A NEW REGION-BASED PDE FOR PERCEPTUAL IMAGE RESTORATION

Baptiste Magnier, Philippe Montesinos, Daniel Diep

A NEW REGION-BASED PDE FOR PERCEPTUAL IMAGE RESTORATION

Baptiste Magnier[1], Philippe Montesinos[1] and Daniel Diep[2]
[1]*LGi2P de l'Ecole des Mines d'Alès, Parc scientifique Georges Besse, 30035 Nîmes cedex 1*
{*baptiste.magnier, philippe.montesinos, daniel.diep*}*@mines-ales.fr*

Keywords: Image restoration, rotating filters, edge detection, anisotropic diffusion.

Abstract: In this paper, we present a new image regularization method using a rotating smoothing filter. The novelty of this approach resides in the mixing of ideas coming both from pixel classification which determines roughly if a pixel belongs to a homogenous region or an edge and an anisotropic perceptual edge detector which computes two precise diffusion directions. These directions are used by an anisotropic diffusion scheme. This anisotropic diffusion is accurately controlled near edges and corners, while isotropic diffusion is applied to smooth homogeneous and highly noisy regions. Our results and a comparison with anisotropic diffusion methods applied on a real image show that our model is able to efficiently regularize images and to control the diffusion.

1 INTRODUCTION

Partial Differential Equations (PDE's) are widely used in image restoration (Perona and Malik, 1990) (Alvarez et al., 1992) (Weickert, 1998) (Tschumperlé, 2006). Indeed, images are considered as evolving functions of time and PDE's enable to smooth the image while preserving important structures or details (Aubert and Kornprobst, 2006). Filtering techniques like (Nagao and Matsuyama, 1979) (Tomasi and Manduchi, 1998) are not adapted to preserve small object in presence of strong noise.

In (Perona and Malik, 1990) and (Black et al., 1998), diffusion is isotropic on homogenous regions but decreases and becomes anisotropic near boundaries. Diffusion control is done with finite differences so that many contours of small objects or small structures are preserved. However, highly noisy images may generate a lot of undesired artifacts.

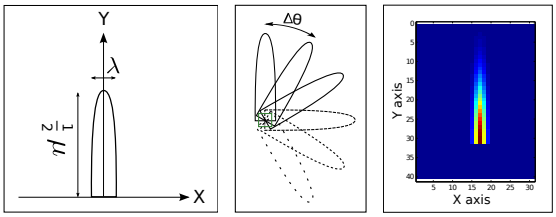
Edge detection is often used to detect image boundaries in order to control a diffusion process and then to preserve contours present in the image in a PDE scheme. The *Mean Curvature motion* method (*MCM*) consists to diffuse only along the contour direction (Catté et al., 1995), even in homogeneous regions. In some diffusion approaches, Gaussian filtering is used for gradient estimation, so the control of the diffusion is more robust to noise (Alvarez et al., 1992) (Weickert, 1998) (Tschumperlé and Deriche, 2005) (Tschumperlé, 2006). The intention is to restrict the diffusion process only along the tangential direction to the gradient near edges and to tune the diffusion using the gradient magnitude. On regions considered as homogenous, the diffusion is isotropic, on the contrary, at edge points, diffusion is anisotropic and inhibited. Nevertheless, it remains difficult to distinguish between high noise and small objects that need to be preserved from the diffusion process. In (Tschumperlé, 2006), the author takes the curvatures of specific integral curves into account in the restoration process. PDE's based on tensor (Weickert, 1998) (Tschumperlé and Deriche, 2005) (Tschumperlé, 2006) are very efficient on noise free-images, but do not provide really homogeneous regions when the noise is high, because the noise creates a *fiber effect* in the image. Indeed, these diffusion methods are adapted for the preservation of thin structures in the image.

In this paper, we present a rotating filter (inspired by (Montesinos and Magnier, 2010), (Magnier et al., 2011c) and (Magnier et al., 2011b)) able to detect homogenous regions and edges regions, even in highly noisy images. Then, we present an anisotropic edge detector which defines two directions for pixels belonging to edges. Finally, we introduce a method for anisotropic diffusion which controls accurately the diffusion near edge and corner points and diffuses isotropically inside noisy homogeneous regions. In particular, our detector provides two different directions on edge and corner points, these informa-

tions enable an anisotropic diffusion in these directions contrary to (Alvarez et al., 1992) where only one direction is considered. In (Magnier et al., 2011c), the authors introduce a new diffusion method to remove the textures. This approache diffuse in two different directions of a contour for each pixel near edges. However, they are not adapted for image restoration because an efficient way of controlling the diffusion is missing. In this paper, we extend this method to restoration. More precisely, the diffusion is controlled both by the gradient value and two directions issued from an anisotropic edge detector (Montesinos and Magnier, 2010).

We first present in section 2 our rotating smoothing filter. A pixel classification using a bank of filtered images is introduced in section 3. Thereafter, we present a anisotropic edge detector based on half smoothing kernels in section 4. Our anisotropic diffusion scheme is introduced in section 5. Section 6 is devoted to experimental results and comparison with other methods. Finally, section 7 concludes this paper.

2 A ROTATING SMOOTHING HALF FILTER



(a) Smoothing filter (b) Rotating filters (c) Discretized filter

Figure 1: A rotating smoothing half filter. For (c): $\mu = 10$ and $\lambda = 1$.

In our method, for each pixel of the original image, we use a rotating half smoothing filter (illustrated in Fig. 1) in order to build a signal s which is a function of a rotation angle θ and the underlying signal. As shown in (Montesinos and Magnier, 2010), (Magnier et al., 2011c) and (Magnier et al., 2011b), smoothing with rotating filters means that the image is smoothed with a bank of rotated anisotropic Gaussian half kernels:

$$G_{(\mu,\lambda)}(x,y,\theta) = C \cdot I_\theta * H(-y) \cdot e^{-\left(\frac{x^2}{2\lambda^2} + \frac{y^2}{2\mu^2} \right)} \quad (1)$$

where I_θ corresponds to a rotated image[1] of orientation θ, C is a normalization coefficient, (x,y) are pixel coordinates, and (μ,λ) the standard-deviations of the Gaussian filter. As we need only the causal part of the filter, we simply "cut" the smoothing kernel by the middle, this operation corresponds to the Heaviside function H and the implementation is quite straightforward.

Some examples of smoothed images using our half kernels $G_{(\mu,\lambda)}(x,y,\theta)$ are available in Fig. 2 (b) and (c).

3 PIXEL CLASSIFICATION

In the following the image is represented as a function defined as :

$$I(x,y) : \mathbb{R}^2 \rightarrow \mathbb{R}.$$

This case corresponds to grey level images.

3.1 Pixel description

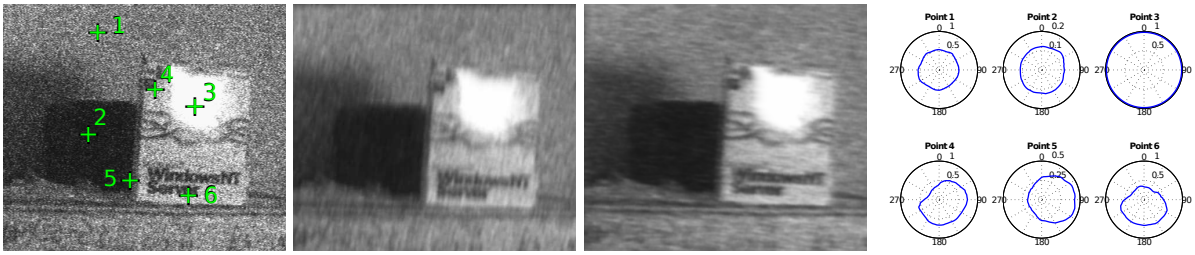
The application of the rotating filter at one point of an image in a 360 scan, provides to each pixel a characterizing signal. In the case of a gray level image, the pixel signal is a single function $s(\theta)$ of the orientation angle θ. Fig. 2(d) is an example of s-functions measured at 6 points located on a noisy image. Each plot represents in polar coordinates the function $s(\theta)$ of a particular point. From these pixel signals, we now extract the descriptors that discriminate edges and regions.

3.2 Flat area detection

The main idea for analyzing a 360 scan signal is to detect significant flat areas, which correspond to homogeneous or noisy regions of the image. Fig. 3(a) shows the pixel signal $s(\theta)$ extracted from a point belonging to a contour. After smoothing, the derivative $s_\theta(\theta)$ is calculated and represented on Fig. 3(b). From $s_\theta(\theta)$, flat areas are detected as intervals (i.e. angular sectors) with a small derivative (close to zero), i.e. sets of values exceeding a given threshold s_{th} in amplitude. Let us note α the largest angular sector. We consider that we detect a flat area when $30 < \alpha < 360$ (degrees).

The noise removal method consists to diffuse isotropically inside homogenous (point 3 of Fig. 2(a))

[1] As explained in (Montesinos and Magnier, 2010), the image is oriented instead of the filter because it increases the algorithmic complexity and allows to use a recursive Gaussian filter (Deriche, 1992).



(a) Points selection in green on a noisy image 420×395

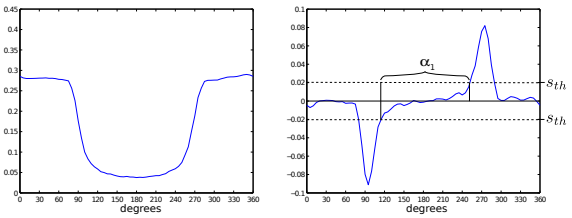
(b) Smoothed image $\theta = 10$ degrees

(c) Smoothed image $\theta = 275$ degrees

(d) Polar representation of $s(\theta)$ for each points of (a)

Figure 2: Points selection and associated signal, $\mu = 10$, $\lambda = 1$ and $\Delta\theta = 5$ (degrees).

and noisy regions (points 1 and 2) i.e. $\alpha = 360$ (degrees). In order to keep sharp contours, the aim of this approach is to diffuse anisotropically at edge points (like points 5 and 6) and corner points (like point 4) so as to preserve borders between regions. Black regions in Fig. 3(c) show regions of flat areas have been detected ($30 < \alpha < 360$ in degrees) in the Fig. 2(a). So this image will be smoothed anisotropically in black regions of Fig. 3(c) and isotropically in white regions. As shown in Fig. 3(c), flat area detection can be seen as a rough edge detection method. In (Magnier et al., 2011c), the curvatures of the signal $s(\theta)$ define two directions used in anisotropic diffusion. These directions are not enough precise for image restoration. Here, we use the directions for the diffusion computed from a new anisotropic edge detector which defines also two directions, but much more precise, resulting in a more precise diffusion.

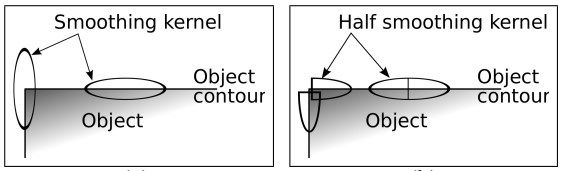


(a) Original signal $s(\theta)$ (b) First derivative of $s(\theta)$

(c) Flat area regions in black

Figure 3: Flat area detections from $s(\theta)$. $\mu = 10$, $\lambda = 1$ and $\Delta\theta = 5$ (degrees).

4 EDGE DETECTION USING HALF SMOOTHING KERNELS



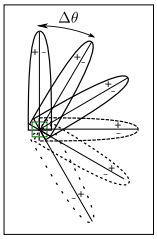
(a) (b)

Figure 4: (a) Full and (b) Half Anisotropic Gaussian kernels at linear portions of contours and at corners.

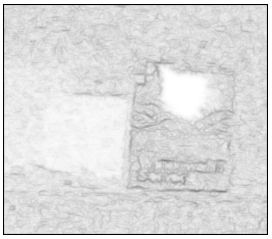
As diagrammed in Fig. 4 (a), steerable (Freeman and Adelson, 1991) (Jacob and Unser, 2004) or anisotropic edge detector (Perona, 1992) perform well to detect large linear structures. However, near corners, the gradient magnitude decreases as the edge information under the scope of the filter decreases. Consequently, the robustness to noise decreases.

A simple solution to bypass this effect is to consider paths crossing each pixel in several directions. The idea developed in (Montesinos and Magnier, 2010) is to "cut" the derivative (and smoothing) kernel in two parts: a first part along a first direction and a second part along a second direction (see Fig. 4 (b). At each pixel of coordinates (x,y), a derivation filter is applied to obtain a derivative information called $Q(x,y,\theta)$:

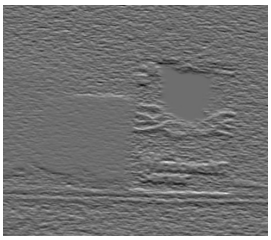
$$Q(x,y,\theta) = I_\theta * C_1 \cdot H(-y) \cdot x \cdot e^{-\left(\frac{x^2}{2\lambda^2} + \frac{y^2}{2\mu^2} \right)} \quad (2)$$



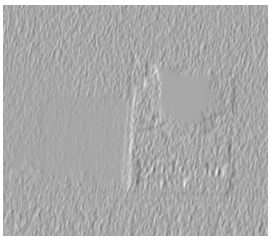
(a) Rotated derivation filter



(b) Gradient image



(c) Image derivative at $\theta = 10^\circ$



(d) Image derivative at $\theta = 275^\circ$

Figure 5: Image in Fig. 2 (a) derivate using different orientations with $\mu = 10$ and $\lambda = 1$. Normalized images.

where C_1 represents a normalization coefficient. As diagrammed in Fig. 6, $Q(x,y,\theta)$ represents the slope of a line issued from a pixel in the perpendicular direction to this line (see Fig. 7(b) for several signals $Q(x,y,\theta)$) issued from several images derivatives (Fig 5). To obtain a gradient $\|\nabla I\|$ and its associated direction η on each pixel P, we first compute global extrema of the function $Q(x,y,\theta)$, with θ_1 and θ_2 (as illustrated in Fig. 7 (c)). θ_1 and θ_2 define a curve crossing the pixel (an incoming and outgoing direction). Two of these global extrema can then be combined to maximize $\|\nabla I\|$, i.e. :

$$\theta_1 = \underset{\theta\in[0,360[}{\arg\max}(Q(x,y,\theta)), \quad \theta_2 = \underset{\theta\in[0,360[}{\arg\min}(Q(x,y,\theta)) \tag{3}$$

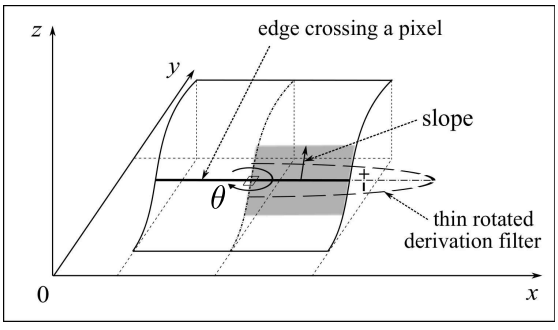
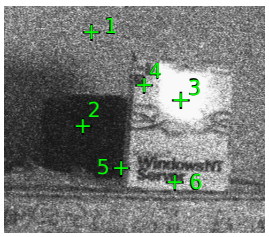


Figure 6: Estimation of the slope turning around a pixel. *The z axis represents the pixel intensity.*

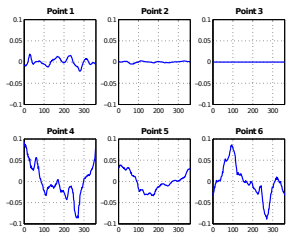
and $\|\nabla I\| = Q(x,y,\theta_1) - Q(x,y,\theta_2)$.

Once obtained $\|\nabla I\|$, θ_1 and θ_2, edges can easily be extracted by computing local maxima of $\|\nabla I\|$ in the direction of the angle $(\theta_1+\theta_2)/2$ followed by an hysteresis threshold (see (Montesinos and Magnier, 2010) for more details). In this paper, we are only interested by the two directions (θ_1,θ_2) and the gradient magnitude used in our diffusion scheme in section 5.2.

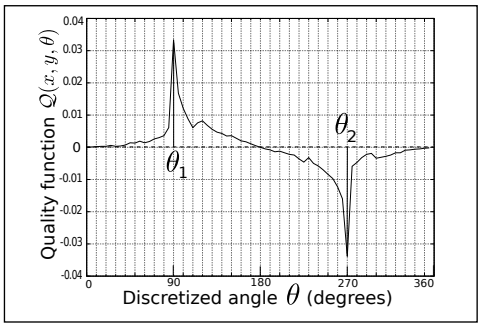
Due to the lengths of the rotating filters, it enables to keep a robustness agains noise and compute two precise diffusion orientations in the directions of the edges. In (Magnier et al., 2011a), the authors have evaluated the edge detection used in this method as a function of noise level.



(a) Points selection in green on image in Fig. 1(d)



(b) $Q(x,y,\theta)$ for each points of (a)



(c) Extrema of a function $Q(x,y,\theta)$

Figure 7: Points selection and its associated $Q(x,y,\theta)$, $\mu = 10$, $\lambda = 1$ and $\Delta\theta = 2^o$.

5 ANISOTROPIC DIFFUSION IN TWO DIRECTIONS WITH PDE

In several diffusion scheme: (Alvarez et al., 1992) (Weickert, 1998) (Tschumperlé and Deriche, 2005) (Tschumperlé, 2006), only one direction is predominantly considered at edges, corner points are treated like edge points, this direction is tangential to the edge. Tensor diffusion schemes preserve edges (Weickert, 1998) (Tschumperlé and Deriche, 2005)

(Tschumperlé, 2006) but in order to remove high noise while preserving contours, the standard deviation of the Gaussian σ must be large. However this solution will blur edges and break corners. For minimizing these effects we are going to consider the two directions (θ_1, θ_2) provided by eq. 3 of the anisotropic edge detector only in areas where flat areas have been detected (Fig. 7(d)).

5.1 Diffusion scheme in two directions

Unlike (Alvarez et al., 1992) (Weickert, 1998) (Tschumperlé and Deriche, 2005) (Tschumperlé, 2006), our control function does not depend only on the image gradient but on a pre-established classification map of the initial image. As stated in section 3, this classification is a rough classification between region and edges. Consequently, a diffusion scheme with only this new control function moves corner points according to the curvature of isointensity lines. As a consequence, this scheme behaves as the MCM scheme (Catté et al., 1995), for example a square is transformed into a circle after some iterations. For minimizing this effect, authors of (Magnier et al., 2011c) consider the two directions ξ_1 and ξ_2 provided by the curvature of the signal $s(\theta)$.

Their diffusion process can now be described by the following PDE:

$$\frac{\partial I_t}{\partial t} = F_A(I_0)\Delta I_t + (1 - F_A(I_0))\frac{\partial^2 I_t}{\partial \xi_1 \partial \xi_2} \qquad (4)$$

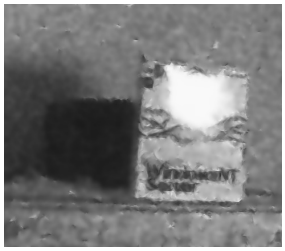
where t is the diffusion time, I_0 is the original image, I_t is the diffused image at time t, (ξ_1, ξ_2) the two directions of the diffusion and F_A represents regions where flat areas are detected (section 3.2) :

- $F_A = 0$ in contours regions
- $F_A = 1$ in homogeneous regions.

However, this diffusion scheme generates a blur effect at edges because that the two directions ξ_1 and ξ_2 of the curvature of $s(\theta)$ are not enough precise than the directions θ_1 and θ_2 computed from the anisotropic edge detector described in section 4.

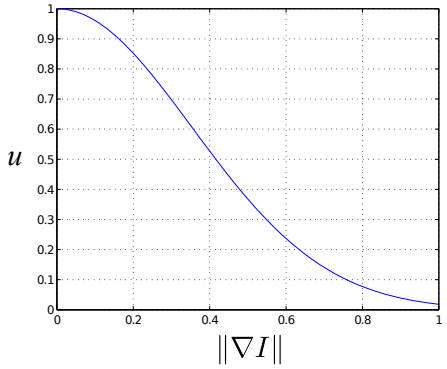
5.2 New perceptual diffusion scheme

In (Alvarez et al., 1992), the aim is to restrict the diffusion process only along the tangential direction to the gradient and tuned by the gradient magnitude. Here, we aim to diffuse only in the θ_1 and θ_2 directions in regions of pixels classified as edge points. Firstly, we control the diffusion in function of the gradient magnitude and secondly in function of the angle

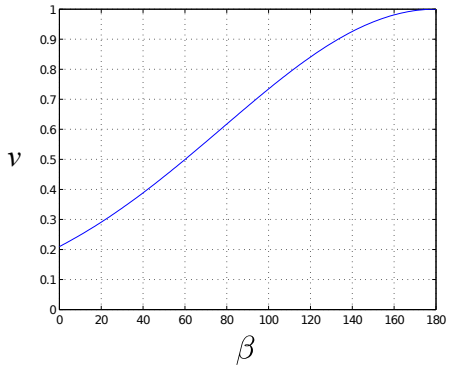


(a) Diffused image without control function

(b) Zoom in (a)



(c) Control function u



(d) Control function v

Figure 8: Diffused image without any control function where blurring effect is too important and the two functions u and v.

between the two directions of the diffusion θ_1 and θ_2. Fig 8 (a) and (b) shows a diffused image without control function where edges are lost and blurred.

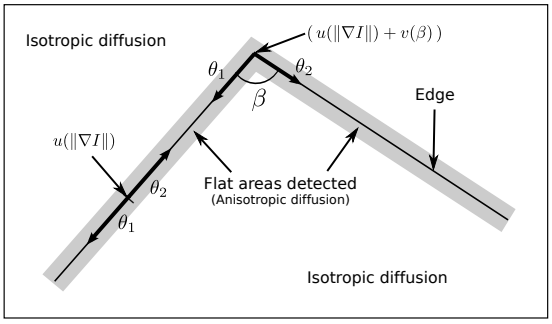
In order to control the diffusion in function of the gradient magnitude, we use the following function u :

$$u(\|\nabla I\|) = e^{-\left(\frac{\|\nabla I\|}{k}\right)^2}, \text{ with } k \in [0,1]. \qquad (5)$$

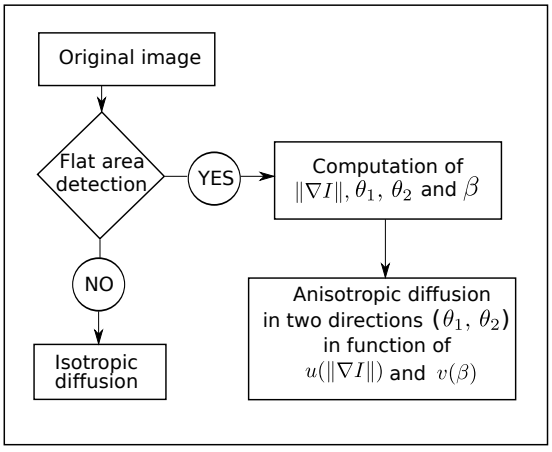
Using the anisotropic perceptual edge detector, we

are able to control the diffusion in function of the angle between θ_1 and θ_2 (see eq. 3) called β. Indeed, at the level of one pixel, the more β is close to 0, the less the diffusion is important. On the contrary, more β is close to 180 (in degrees), more the smoothing is important. The angular control function v can be defined as follows :

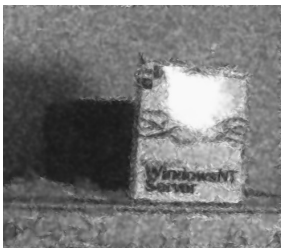
$$v(\beta) = e^{-\left(\frac{180-\beta}{180 \cdot h}\right)^2}, \quad (6)$$



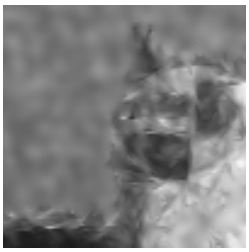
(a) Schema of our diffusion



(b) Organigramm of our method



(c) Diffused image, 15 iterations



(d) Zoom in (c)

Figure 9: Different steps of our diffusion scheme. (c) and (d) correspond to the image in 7 (a) diffused with our method, the blurring effect is small at edges.

with $h \in [0,1]$ and $\beta = \theta_1 - \theta_2\,[180]$ (in degrees).

The new diffusion process presented in Fig. 9 (a) and (b) is described by the following PDE :

$$\frac{\partial I_t}{\partial t} = F_A(I_0)\Delta I_t + (1 - F_A(I_0)) \cdot \left(\frac{u(\|\nabla I\|) + v(\beta)}{2}\right) \frac{\partial^2 I_t}{\partial \theta_1 \partial \theta_2} \quad (7)$$

Contrary to (Alvarez et al., 1992), we do not want to inhibit the diffusion at edges because the two directions of the diffusion θ_1 and θ_2 are sufficiently precise to preserve contours. The values $k = 0.5$ and $h = 0.8$ enable to accurately control the diffusion along edges and corners.

6 EXPERIMENTAL RESULTS

In this section, we present some quantitative and qualitative results. In order to carry out some qualitative results, we have conducted a number of tests with real images. We analyzed the effect of adding a uniform white noise on the original image using the following formula:

$$I_m = (1 - L) \cdot I_0 + L \cdot I_N \quad \text{with} \quad L \in [0,1]\,,$$

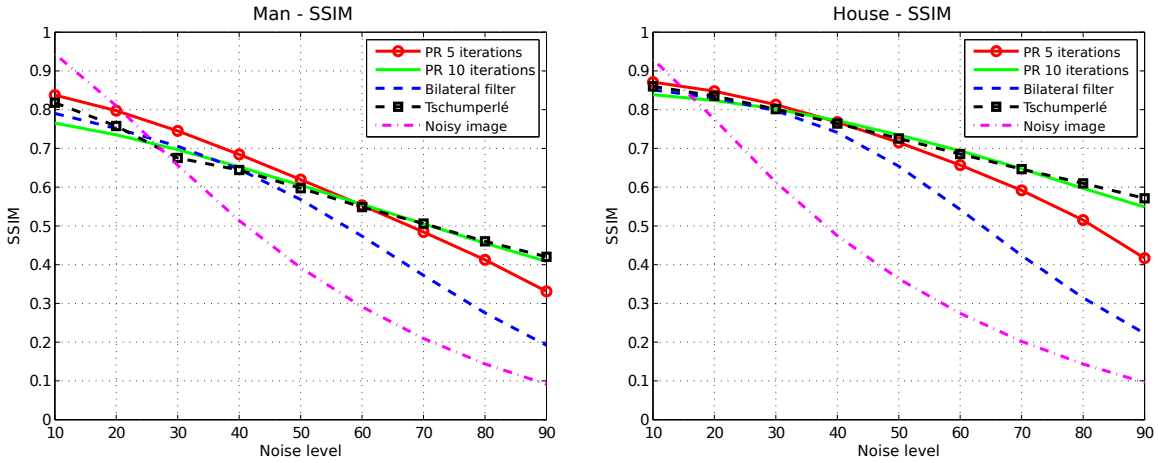
where I_0 is the original image, I_N an image of random uniform noise, I_m the resulting noisy image and L the level of noise.

In our test, we performed an anisotropic diffusion and compared our results with several methods.

6.0.1 Qualitative results

We have called our approach PR for Perceptual image Restoration. In the images presented in Fig. 11 (b) and 12 (b), the aim is to smooth the noise present in the different images preserving all objects. We used our scheme with $\mu = 5$, $\lambda = 1$ and $\Delta\theta = 5^\circ$ for flat area detection and the value of the threshold in amplitude s_{th} is equal to 0.05. Parameters used in anisotropic edge detector in order to compute (θ_1, θ_2) are $\mu = 5$, $\lambda = 1.5$ and $\Delta\theta = 2^\circ$. The result of our anisotropic diffusion is presented in the Fig. 11 (h), (i) and 12 (h), (i) after 5 or 10 iterations.

We compare our result with several approaches as well as the well known Nagao (Nagao and Matsuyama, 1979) and bilateral filters (Tomasi and Manduchi, 1998). We compare also with other PDE's approaches of Alvarez et al. (Alvarez et al., 1992), Perona-Malik (Perona and Malik, 1990) and Tschumperlé (Tschumperlé, 2006). For the bilateral filter, Nagao filter and Perona-Malik method, results



(a) Image in Fig. 11 (a)

(b) Image in Fig. 12 (a)

Figure 10: SSIM evolution in function of the noise level.

are noisy in the different images (Fig. 11 and 12 (c), (d), and (f)). The approaches of Alvarez et al. and Tschumperlé remove the noise but blur edges.

6.0.2 Influence of noise

Curves have been plotted on Fig. 11 (j), (k) and 12 (j), (k), they present respectively the evolution of the PSNR (*Peak Signal to Noise Ratio*) and RMSE. We have compared our result after 5 and 10 iterations with the bilateral filter, and the method of Tschumperlé. The bilateral filter is used with two iterations and the algorithm of Tschumperlé with 20 iterations and a standard deviation of the gaussian $\sigma = 1$ for each test. For low levels of noise ($L < 0.5$), our approach and this of Tschumperlé perform well in terms of this two measures. After 10 iterations our result seems better than 5 iterations when the noise is high. However, PSNR and RMSE measures are not fully consistent with human eye perception, that is why we have measured a similarity metric.

6.0.3 Perceptual evolution

Instead of calculating a local difference like the PSNR which does not take into account the perception of the image, it exists a measure that estimates the similarity between two images. This metric called SSIM (*Structural SIMilarity*) measures similar structures between two images (Wang et al., 2004). It is based on several windows of the image.

The measure will yield values between zero and one, when the result is close to one, the structures are considered as very well preserved. The total measure of SSIM on the whole image is given by the average SSIM on all windows.

Thus, we conducted SSIM measures plotted in Fig. 10 in function of the level of noise on the two images shown in Fig. 11 (a) and 12 (a). This measure reflects better the preservation of the contours.

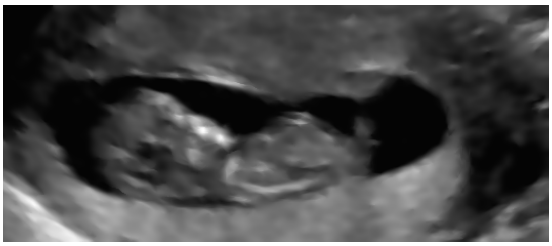
We observe, for our method, when the noise is less than fifty percent in the image, compared to 10 iterations, 5 iterations keeps the shape better but when it exceeds this threshold, 10 iterations are necessary. The difference in the SSIM decreases between 5 and 10 iterations for noise less than fifty percent when the image is composed of fine textures on the image of the house in Fig. 12 (a). Our approach removes the fine textures and diffuse isotropically, resulting in a loss of information compared to the original image.

Nevertheless, when the image is composed mainly of small objects, the result of the SSIM seems better for our approach with five iterations for the images in Fig. 11 (a). On the contrary, an image composed of large objects, like the house in Fig. 12 (a), SSIM values crosses from a noise level of around forty percent for 5 or 10 iterations. Beyond, 10 iterations are necessary.

The bilateral filter gives good results when the noise is small but the SSIM values decrease rapidly. In general, Tschumperlé's scores are better than the bilateral filter. However, our method with 10 iterations gives the best results in terms of SSIM when the noise level is between 30 and 70 percent (above 70 percent, the noise is too hard to compare such measures).

Fig. 13 shows a result of an ultrasound image of a fetus composed of a high noise. After diffusion, edges

(a) Original image 533×231



(b) Restored image

Figure 13: Ultrasound image.

of the fetus are very sharped and the noise is totally removed.

7 CONCLUSIONS

We have proposed in this paper a new method for diffusing the noise in images by pixel classification using a rotating smoothing filter followed by a PDE. Our classification method seems very promising as we have been able to classify correctly homogenous regions and edge regions for various image types. The diffusion process that we have developed precisely combines isotropic and anisotropic diffusion, and enables to keep edges and corners of different objects in highly noisy images. Comparing our results with existing algorithms allows us to validate our method. As introduced in Fig. 13, next on our agenda is to enhance this method for medical images.

REFERENCES

Alvarez, L., Lions, P.-L., and Morel, J.-M. (1992). Image selective smoothing and edge detection by nonlinear diffusion, ii. *SIAM Journal of Numerical Analysis*, 29(3):845–866.

Aubert, G. and Kornprobst, P. (2006). *Mathematical problems in image processing: partial differential equations and the calculus of variations (second edition)*, volume 147. Springer-Verlag.

Black, M., Sapiro, G., Marimont, D., and Heeger, D. (1998). Robust anisotropic diffusion. *IEEE Transactions on Image Processing*, 7(3):421–432.

Catté, F., Dibos, F., and Koepfler, G. (1995). A morphological scheme for mean curvature motion and applications to anisotropic diffusion and motion of level sets. *SIAM J. Numer. Anal.*, 32:1895–1909.

Deriche, R. (1992). Recursively implementing the gaussian and its derivatives. In *International Conference On Image Processing. A longer version is INRIA Research Report RR-1893*, pages 263–267.

Freeman, W. T. and Adelson, E. H. (1991). The design and use of steerable filters. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13:891–906.

Jacob, M. and Unser, M. (2004). Design of steerable filters for feature detection using canny-like criteria. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(8):1007–1019.

Magnier, B., Montesinos, P., and Diep, D. (2011a). Fast Anisotropic Edge Detection Using Gamma Correction in Color Images. In *IEEE 7th International Symposium on Image and Signal Processing and Analysis*, pages 212–217.

Magnier, B., Montesinos, P., and Diep, D. (2011b). Ridges and valleys detection in images using difference of rotating half smoothing filters. In *Advances Concepts for Intelligent Vision Systems*, pages 261–272.

Magnier, B., Montesinos, P., and Diep, D. (2011c). Texture Removal in Color Images by Anisotropic Diffusion. In *International Conference on Computer Vision Theory and Applications (VISAPP 2011)*, pages 40–50.

Montesinos, P. and Magnier, B. (2010). A New Perceptual Edge Detector in Color Images. In *Advances Concepts for Intelligent Vision Systems*, pages 209–220.

Nagao, M. and Matsuyama, T. (1979). Edge preserving smoothing. *CGIP*, 9:394–407.

Perona, P. (1992). Steerable-scalable kernels for edge detection and junction analysis. *IMAVIS*, 10(10):663–672.

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

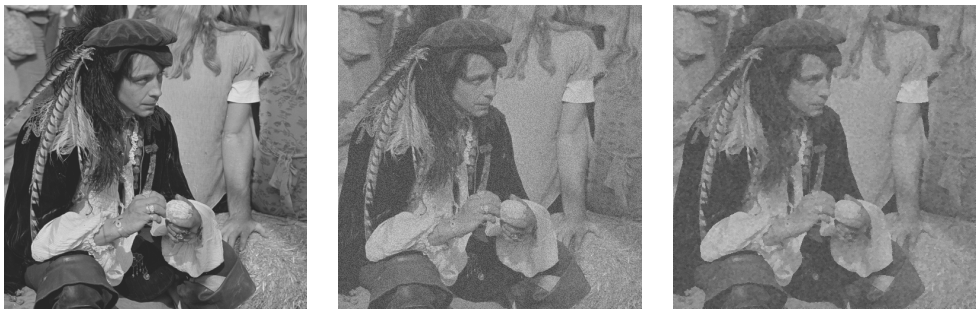
Tomasi, C. and Manduchi, R. (1998). Bilateral filtering for gray and color images. In *International Conference on Computer Vision*, pages 839–846. IEEE.

Tschumperlé, D. (2006). Fast anisotropic smoothing of multi-valued images using curvature-preserving PDE's. *International Journal of Computer Vision*, 68(1):65–82.

Tschumperlé, D. and Deriche, R. (2005). Vector-valued image regularization with pdes: A common framework for different applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 506–517.

Wang, Z., Bovik, A., Sheikh, H., and Simoncelli, E. (2004). Image quality assessment: From error visibility to structural similarity. *Image Processing, IEEE Transactions on*, 13(4):600–612.

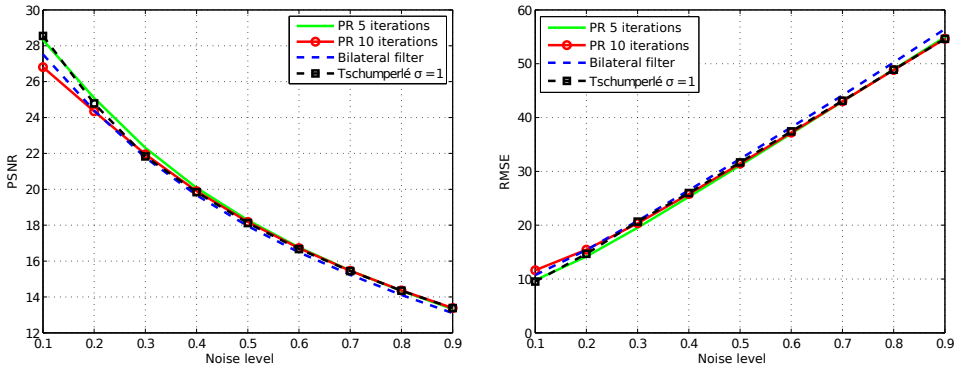
Weickert, J. (1998). Anisotropic diffusion in image processing. *Teubner-Verlag, Stuttgart, Germany*.



(a) Original image 256×256 (b) Noisy image, $L = 0.7$ (c) Nagao filter

(d) Perona-Malik, 100 iterations (e) Alevarez et al. 20 iterations (f) Bilateral filter 2 iterations

(g) Tschumperlé, 20 iterations, σ=1 (h) Our algorithm 5 iterations (i) Our algorithm 10 iterations



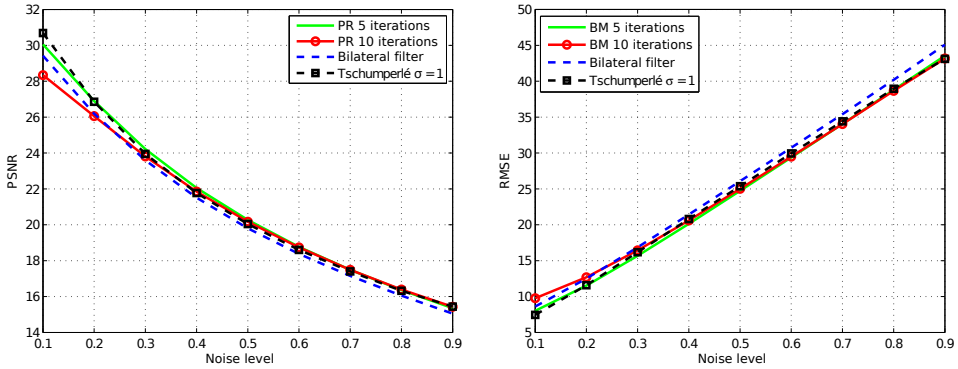
(j) PSNR evolution (k) RMSE evolution

Figure 11: Image restoration on an image containing small objects.

(a) Original image 256×256 (b) Noisy image, $L = 0.7$ (c) Nagao filter

(d) Perona-Malik, 100 iterations (e) Alevarez et al. 20 iterations (f) Bilateral filter 2 iterations

(g) Tschumperlé, 20 iterations, σ=1 (h) Our algorithm 5 iterations (i) Our algorithm 10 iterations



(j) PSNR evolution (k) RMSE evolution

Figure 12: Image restoration on an image containing large structures.