Sector-based Diffusion Filtering

R. Dargent (1), O. Lavialle (1)(2), S. Guillon (3), P. Baylou (3)

(1) Equipe Signal et Image UMR LAP CNRS 5131, BP 99 33402 TALENCE Cedex FRANCE
(2) ENITA de Bordeaux, BP 201 33175 GRADIGNAN Cedex France
(3) TOTAL TG/EM/SISMAG CSTJF, Av Larribau, 64018 PAU FRANCE

Abstract

In this paper, we propose a new approach devoted to the denoising and the enhancing of strongly oriented 3-D images. In particular, the paper focuses on seismic data composed of a stack of layers disturbed by noise and broken by faults. The denoising of those data is a preprocessing used to improve the detection of the faults.

Our method is based on an anisotropic forward and backward diffusion scheme, which takes advantage of the computation of a “regional” orientation. This approach allows the recovering of the plan which is tangent to the current layer and the corresponding normal direction. Then the diffusion goes forward along the layer in order to smooth the noise, and backward along the normal to separate the layers.

1. Introduction

Non-linear diffusion processes have been widely used during the last decade for edge-preserving denoising. The principle of these approaches was first introduced by Perona and Malik [9] through the following Partial Differential Equation (PDE):

\[
\frac{\partial U}{\partial t}(M,t) = \text{div}(c(M,t) \cdot \nabla U(M,t))
\]

where \( M \) is the current point and \( c(M,t) \) is the diffusivity. \( c(M,t) \) is a decreasing function of the gradient, such as:

\[
c(M,t) = g\left(\|\nabla U(M,t)\|\right) = e^{-\frac{\|\nabla U(M,t)\|^2}{K}}
\]

In (2), \( K \) is a diffusion threshold. Practically, this method is able to remove the noise, while preserving the edges for a long time. Even better, when the contrast is higher than \( K \), the edges are enhanced. Unfortunately, there remain some drawbacks, as shown in [12] and [13], and particularly the presence of a staircasing effect in the case of slow varying edges. Another undesired effect of this model is the pinhole effect [8] which occurs when a group of pixels with intermediate gray level is present in a high contrast zone.

In [3], Catte et al. show that the Perona-Malik process is an ill-posed problem in that it can enhance the noise and produce strong oscillations. To correct that, they propose a regularized model that uses a smoothed gradient:

\[
c(M,t) = e^{-\frac{\|\nabla U(M,t)\|^2}{K}}
\]

where \( G_\sigma \) is a Gaussian kernel of standard deviation \( \sigma \). In that case, Catte and al. prove the existence of a unique and regular solution. The noise and the objects smaller in size than \( \sigma \) are removed.

Another model of diffusion based on mean curvature motion was proposed by Alvarez et al. in [1]:

\[
\frac{\partial U}{\partial t} = g\left(\|\nabla U\|\right)\left[1-h(\|\nabla U\|)\cdot \Delta U + h(\|\nabla U\|) \cdot U_{\xi \xi}\right]
\]

where \( h \) is a smooth non-decreasing function. For gradient values lower than some threshold \( e \), the process acts like an isotropic diffusion. For values higher than \( 2e \), the diffusion is anisotropic, driven by the directional derivative \( U_{\xi \xi} \), where \( \xi \) represents the direction orthogonal to the gradient.

In fact, after a detailed analysis of the classical diffusion processes, as done in [4], [6] and [7], it can be shown that all the 2-D processes can be split into two parts:

\[
\frac{\partial U}{\partial t} = c_\xi U_{\xi \xi} + c_\eta U_{\eta \eta}
\]

where \( \eta \) represents the direction of the gradient. It can be noted that in the 3-D case, the subspace that is orthogonal to the gradient has dimension 2.

So as to simultaneously enhance, sharpen and denoise an image, Gilboa et al. propose in [5] a Forward and Backward scheme which modifies the diffusivity in the equation (1) as follows:

\[
g(\|\nabla U\|) = \frac{1}{1 + \left(\frac{\|\nabla U\|}{K_f}\right)^n} - \frac{\alpha}{1 + \left(\frac{\|\nabla U\| - K_h}{w}\right)^{2m}}
\]

For low values of \( \|\nabla U\| \), the function acts like the one in (2), with \( K_f \) as diffusion threshold, therefore, the diffusion goes forward. For values between \( K_f-w \) and \( K_f+w \), the coefficient is negative (with a magnitude driven by \( \alpha \)), and the diffusion goes backward (as if the time would go...
back). In that last case, instead of smoothing, the process sharpens the image.

In order to get a more directional behaviour of the diffusion process, Weickert proposed in [12], [13] an approach where the scalar diffusivity becomes a diffusion tensor:

\[
\frac{\partial U}{\partial t} = \text{div}\left( D(J_{\rho}(\nabla U_{\sigma})), \nabla U \right) \tag{7}
\]

\[
J_{\rho}(\nabla U_{\sigma}) = G_{\rho}^T(\nabla U_{\sigma} \cdot \nabla U_{\sigma}^T) \tag{8}
\]

\(D\) is a function of the structure tensor \(J_{\rho}\) defined in (8), where \(U_{\sigma}\) is a smoothed version of \(U\). \(D\) has the same eigenvectors as \(J_{\rho}\). Its eigenvalues are chosen and depend on the contrast along the eigendirections. In the case of a non oriented region, the difference between the eigenvalues is low, but it heightens if the region contains anisotropic structures. Thus, the diffusion process occurs along each eigenvector, weighted by the corresponding eigenvalue. The directional driving of the method makes it possible to close even interrupted lines. Unfortunately, it sometimes develops false anisotropic structures.

Other methods have more recently been developed that combine several of the previous methods. For example, in [11], the authors propose an algorithm which takes advantages of both scalar and tensor driven diffusions: indeed, tensor based diffusion is efficient in presence of anisotropic structures, which can even be restored when they are interrupted. But this can also create false structures, where there is none, for example in a noisy background. In the last case, the scalar diffusion gives better results. The idea of the so called mixed-diffusion is then to use the tensor based diffusion when there exists structures, and the scalar one elsewhere. The decision is taken according to a value of confidence in the orientation (in the sense of Rao [10]).

In this paper, we present a method which aim is to filter strongly oriented images, like seismic data. In this case, the data are 3-D blocks of pixels, which one could describe as a stack of parallel surfaces, broken by faults. The surfaces compose a nearly regular alternation of black and white, with a sinusoidal profile. Here, the problem is to smooth along the surfaces, while enhancing the faults. The filtering is a preprocessing and the overall aim of the data processing consists in detecting the faults. Then, the efficiency of the method can be demonstrated using the results obtained on the filtered images by a faults detector developed by TOTAL and based on a coherence measure. As the size of the data can be huge, a particular attention has to be focused on the time consumption of the algorithm.

The method presented in section 2 uses then the available information of regional orientation given at each pixel to define two neighborhoods in which a forward or backward process of diffusion is applied. In Section 3, the efficiency of our approach is illustrated through some experimental results. Finally, in section 4 we present the conclusions and propose some perspectives.

2. Sector-based diffusion filter

The problem of faults detection is an important step of the seismic data analysis. Unfortunately, seismic data often contain noise, which disturbs the faults detector and make it find faults where there is none. The aim of the Sector-based diffusion filter is then to reduce the noise, and to enhance the faults, that is to filter the data so as the faults detector will find less wrong faults.

In order to present the principle of our approach, we will start with a 1-D Perona and Malik diffusion process as explained in subsection 2.1, which we will transpose to the 3-D space. Then, so as to reduce the iteration number, and the computation time, the 6-neighborhood is extended to any size of neighborhood as shown in subsection 2.2. In subsection 2.3 we precise how to obtain a regional orientation, which will be injected in our model in subsection 2.4.

2.1. 1-D filter

In order to build a 3-D diffusion process, we first consider the basic equation (1). Let us write this equation in the 1-D case:

\[
\frac{\partial U}{\partial t} = \text{div}\left( g \left( \frac{\partial U}{\partial x} \right) \frac{\partial U}{\partial x} \right) \tag{9}
\]

which bring us to the discrete 1-D filter:

\[
U_{\text{new}}(x) = U(x) + dt \cdot \frac{g \left( [U_{x+}] \right) \cdot U_{x+} + g \left( [U_{x-}] \right) \cdot U_{x-}}{dx} \tag{10}
\]

with \(U_{\pm} = \frac{U(x \pm dx) - U(x)}{dx}\), \(dx\) being the step of the grid.

2.2. Extension of the neighborhood

First, we extend the previous filter to the case of a 6-neighborhood. To do that, we simply superpose three 1-D diffusion processes, one along each axis, and we get:

\[
U_{\text{new}}(x) = U(x) + dt \cdot \left( \frac{g_u(U_{x+y}) + g_u(U_{x-y})}{dy} + \frac{g_u(U_{y+z}) + g_u(U_{y-z})}{dz} \right) \tag{11}
\]
where \( g_u(s) = g(\|s\|) \cdot s \). Moreover, each \( U \) can be seen as \( \frac{U(M) - U(P)}{PM} \), \( P \) being the point to be updated (that is \( U(P) = U(\delta) \), and \( M \) one of the points of the 6-neighborhood of \( P \). Therefore, (11) can be rewritten:

\[
U^{\text{new}} = U(P) + dt \cdot \sum_{M \in \mathcal{V}(P)} \frac{1}{PM} \cdot g_u \left( \frac{\Delta U_{PM}}{PM} \right) \quad (12)
\]

with \( \Delta U_{PM} = U(M) - U(P) \).

\( V(P) \) is here the 6-neighborhood of \( P \), but it can easily be extended to any neighborhood of \( P \). In fact, the use of larger neighborhoods can be useful to reduce the number of iterations, and so, the computation time. To do this, the idea is to take into account more than 6 points at each iteration, so as, in a way, to fusion many iterations into only one. Therefore, the use of an extended neighborhood seems quite obvious.

### 2.3 Getting orientation

When looking at seismic images, one can observe that the data are composed of a stack of parallel surfaces and so are strongly oriented. The aim is to smooth along the surfaces, while preserving the differences on the normal direction. Thus, the process behavior must be divided into a forward diffusion along the surfaces, and a backward one on the normal. Finally, the problem is to obtain a robust estimation of the dip (orientation of the surfaces).

The local orientation estimation is obtained by the use of a classical gradient, which provides us with a vector orthogonal to the local surface.

In order to smooth this local orientation and get a more regional information, a Principal Component Analysis is computed on a cube centered on the current pixel. Let \( V = \{V_i\}_{1 \leq i \leq n} \) be the field of \( n \) vectors belonging to the PCA support. Then, the regional orientation is obtained as the principal axis of the moment tensor [12]:

\[
M = \frac{1}{n} \sum_{i=1}^{n} V_i V_i^T \quad (13)
\]

This can be seen as a Principal Component Analysis of the autocorrelation matrix of the considered vectors. So, the orientation of the eigenvector corresponding to the highest eigenvalue gives us the direction normal to the surface. It should be noted that the obtained orientation vector includes no information of sign.

### 2.4 Sector division

The regional orientation provides us with the knowledge of the direction that is orthogonal to the local layer. Along that direction, the diffusion must go backward, while it must go forward on the layer. Therefore, the neighborhood of the current pixel is divided into sectors whose pixels will act forward, backward, or not (Fig. 1). The sectors are limited by two coaxial cones whose axis is directed by the first eigenvector of the PCA described above.

The sector 1 wraps the tangential plan, and defines the forward diffusion zone, while the sector 2 is located around the normal direction and defines the backward diffusion zone. The sector 3 is an inert zone constituted of all the other points of the neighborhood which will not act.

Concerning the diffusion equation, the information given by the orientation is introduced into the equation (12) as a coefficient \( c(M) \), whose value is \( C_f > 0 \) into the sector 1, \( C_b < 0 \) into the sector 2, and 0 into the sector 3:

\[
U^{\text{new}} = U(P) + dt \cdot \sum_{M \in \mathcal{V}(P)} \frac{c(M)}{PM} g_u \left( \frac{\Delta U_{PM}}{PM} \right) \quad (14)
\]

To be noted that the division of the neighborhood into sectors has no interest if its size remains 6 pixels. Therefore, the smallest neighborhood that we use is the 26 one corresponding to a 3³ cube.

### 3. Experimental results

The aim of the filtering is to reduce the noise in the seismic data, while preserving the faults, so as to improve the detection of them (see [2]). The efficiency of our filter will then be proved if the faults detector produces a more accurate result. Thus we will use the results obtained by the faults detector developed by TOTAL to evaluate the quality of the filtering.

As the seismic data produced by TOTAL are confidential, we will show here results obtained with synthetic 3D images, composed of a stack of planes,
broken by two faults, a right one and a curved one. Fig 2a shows a front view of the data block. A gaussian white noise was added to this block leading to an SNR of 5dB (Fig 2c). Finally the restored block using our approach is given in Fig 2e. The right part of the Fig 2 presents a top view of the faults blocks respectively computed on the original block (2b), the noisy block (2d) and the restored one (2f).

As can be seen, the smoothing has been done along the layers; however the faults have been enhanced. This allows the faults detector to produce better results, while noisy points, resulting of a misdetection of a fault due to the noise become rare.

4. Conclusion

This paper presents a method of 3-D data filtering, which is able to smooth along a surface, while preserving breakings. This method is particularly efficient with seismic images. The process of filtering itself follows a step of computation of a regional orientation, which indicates for each pixel the dip of the local layer. According to it, the neighborhood is divided into sectors, into which a separated diffusion process is applied: a forward one for smoothing and a backward one for sharpening.

Up to now, the diffusion process has kept the 26 neighborhood. As an extension of the method, and in order to reduce the computation time, future works will be devoted to the extension of the neighborhood, and to the study of the impact that this extension can have on the quality of the results.

5. References