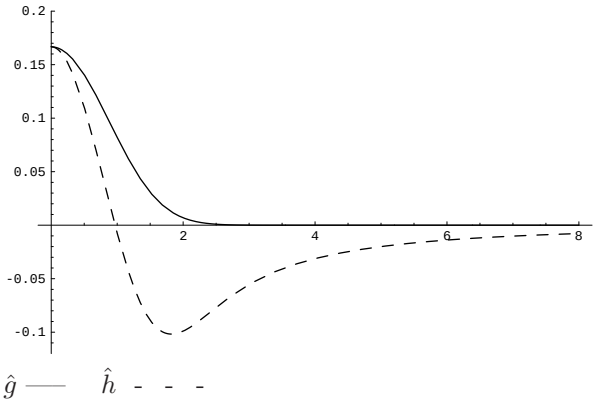

Figure 5: The functions $\hat{g}$ and $\hat{h}$ weighting diffusion tangent to the level line and diffusion orthogonal to the level line for the sigma filter. They have the same shape as for a Perona-Malik equation. This implies that the sigma filter has the same qualitative action as the Perona-Malik equation. It can be considered as an implementation of a Perona-Malik equation, particularly remarkable as it performs like a local adaptive average equation.

amounts in fact to iterate a radial median on the image [21]. Let us give a more geometric explanation about the mean curvature motion. This equation writes

$$\frac{\partial u}{\partial t} = |Du| \operatorname{div}\left(\frac{Du}{|Du|}\right). \tag{8}$$

In order to understand its geometric meaning one has first to introduce the principal curvatures of the level surface of u.

Definition 1 *Let $u : \mathbb{R}^N \to \mathbb{R}$ be C^2 at $\boldsymbol{x}_0$, with $Du(\boldsymbol{x}_0) \neq 0$. The principal curvatures of u at $\boldsymbol{x}_0$ are the real numbers*

$$\kappa_j = \frac{\mu_j}{|Du(\boldsymbol{x}_0)|},$$

where μ_j are the eigenvalues of $D^2u(\boldsymbol{x}_0)$ restricted to $Du(\boldsymbol{x}_0)^\perp$.

Definition 2 *The mean curvature of a C^2 function $u : \mathbb{R}^N \to \mathbb{R}$ at $\boldsymbol{x}_0 \in \mathbb{R}^N$ is the sum of the principal curvatures at $\boldsymbol{x}_0$. It is denoted by* $\operatorname{curv}(u)(\boldsymbol{x}_0)$ *and satisfies*

$$\operatorname{curv}(u) = \operatorname{div}\Big(\frac{Du}{|Du|}\Big).$$

It is an easy calculation to check that, if u is C^2 and $g : \mathbb{R} \to \mathbb{R}$ a "contrast change", namely a non decreasing C^2 function with $g'((u(\mathbf{x}))) \neq 0$, then $curv(g \circ u)(\mathbf{x}) = curv(u)(\mathbf{x})$. This means that the mean curvature only depends on the geometry of the level surface and on the sign of u (indeed $curv(-u) = -curv(u)$.)

Finally the next proposition permits to interpret the mean curvature motion as a geometric motion of the level surface.

Proposition 2 *Let $u(t, \boldsymbol{x})$ be a solution of the mean curvature motion which is C^2 around $\boldsymbol{x}$ and such that $Du(\boldsymbol{x}) \neq 0$. Then the level surface of u passing by $\boldsymbol{x}$ moves according to the equation*

$$\frac{\partial \boldsymbol{x}}{\partial t} = curv(u)n(\boldsymbol{x}),$$

where $n(\boldsymbol{x}) = \frac{Du}{|Du|}(\boldsymbol{x})$ is the normal to the level surface at $\boldsymbol{x}$.

Notice that $curv(u)(\mathbf{x})$ and $n(\mathbf{x})$ are invariant with respect to contrast changes and that they both change sign when u is changed into $-u$. Thus $curv(u)(\mathbf{x})n(\mathbf{x})$ does not change sign and the motion only depends on the level surface of u passing by $\mathbf{x}$. This is why this motion is called geometric.

In three dimensions the level surfaces have two principal curvatures. In the case of a cylindrical surfaces, the highest curvature is the inverse of the radius of the cylinder. Thus under mean curvature motion such a surface evolves by getting thinner and eventually collapses. The speed being proportional to the curvature, broader cylinders hardly move, while thinner cylinders disappear quickly. Thus, applying the mean curvature motion to a set of connected tubes like the arteries network leads to the elimination of the smaller arteries. The straight broad arteries are instead nearly steady. In consequence the mean curvature motion is a simplification tool for irrigation networks. It can be implemented very easily by iterating a median filter [27]. However, we shall see that in practice the removal effect for thin arteries is much too quick and makes mean curvature motion unfit for 3D medical imaging. This is illustrated in the comparative experiments of Figure 9.

In [26], Weickert et al. introduced a variant of the curvature equation, with nonlocal estimate of the direction orthogonal to the gradient: the diffusion direction $d = SEigen(k * (Du \otimes Du))$ is computed as the eigenvector of the least eigenvalue of $k * (Du \otimes Du)$: if the convolution kernel is removed, this eigenvector simply is $Du^\perp$.

The Rudin-Osher-Fatemi [32] total variation model evolves the image by the steepest descent of the image total variation $TV(u) = \int |Du(\mathbf{x})| d\mathbf{x}$ and its evolution reads, at least formally

$$u_t = \mathrm{div}(\frac{Du}{|Du|}). \tag{9}$$

There is a clear link with the mean curvature motion and, in fact the mean curvature motion can be shown to decrease the total variation (and so does also the heat equation, by Jensen's inequality).

3 Linearizing all filters

3.1 Linearizing Perona-Malik and the mean curvature

We have seen in Section 2.3 that sigma filters implement a Perona-Malik equation. By linearized version of the Perona-Malik equation we mean, as for the linearized version of the mean curvature motion, that the diffusion directions are fixed and given by the initial image gradient Du_0. Thus the equation in two dimensions is

$$\frac{\partial u}{\partial t} = g(|Du_0|^2) u_{\xi_0 \xi_0} + h(|Du_0|^2) u_{\eta_0 \eta_0}$$

where η_0 denotes the coordinate parallel to Du_0 and ξ_0 the orthogonal one. In the same way, the mean curvature motion being a diffusion in the plane tangent to the level surface of u can be written

$$\frac{\partial u}{\partial t} = \Delta_{Du^\perp} u,$$

where $\Delta_{\vec{w}^\perp} u = \Delta u - D^2 u(\frac{\vec{w}}{|\vec{w}|}, \frac{\vec{w}}{|\vec{w}|})$ denotes the Laplacian of u restricted to the plane orthogonal to a vector $\vec{w}$. By fixing the diffusion plane to be always the tangent plane to the level surface of u_0, the fidelity of the evolution to the initial image is intuitively reinforced. The equation becomes linear and reads

$$\frac{\partial u}{\partial t} = \Delta_{Du_0^\perp} u.$$

As can be appreciated in Figure 6 the result of a linearized mean curvature motion starting from u_0 is fairly equivalent to the mean curvature motion starting from u_0. Some more fidelity to u_0 is, however, gained by this linearization.

3.2 Linear heat equations based on sigma filter and NL-means

As we shall see in Section 4.1, the sigma filter and NL-means can be used for a supervised segmentation of a medical image starting from seed regions. These filters can be applied iteratively, in which case we have the choice of fixing the weights or to let them evolve. If we fix them, we are led to a linearized version which can be interpreted as a heat equation. Calling now u_0 the initial image, let us set for $\mathbf{x}, \mathbf{y}$ in the image domain Ω

$$\tilde{w}(\mathbf{x}, \mathbf{y}) = e^{-\frac{-|u_0(\mathbf{y}) - u_0(\mathbf{x})|^2}{h^2}} \text{ or } e^{-\frac{\int_{\mathbb{R}^2} G_a(t) |u_0(\mathbf{x}+t) - u_0(\mathbf{y}+t)|^2 dt}{h^2}}$$

and to get normalized weights

$$w(\mathbf{x}, \mathbf{y}) = \frac{\tilde{w}(\mathbf{x}, \mathbf{y})}{\int_\Omega \tilde{w}(\mathbf{x}, \mathbf{y}) d\mathbf{y}}.$$

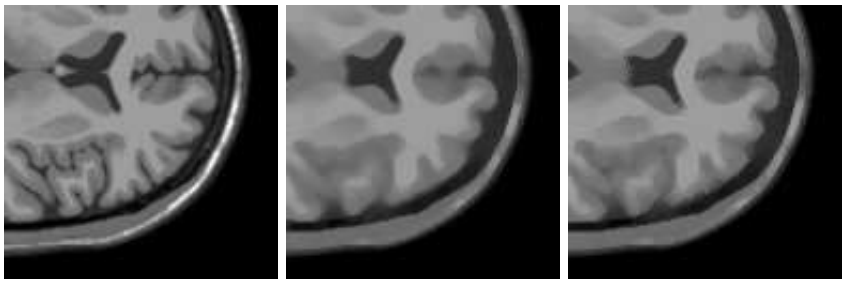

Figure 6: Median and linearized median. On the left a piece of brain image slice. Middle: iterated median filter, equivalent to the mean curvature motion $\frac{\partial u}{\partial t} = \Delta_{Du^\perp} u$. Right, linearized mean curvature motion, which amounts to smooth u in the direction tangent to the level lines of the initial u_0, starting from u_0. The linear equation is $\frac{\partial u}{\partial t} = \Delta_{Du_0^\perp} u$ and the effect is actually quite similar to the curvature motion, but slightly more faithful to the original.

Consider the weighted average linear operator defined for any function u on Ω by

$$L_{NF(u_0)}u(\mathbf{x}) =: \int_\Omega w(\mathbf{x},\mathbf{y})u(\mathbf{y})d\mathbf{y}.$$

Following the formalism in [16] for any function u on Ω the difference

$$\Delta_{NF(u_0)}u(\mathbf{x}) =: \int_\Omega w(\mathbf{x},\mathbf{y})u(\mathbf{y})d\mathbf{y} - u(\mathbf{x})$$

is called the *Laplacian of u for the weights $w(\mathbf{x},\mathbf{y})$*. This terminology can be viewed as a generalization of the asymptotic equivalence between local averaging and the Laplacian explained in Section 2.1.

Thus iterating the linear smoothing operator $u^{n+1}(\mathbf{x}) = L_{NF(u_0)}u(\mathbf{x})$ becomes an implementation of the generalized heat equation

$$\frac{\partial u}{\partial t} = \Delta_{NF(u_0)}, \quad u(0) = u_0$$

whose solution is

$$u(t) = e^{t(L_{NF(u_0)}-I)}u_0 = e^{t\Delta_{NF(u_0)}}u_0.$$

3.3 Comparing filters with their linear counterpart

Figure 7 shows a comparison of four diffusion processes. On the right fully non linear processes such as the mean curvature motion, the Perona-Malik equation, the sigma filter and the NL-means have been applied while on the left column their linearized versions are tested at the same evolution time. The aim of Figure 7 is definitely not to recommend the use of such filters for portraits, but only to demonstrate that the fidelity to the original image is increased when using the linearized version. This fact is quite apparent with the iterated linearized sigma-filter. In spite of the consistency of the sigma filter with the Perona-Malik equation, the visual aspects of the results by both filters are quite different. Both create shocks, namely new edges, but these edges are step edges for the Perona-Malik equation in both nonlinear and linear version. The edges created by the sigma filter are step edges, but this staircase effect does not occur in the linear version! In summary the *linearized version of the neighborhood (or sigma-, or bilateral) filter shows better image quality and fidelity.* We observe the same effect in the NL-means algorithm. The iterated version also produces shocks. These shocks are avoided by its linear counterpart.

Figure 7: Comparison of linear (left images) and nonlinear (right images) diffusion schemes. From top to bottom: original image, mean curvature motion, the Perona-Malik equation, the sigma filter and the NL-means. This experiment shows first that linearized versions are more faithful for the original. This is not a surprise, since the original image features are kept in the linearized version while they are updated in the nonlinear version. The third row illustrates how much better the linearized version of the neighborhood (or sigma-, or bilateral) filter performs with respect to the non linear one. We observe the same effect in the NL-means algorithm. The iterated version produces shocks as the neighborhood filter, which are avoided by its linear counterpart. See Section 3 for more details.

3.4 Variational interpretation of all filters

All PDEs considered above, with the exception of the mean curvature motion, derive from an energy functional. The heat equation (2) is the steepest descent from u_0 for the energy $\int_\Omega |Du|^2$. The Perona-Malik equation (4) is the steepest descent for $\int_\Omega G(Du)$ where G is a primitive of g. In [33] Kindermann-Osher-Jones interpreted the NL-means and the sigma filters in general as regularizations based on nonlocal functionals in the general form

$$\int_{\Omega\times\Omega} g\left(\frac{|u(\mathbf{x})-u(\mathbf{y})|^2}{h^2}\right) w(|\mathbf{x}-\mathbf{y}|)d\mathbf{x}d\mathbf{y},$$

where $w(|\mathbf{x}-\mathbf{y}|)$ is a spatial weight function. Gilboa et al. [16] consider the general kind of quadratic non local functional

$$J(u) := \int_{\Omega\times\Omega} (u(\mathbf{x})-u(\mathbf{y}))^2 w_{u_0}(\mathbf{x},\mathbf{y})d\mathbf{x}d\mathbf{y}. \tag{10}$$

where $w_{u_0}(\mathbf{x},\mathbf{y})$ now depends on the initial image u_0. The weight function $w(\mathbf{x},\mathbf{y}) \geq 0$ is always assumed nonnegative. The corresponding Euler-Lagrange equation for minima is $L(u)(\mathbf{x}) = 0$ with

$$L(u)(\mathbf{x}) = \int_\Omega (u(\mathbf{x})-u(\mathbf{y}))w_{u_0}(\mathbf{x},\mathbf{y})d\mathbf{y} \tag{11}$$

and the authors in [16] considered the gradient descent of the minimization for $J(u)$. The limit of the gradient descent for $J(u)$ is solution to (11) and is obtained by the numerical iteration

$$u_{n+1}(\mathbf{x}) = \frac{1}{C(\mathbf{x})}\int_\Omega u_n(\mathbf{y})w_{u_0}(\mathbf{x},\mathbf{y})d\mathbf{y}, \tag{12}$$

where $C(\mathbf{x}) = \int_\Omega w(\mathbf{x},\mathbf{y})d\mathbf{y}$ is a normalizing factor and u_0 is given as an initial condition of the process. The freedom of this initial condition and the weight distribution (independent of u) makes this formulation a linear and powerful tool for image processing.

Surprisingly enough and as anticipated in the seminal Grady work to be commented upon next, one can get good segmentation results with quadratic functionals which are by far simpler than the classic energy segmentation functionals which we shall now recall briefly for comparison. The most emblematic ones are the total variation restoration [32] where a restored image u is obtained from the original u_0 by minimizing a smoothness+fidelity functional

$$E_{TV}(u) = \int_\Omega |Du(\mathbf{x})|d\mathbf{x} + \lambda\int_\Omega (u-u_0)^2(\mathbf{x})d\mathbf{x};$$

and the Mumford-Shah energy [29] which seeks explicitly a set of boundaries K in the image,

$$E_{MS}(u) = \int_{\Omega\setminus K} |Du|^2 + \lambda(u-u_0)^2(\mathbf{x})d\mathbf{x} + \text{measure}(K).$$

Here K the set of discontinuities of u. The term measure(K) is the area of K in 3D images and its length in 2D images.

The ROF functional is actually a restoration algorithm and does not give a segmentation. The Mumford-Shah functional is highly non convex and not easy to deal with numerically. In addition, it suffers the same drawbacks we mentioned for the mean curvature motion: Since it minimizes the area in three dimensions, it has a tendency to eliminate small tubes and is therefore not fit for the extraction of intricate topological objects.

4 Enhancement by linear diffusion

In this section experiments on common 3D CT diagnostic images, are presented. In general, MR images are used to characterize soft tissue information. Changes in tissue properties can be differentiated by variations in gray scales. It is commonly used to inspect organ morphology and locate tumors. CT images, on the other hand, show clear bone structure with less tissue information. With the presence of radioactive contrast (iodine), detailed arterial tree structures are enhanced and shown as a bright area. This is typically used to screen for arterial disease, vessel occlusion, and malformation. CT imaging modalities support clinical diagnosis; therefore, a large amount of data needs to be analyzed in a timely fashion [39, 28, 9, 31].

4.1 Grady's multi-labeled segmentation

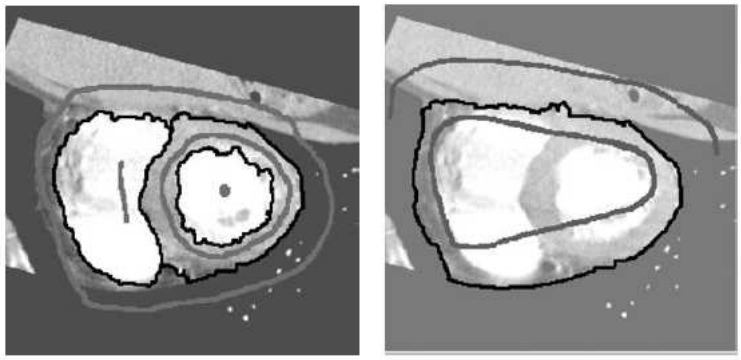

Figure 8: (Taken from [18].) The Grady segmentation method is a linearized sigma filter applied to propagate seed regions. The grey curves are user defined seed region. In that case one of the seed region is put to 1 and the other to zero. A diffusion with sigma filter weights computed on the original image u_0 is applied until a steady state is attained. This gives at each pixel $\mathbf{y}$ a value $p_1(\mathbf{y})$ between 0 and 1 which is interpreted as the probability for $\mathbf{y}$ to belong to the region of the first seed. A threshold at 0.5 gives the black curves separating the regions of both seeds. Like the active contour method, this method is highly dependent on the initial seeds. It is, however, much less sensitive to noise than the snakes method and permits to initialize fairly far from the desired contours. We shall see that by the histogram concentration effect one can get similar or better results without any initialization. See Section 4.1.

In [17], [19], a novel method was proposed for performing multi-label, semi-automated medical image segmentation. Given a small number of pixels with user-defined labels which are called seeds, this method computes the probability that a random walker starting at each unlabeled pixel will first reach one of the pre-labeled pixels. By assigning each pixel to the label for which the greatest probability is calculated, a high-quality image segmentation can be obtained. With each unlabeled pixel, a K-tuple vector is assigned that represents the probability that a random walker starting from this unlabeled pixel first reaches each of the K seed points. A final segmentation may be derived from these K-tuples by selecting for each pixel the most probable seed destination for random walker. By biasing the random walker to avoid crossing sharp intensity gradients, a quality segmentation is obtained that respects object boundaries (including weak boundaries). The image (or volume) is treated as a graph with a fixed number of vertices and edges. Each edge is assigned real-valued

weight corresponding to the likelihood that a random walker will cross that edge (e.g., a weight of zero means that the walker may not move along that edge). By a classical result the probability that a random walker first reaches a seed point exactly equals the solution to the heat equation [3] with boundary Dirichlet conditions at the locations of the seed points, the seed point in question being fixed to unity, while the others seeds are set to zero. This idea was not quite new. Region competition segmentation is an old concept [44]. One can also refer to an algorithm developed for machine learning by Zhu et. al [45] which also finds clusters based upon harmonic functions, using boundary conditions set by a few seed points. [35] also involves weights in the image considered as a graph and takes seed points. The method is also directly related to the Sapiro et al. recent image coloring method by diffusion from seeds [42] (see also [34]).

Let us review the main and simple idea of the diffusion equations on a graph. Each pixel (or voxel) is a graph node and neighboring pixels i and j are linked by a non oriented edge with weight w_{ij}. This weight represents the amount of similarity between pixels i and j, which in the Grady et al. paper is computed as

$$w_{ij}(u) = e^{-\frac{(u(i)-u(j))^2}{h^2}}, \tag{13}$$

where $u(i)$ denotes the grey value at i and h is free parameter left to the user and the only parameter of the algorithm. This is nothing but a sigma filter!

The diffusion algorithm defined by the authors can be viewed in two different ways. First, we can see it as a mere diffusion on the graph of a probability distribution. $K = \bigcup_l K_k$ is a set of seed points. Each set K_k is manually indicated by the user as associated with an image region, with label k. Set $p_k^0(i) = 1$ if $i \in K_k$, $p_k^0(i) = 0$ otherwise and consider the iteration

$$p_k^{n+1}(i) = \frac{\sum_{j\in J(i)} w_{ij}(u) p_k^n(j)}{\sum_{j\in J(i)} w_{ij}} \text{ if } i \notin K, \;\; p_k^{n+1}(k) = p_k^0 \text{ otherwise,} \tag{14}$$

where $J(i)$ is the set of neighboring pixels of i. At the seed points the value $p_k^{n+1}(l)$ remains fixed to 1 in K_k and 0 otherwise. This represents a Dirichlet boundary condition.

Notice that the notions of Markov walk probabilities and diffusion vectors should be well separated: given a Markov matrix M such that $M \cdot 1 = 1$, the left eigenfunctions, ψ_l of M are the asymptotic probabilities of the Markov walk defined on the data by M, and we have $\sum_i \psi_l(i) = 1$. In contrast, the right eigenvectors ψ_l are not probability vectors and are often denoted as diffusion vectors, due to their use in spectral/diffusion embeddings. According to equation (14), this approach is a diffusion.

It is a classical result that p_k^{n+1} converges towards a function p_k such that $0 \leq p_k \leq 1$. The value $p_k(i)$ is interpreted as the *probability that a random walker starting at pixel i will reach k before reaching one of the other pre-labeled pixels $l \in K$*. By choosing for each pixel i the seed realizing $\max p_k(i)$, one obtains a set of regions labeled by $k \in K$ which is the desired segmentation. The steady state p_k is solution of a discrete Laplace equation with Dirichlet boundary condition on the set K of seeds,

$$\Delta_w p_k(i) =: \frac{\sum_{j\in J(i)} w_{ij}(u) p_k^n(j)}{\sum_{j\in J(i)} w_{ij}} - p_k^{n+1}(i) = 0, \;\; p_k(l) = p_k^0(l) \text{ for } l \in K. \tag{15}$$

In [20] a numerical solution is proposed to this numerical problem. Actually the iteration (14) works as well but can be a bit slow.

Thus, the Grady segmentation method is a linearized sigma filter applied to propagate seed regions. Figure 8 taken from [18] illustrates the process on a two chamber view of a cardiac image. The grey curves are user defined seed regions roughly denoting the ventricles in the image. In that case one of the seed regions is put to 1 and the other to zero. A

diffusion with sigma filter weights computed on the original image u_0 is applied until a steady state is attained. This gives at each pixel $\mathbf{y}$ a value $p_1(\mathbf{y})$ between 0 and 1 which is interpreted as the probability for $\mathbf{y}$ to belong to the region of the first seed. In this binary case a single threshold at 0.5 gives the black curves separating the regions of both seeds. Like the active contour method, this method is highly dependent on the initial seeds. It is, however, much less sensitive to noise than the snakes method [8] and permits to initialize fairly far from the desired contours. We shall see that by the histogram concentration phenomenon (Section 4.3) one can get similar or better results without any initialization.

4.2 Proposed diffusion approach

We now focus on 3D imaging. As a result of the discussion of the preceding section our proposed model boils down to a linear algorithm which iterates a fixed average distribution (12) at each point of the 3D volume. This distribution is $\mathbf{x}$-dependent and is computed only once. But this is a huge amount of data when dealing with 3D structures. For this reason the spatial support of the weight distribution is limited by setting $w(\mathbf{x}, \mathbf{y}) = 0$ if $\|\mathbf{x} - \mathbf{y}\| > \epsilon$.

The exponential function computing weights for NL-means and sigma filters is simplified to a threshold function. This threshold strategy reduces the amount of data necessary to represent the probability distribution of each point and actually increases the enhancing effect of the algorithm. Thus,

$$w(\mathbf{x}, \mathbf{y}) = \begin{cases} 1 & \text{if } \|\mathbf{x} - \mathbf{y}\| < \epsilon,\ \|u(B(\mathbf{x}, \rho)) - u(B(\mathbf{y}, \rho))\| < \tau \\ 0 & \text{otherwise} \end{cases}$$

where ϵ denotes the spatial support of the weight distribution, ρ denotes the size of the 3-dimensional comparison neighborhood and τ is the grey level distance threshold. When $\rho = 0$ the weight distribution yields a sigma filter and when $\rho > 0$ the NL-means algorithm (comparing the grey level values in a cube of size $(2\rho + 1)^3$). In the experimentation we have set ρ to one, thus comparing a $3 \times 3 \times 3$ cube. By using this cube we actually compare 27 values, which is enough to take into account the local geometry around each point and to be robust to noise. The parameter epsilon has been set to 3, comparing a 7×7 neighborhood in each slide and 5 slices for each point. This neighborhood is larger inside each slice because the resolution in the third dimension is smaller than in each image. The only parameter which is critical is the filtering parameter τ, which is the same parameter appearing in sigma-filters and mean-shift, and for which the statistical theory doesn't give any specification (in principle it could be fixed at any value and still deliver an optimal asymptotic estimation). In practice this parameter must be fixed empirically for a given medical application and a given apparatus.

For each initial image u the weight configuration is computed at each point of the volume. Depending now on the initial condition u_0 of the iterative process we can enhance the image or propagate initial seeds. Choosing $u_0 = u$ and iterating the above procedure removes the noise and the superfluous details while enhancing the main edges.

If instead of diffusing the original image some initial seeds are propagated the 3D image gets segmented. Assume the user fixes initial values $f(\mathbf{x})$ on a subset $\Omega_0 \subset \Omega$, and $u_0(\mathbf{x}) = f(\mathbf{x})$ for $\mathbf{x} \in \Omega_0$ and 0 elsewhere. In that case, the iterative procedure is only applied to pixels not manually set, which is equivalent to the minimization of the functional

$$\int_{\Omega \times \Omega} (u(\mathbf{x}) - u(\mathbf{y}))^2 w(\mathbf{x}, \mathbf{y}) d\mathbf{x} d\mathbf{y}$$

with boundary constraints $u(\Omega_0) = f(\Omega_0)$. The flexibility of this second approach makes it suitable for several applications including colorization, interpolation and segmentation.

However, we shall see in the experimental section that this method is overcome by an unsupervised one, thanks to the concentration phenomenon explained in Section 4.3.

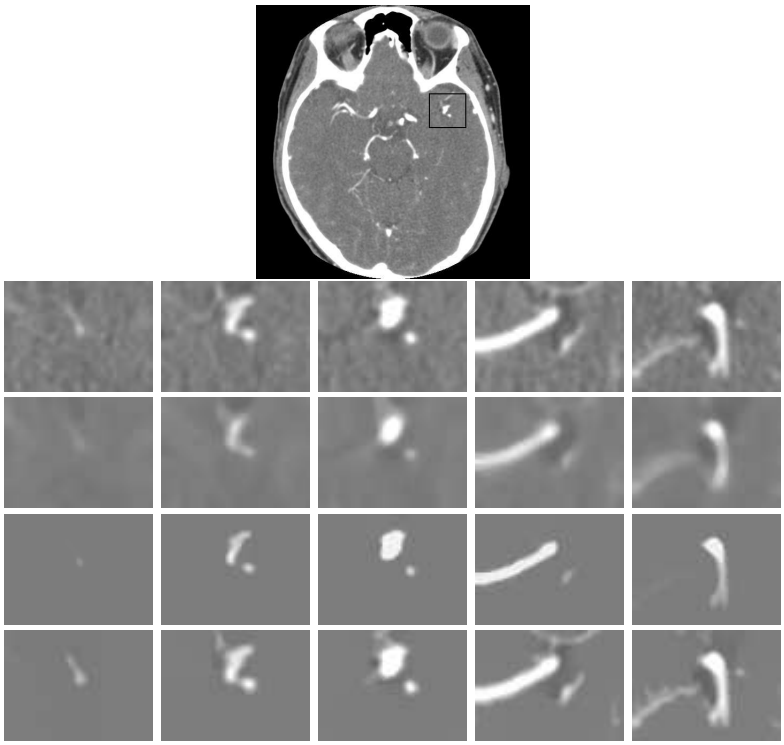

Figure 9: Comparative behavior of discussed methods in 3D. Application to a 3D angiograhy CT image of the head where blood vessels should be segmented. Top: One slice image of the CT volume data with marked interested area. Middle: Display of interest area for several slices of the 3D image. Second row: filtered slices by using median filter. Third row: sigma filter. Fourth row: 3D nonlocal heat equation. Bottom: filtered slices by using the linear method with 3D NL-means weights. The whole sequence has been treated as a 3D image with a weight support of $(5 \times 5 \times 3)$ and a comparison window of $3 \times 3 \times 3$. The background is flattened and blood vessels are enhanced. Thus, a better segmentation is possible by a simple threshold as justified by Figure 12.

4.3 The histogram concentration phenomenon.

Our experiments begin with a computed tomography (CT) angiography image of the brain (see Figure 9, top) where the objective is to extract an object with complex geometry, the cerebral arteries. The challenge is to preserve small surrounding arteries. Also, we want to minimize contour smoothing and maintain the original shape. As displayed in Figure 10, top-left, a simple threshold does not perform the task because of strong oscillations surrounding the vessels. The complexity of the task is extremely well analyzed in Frangi's PhD [15]. This threshold was performed manually trying to get the best visual result. The next two experiments display the extraction of the arteries boundary after an iterated median filter (mean curvature motion) has been applied and the thinning of arteries and disappearance of smaller ones is patent. The third process which was tried is the sigma

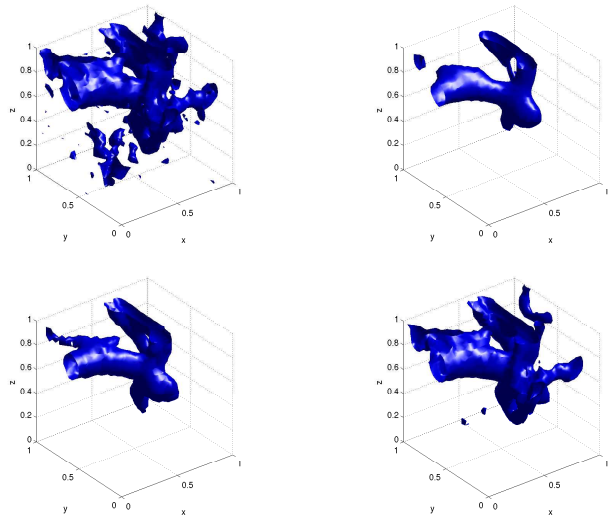

Figure 10: From top to bottom and left to right: original iso-surface of the 3D image, same iso-surface filtered by iterative median filter, by linear sigma filter, and by linear NL-means. The iso-surface extracted from the original image presents many irregularities due to noise. The median filter makes them disappear but makes important parts disappear and some vessels disconnect or fuse. Linear NL-means keeps most vessels and maintains the topology.

filter (with the same weights as in Grady et al. algorithm). The result conserves the topology slightly better, but is far from usable. The fourth image displays the filtered image by the proposed model with NL-means weights and an automatic extraction of the arteries boundaries by unsupervised threshold.

How can such an automatic threshold be found? Figure 9 shows the problem. The top image shows one slice of a 3D CT image with interest area surrounded by a parallelepiped. The next row shows several slices of this area of interest. It can be appreciated, first that the background of arteries has a lot of oscillating clutter and, second, that the grey level value in arteries varies a lot, thus making an automatic threshold problematic. The best way actually to convince oneself that even in this small area a direct threshold wouldn't do the job is to refer to the histograms of Figure 12. The first histogram which is Gaussian-like and poorly concentrated corresponds to the background. The background mode decreases slowly. On the far right part of the histogram one can see a small pick corresponding to very white arteries. The fixing of an accurate threshold in the slowly decreasing background mode is problematic. The top right histogram shows what happens after the application of a median iterative filtering (the mean curvature motion). The histogram does not concentrate at all. The bottom left histogram is obtained after applying the linearized sigma filter described in Section 3.2. The bottom right histogram the one obtained by the linearized NL-means described in the same section. In both cases, one observes that the background mode of the histogram is strongly concentrated on a few grey level values. An automatic threshold is easily fixed by taking the first local minimum after the main histogram peak. This histogram concentration is very similar to the obtained by the mean-shift approach [11] where the neighborhood filter is nonlinearly iterated. In that case, the authors show that clusters tend to its mean yielding piecewise constant image.

The histogram concentration phenomenon is actually visible in the comparative evolu-

tion of some slices under the various considered filters, as shown in Figure 9. The first row shows these slices picked in the interest area. The topology killing effect of the median filter (mean curvature motion) as predicted in Section 2.4 is illustrated on the second row: small arteries tend to vanish and larger ones shrink and become circular as shown in the third slice showing an artery section. The third row is dedicated to the linear sigma filter, which corresponds to Grady's method applied directly to the image instead of using seeds. It is quite apparent that well contrasted objects are well maintained and the contrast augmented, in agreement with the consistency of this recursive filter with the Perona-Malik equation. However, the less contrasted objects tend to vanish because, on them, the evolution becomes similar to an isotropic heat equation. The fourth row is the result of applying the 3D non-local linear heat equation, where the Laplacian coefficients are computed from the original image. The whole sequence has been treated as a 3D image with a weight support of $(7 \times 7 \times 3)$ and a comparison window of $3 \times 3 \times 3$. Clearly the background is flattened and blood vessels are enhanced on this background. A threshold just above the homogeneous made background level should give back arteries, and this indeed occurs. Thus in that case the 3d visualization of objects with complex topology like the cerebral arteries can be achieved by an automatic threshold. The exact segmentation of the artery is a more difficult problem. Even if the histogram is concentrated, a different choice of the visualization threshold can produce slightly different surfaces.

The validity of these conclusions can be checked on the test experiment of Figure 11. This experiment was performed on a simulated section of an artery. The first image shows the grey level section and its central profile. This is a noisy and blurry blob with a tiny boundary contrast with the background, in conformity with real data. The left hand image of the first row displays the image histogram, where the high peak on the left corresponds to the black background and the grey values of the artery are spread out on a wide interval. The second row shows the same display after a Gaussian mean (heat equation) has been performed. This process creates blur and, as expected, the histogram does not discriminate the artery. The very same observation holds with the application on the third row of the sigma-filter. Only the NL-means linearized filter on the fourth row permits actually to detect the artery boundary and to perform an internal diffusion of the color inside it and outside it. The histogram concentration effect occurs and permits to separate neatly figure from background.

4.4 Comparison of multi-date data

The proposed filtering strategy is well adapted to the analysis of the evolution of the same patient. In this case, we are interested on the evolution of the surface and therefore in an automatically extracted and artifact free volume.

The experiment in Figure 14 involves two CT scans of the same patient at a one year interval. The densities of foreground and background have changed. Thus the same threshold cannot be applied to both images to compare structures. However, the automatic application of the threshold procedure yields fairly the same structures. This is due to the histogram concentration phenomenon, which permits to segment automatically both images. The first row shows two slices of the two images with one year interval. Because of a different orientation of the patient's head the slices are not in good correspondence. The interest area, a small parallelepiped, can be approximately picked in both volume images. The linear NL-means algorithm was applied to both images and then the automatic threshold procedure. The iso-surfaces at this threshold value are shown in the second row, with approximately the same perspective. They are in fair correspondence.

4.5 Discussion

The change of dynamic between different images is not a strong objection to the preceding method, since the threshold is fixed by the histogram peak after its concentration has been obtained. However, a previous histogram equalization before processing both images might be useful. Julie Delon [13] demonstrated that the Cox et al. [12] equalization procedure permits to improve drastically the comparison of two images by warping their histograms towards their "midway" histogram.

The results of the experiments of the former section will have to be confirmed by clinical studies. They shouldn't not be taken farther than what the real claims are. We just claimed that a certain procedure, applied to images taken from the same apparatus and the same patient at several dates, can lead to very similar local segmentation and permit comparisons. The interpretation of what is segmented is left to medical doctors. We only claim that the surface extracted could be extracted without supervision, by an automatic threshold. Intuitively, the level surface extracted is one of the more contrasted level surfaces between two different phases, no matter what those phases are from the anatomic viewpoint. Thus, we didn't try to compare with anatomic databases, but only compared on several multi-date images of the same patient. Of course if the average contrast between two organs were null, then no threshold would work.

References

[1] G.A. Agam, S.G. Armato, and C. Wu. Vessel tree reconstruction in thoracic CT scans with application to nodule detection. *Medical Imaging, IEEE Transactions on*, 24(4):486–499, 2005.

[2] D. Barash. Fundamental relationship between bilateral filtering, adaptive smoothing, and the nonlinear diffusion equation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(6):844–847, 2002.

[3] Y. Boykov and M.P. Jolly. Interactive graph cuts for optimal boundary and region segmentation of objects in NDimages. *International Conference on Computer Vision*, 1:105–112, 2001.

[4] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 23(11):1222–1239, 2001.

[5] A. Buades, B. Coll, and J. M. Morel. A review of image denoising algorithms, with a new one. *Multiscale Modeling Simulation*, 4(2):490–530, 2005.

[6] A. Buades, B. Coll, and J.M. Morel. Neighborhood filters and PDE's. *Numerische Mathematik*, 105(1):1–34, 2006.

[7] J.F. Carrillo, M.H. Hoyos, E.E. Dávila, and M. Orkisz. Recursive tracking of vascular tree axes in 3D medical images. *International Journal of Computer Assisted Radiology and Surgery*, 1(6):331–339, 2007.

[8] V. Caselles, R. Kimmel, and G. Sapiro. Geodesic Active Contours. *International Journal of Computer Vision*, 22(1):61–79, 1997.

[9] C. Chen, S.C. Hsieh, W.M. Choi, P. Chiang, J.C.W. Chien, and W.P. Chan. Computed tomography angiography in detection and characterization of ruptured anterior cerebral artery aneurysms at uncommon location for emergent surgical clipping. *Clinical Imaging*, 30(2):87–93, 2006.

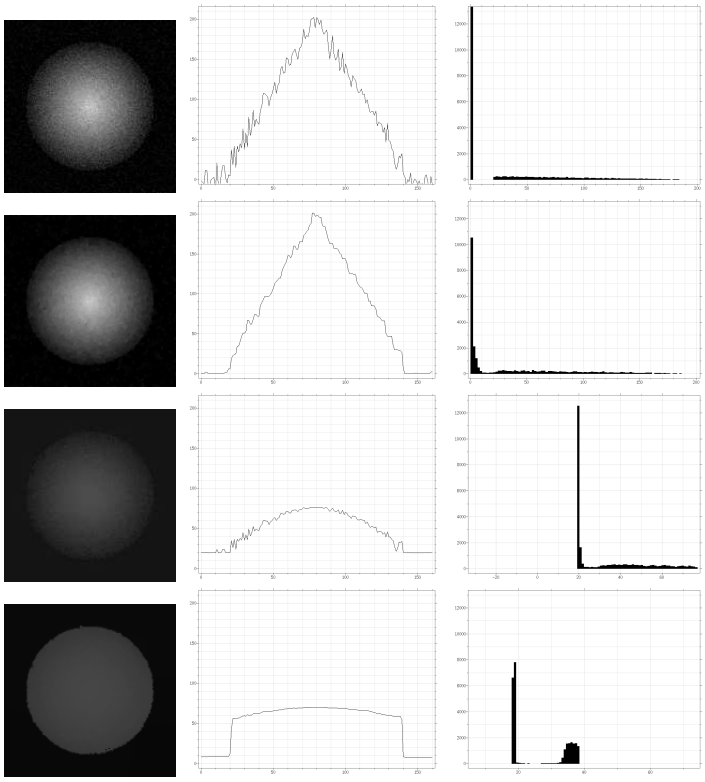

Figure 11: Experiment on a 2D image. From top to bottom: original image, iterative application of a Gaussian mean, iterative application of a median filter, proposed method with linear sigma filter weights and proposed method with linear NL-means weights. On the middle central line of each image. On the right, histogram of the filtered image

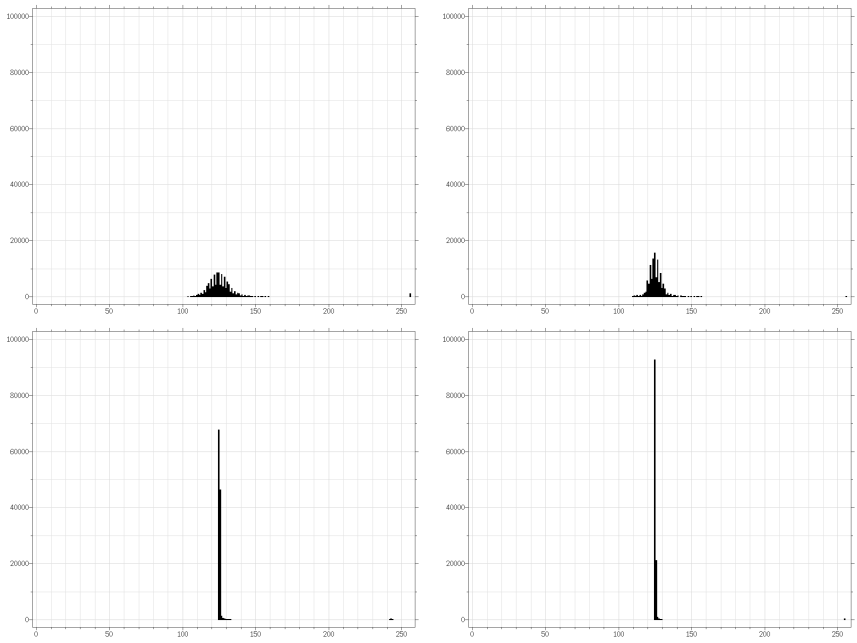

Figure 12: Grey level histogram of 3D areas of interest. Top left: original 3D image before. Top right: after median filtering. Bottom left: after proposed method with sigma filter weights. Bottom right; proposed method with NL-means weights The background is now represented by a few grey level values when the volume is filtered by the proposed method. A threshold can therefore be more easily and automatically applied.

[10] J. Chen and A.A. Amini. Quantifying 3-D vascular structures in MRA images using hybrid PDE and geometric deformable models. *Medical Imaging, IEEE Transactions on*, 23(10):1251–1262, 2004.

[11] D. Comaniciu and P. Meer. Mean shift: a robust approach toward feature space analysis. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(5):603–619, 2002.

[12] IJ Cox, S. Roy, and SL Hingorani. Dynamic histogram warping of image pairs for constant imagebrightness. *Image Processing, 1995. Proceedings., International Conference on*, 2, 1995.

[13] J. Delon. Movie and video scale-time equalization application to flicker reduction. *Image Processing, IEEE Transactions on*, 15(1):241–248, 2006.

[14] A.A. Efros and T.K. Leung. Texture synthesis by non-parametric sampling. *International Conference on Computer Vision*, 2(9):1033–1038, 1999.

[15] AF Frangi, WJ Niessen, and MA Viergever. Three-dimensional modeling for functional analysis of cardiacimages, a review. *Medical Imaging, IEEE Transactions on*, 20(1):2–5, 2001.

[16] G. Gilboa and S. Osher. Nonlocal linear image regularization and supervised segmentation. *Multiscale Modeling Simulation*, 6(2):595–630, 2007.

[17] L. Grady. Random Walks for Image Segmentation. *IEEE Trans. on Pattern Anal. and Mach. Intel.*, 28(11):1, 2006.

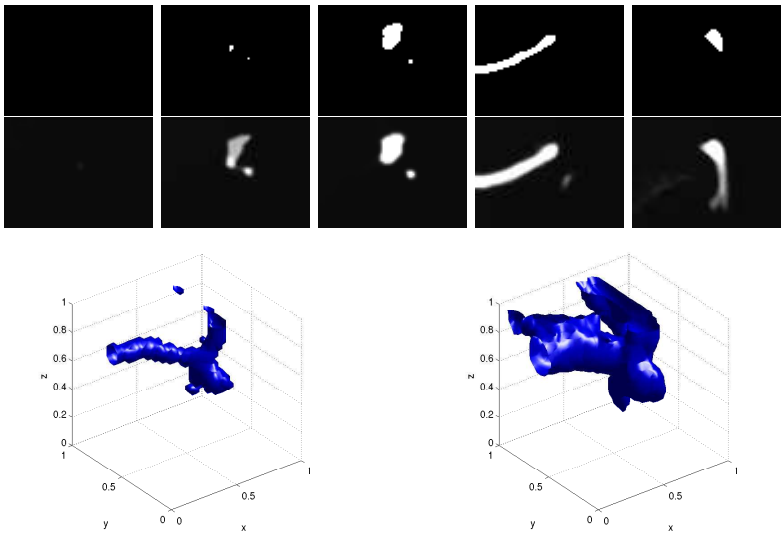

Figure 13: This experiment, to be compared to the former one, shows the result of Grady's seed neighborhood diffusion and compares it to the linearized NL-means (figure 10). The seed region is taken to be the set of all pixels which are nearly white. Five slices of the seed region are displayed on the first row of images. The external surface of the seed region is displayed in the bottom left image. The second row shows the final state of the algorithm and the bottom right image the reconstructed iso-surface. Details (smaller vessels) are lost: compare with Figure 10.

[18] L. Grady and G. Funka-Lea. Multi-label image segmentation for medical applications based on graph-theoretic electrical potentials. *Computer Vision and Mathematical Methods in Medical and Biomedical Image Analysis, ECCV*, pages 230–245, 2004.

[19] L.J. Grady. *Space-variant Computer Vision: A Graph-theoretic Approach*. PhD thesis, Boston University, 2004.

[20] L.J. Grady and E.L. Schwartz. *Anisotropic Interpolation on Graphs: The Combinatorial Dirichlet Problem*. Boston University, Center for Adaptive Systems and Dept. of Cognitive and Neural Systems, 2003.

[21] F. Guichard, J.M. Morel, and Robert Ryan. Contrast invariant image analysis and pde's. *http://www.cmla.ens-cachan.fr/Utilisateurs/morel/JMMBookOct04.pdf*, 1, 2001.

[22] B.B. Kimia, A. Tannenbaum, and S.W. Zucker. On the Evolution of Curves via a Function of Curvature. I. The Classical Case. *Journal of Mathematical Analysis and Applications*, 163(2):438–458, 1992.

[23] C. Kirbas and F. Quek. A review of vessel extraction techniques and algorithms. *ACM Computing Surveys*, 36(2):81–121, 2004.

[24] V. Kolmogorov and R. Zabin. What energy functions can be minimized via graph cuts? *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 26(2):147–159, 2004.

[25] J.S. Lee. Digital image smoothing and the sigma filter. *Computer Vision, Graphics, and Image Processing*, 24:255–269, 1983.

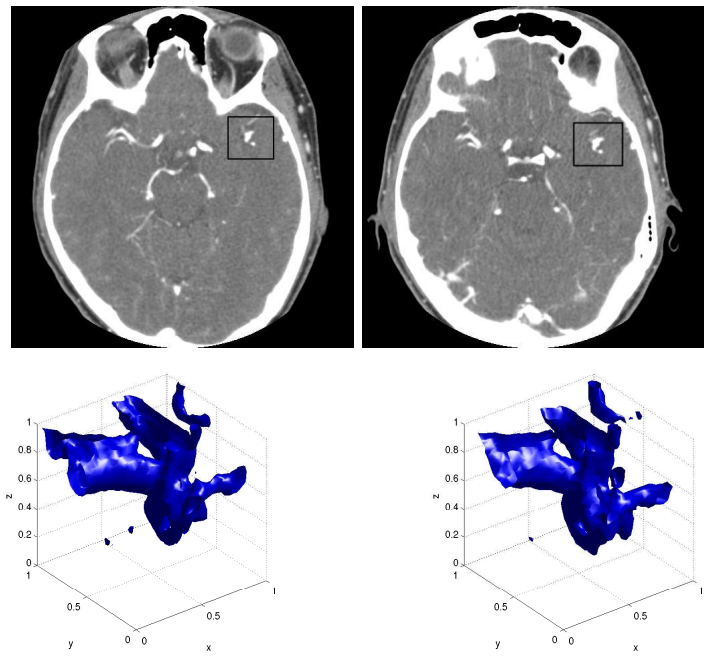

Figure 14: This experiment tests the feasibility of an automatic approach to the evolution analysis of the same patient. It involves two CT scans of the same patient at a one year interval. The densities of foreground and background has changed. Thus the same threshold cannot be applied to both images to compare structures. However, the automatic application of the threshold procedure described in Section 4.3 yields fairly the same structures. This is due to the histogram concentration phenomenon, which permits to segment automatically both images. The first row shows two slices of the two images with one year interval. Because of different orientation of the head the slices are not in good correspondence, but the interest area, a small parallelepiped, can be approximately picked in both volume images. The linear NL-means algorithm was applied to both images and then the automatic threshold procedure. The iso-surfaces at this threshold value are shown in the second row, with approximately the same perspective.

[26] E. Meijering, W. Niessen, J. Weickert, and M. Viergever. Evaluation of Diffusion Techniques for Improved Vessel Visualization and Quantification in Three-Dimensional Rotational Angiography. *Proceedings of the MICCAI*, pages 177–185, 2001.

[27] B. Merriman, J.K. Bence, and S. Osher. *Diffusion Generated Motion by Mean Curvature*. Dept. of Mathematics, University of California, Los Angeles, 1992.

[28] F.E. Moron, M.C. Morriss, J.J. Jones, and J.V. Hunter. Lumps and Bumps on the Head in Children: Use of CT and MR Imaging in Solving the Clinical Diagnostic Dilemma. *Radiographics*, 24:1655–74, 2004.

[29] D. Mumford and J. Shah. *Optimal Approximations by Piecewise Smooth Functions and Associated Variational Problems*. Center for Intelligent Control Systems, 1988.

[30] P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(7):629–639, 1990.

[31] S.R. Pomerantz, G.J. Harris, H.J. Desai, and M.H. Lev. Computed Tomography Angiography and Computed Tomography Perfusion in Ischemic Stroke: A Step-by-Step Approach to Image Acquisition and Three-Dimensional Postprocessing. *Seminars in Ultrasound, CT, and MRI*, 27(3):243–270, 2006.

[32] L. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60(1-4):259–268, 1992.

[33] S. Osher S. Kindermann and P. Jones. Deblurring and denoising of images by nonlocal functionals. *SIAM Multiscale Modeling and Simulation*, 4(4):1091 – 1115, 2005.

[34] G. Sapiro and DL Ringach. Anisotropic diffusion of multivalued images with applications to color filtering. *Image Processing, IEEE Transactions on*, 5(11):1582–1586, 1996.

[35] J. Shi and J. Malik. Normalized Cuts and Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

[36] D. Siablis, G.C. Kagadis, M.T. Karamessini, D. Konstantinou, D. Karnabatidis, T. Petsas, and G.C. Nikiforidis. Intracranial aneurysms: reproduction of the surgical view using 3D-CT angiography. *European Journal of Radiology*, 55(1):92–95, 2005.

[37] S.M. Smith and J.M. Brady. SUSAN-A New Approach to Low Level Image Processing. *International Journal of Computer Vision*, 23(1):45–78, 1997.

[38] C. Swingen, R.T. Seethamraju, and M. Jerosch-Herold. An approach to the three-dimensional display of left ventricular function and viability using MRI. *Int J Cardiovasc Imaging*, 19(4):325–36, 2003.

[39] WJ Thoman, M. Ammirati, LP Caragine Jr, JM McGregor, A. Sarkar, and EA Chiocca. Brain tumor imaging and surgical management: the neurosurgeon's perspective. *Topics in Magnetic Resonance Imaging*, 17(2):121–6, 2006.

[40] C. Tomasi and R. Manduchi. Bilateral filtering for gray and color images. *Proceedings of the Sixth International Conference on Computer Vision*, page 839, 1998.

[41] L.P. Yaroslavsky. *Digital Picture Processing*. Springer-Verlag New York, Inc. Secaucus, NJ, USA, 1985.

[42] L. Yatziv and G. Sapiro. Fast image and video colorization using chrominance blending. *Image Processing, IEEE Transactions on*, 15(5):1120–1129, 2006.

[43] P.J. Yim, J.J. Cebral, R. Mullick, H.B. Marcos, and P.L. Choyke. Vessel surface reconstruction with a tubular deformable model. *Medical Imaging, IEEE Transactions on*, 20(12):1411–1421, 2001.

[44] S.C. Zhu and A. Yuille. Region Competition: Unifying Snakes, Region Growing, and Bayes/MDL for Multiband Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(9):884–900, 1996.

[45] X. Zhu, J. Lafferty, and Z. Ghahramani. *Semi-supervised Learning: From Gaussian Fields to Gaussian Processes*. School of Computer Science, Carnegie Mellon University, 2003.