TEXTURE REMOVAL BY PIXEL CLASSIFICATION USING A ROTATING FILTER

Baptiste Magnier, Philippe Montesinos, Daniel Diep

TEXTURE REMOVAL BY PIXEL CLASSIFICATION USING A ROTATING FILTER

Baptiste Magnier, Philippe Montesinos, Daniel Diep

Ecole des Mines d'ALES,
LGI2P,
Site EERIE, Parc Scientifique G.Besse
30035 Nimes Cedex 1
{Baptiste.Magnier,Philippe.Montesinos,Daniel.Diep}@mines-ales.fr

ABSTRACT

In this paper, we present a new method for removing texture in images using a smoothing rotating filter. From this filter, a bank of smoothed images provides pixel signals able to classify a pixel as a texture pixel, a homogenous region pixel or an edge pixel. Then, we introduce a new method for anisotropic diffusion which controls accurately the diffusion near edge and corner points and diffuses isotropically inside textured regions. Several results applied on real images and a comparison with anisotropic diffusion methods show that our model is able to remove the texture and control the diffusion.

Index Terms— Smoothing filter, rotating filter, texture, image segmentation, anisotropic diffusion.

1. INTRODUCTION

The textural analysis has been a domain of active research for almost forty years [4] because texture brings many problems for image segmentation [1]. The texture is related to the spatial distribution or statistical grayscale intensity and contains important information about the structural arrangement of image surfaces and their relationships with their direct environment. It is easy to a human observer to recognize a texture [5], however it remains difficult to precisely define and analyze it automatically [9]. This difficulty is reflected by the high number of different definitions of the texture. In this paper, we do not analyze the texture but simply identify and diffuse anisotropically it [11] so that it appears as an uniform region.

In image restoration, edge detection is often used to detect image boundaries in order to control a diffusion process. For example, in [10], images derivatives alongs four directions are computed providing edge informations. On homogenous regions, the diffusion is isotropic, on the contrary, at edge points, diffusion in inhibited. In [6], diffusion is isotropic on homogenous regions but decreases and becomes anisotropic near boundaries. Control is done with finite differences so many contours of small objects or small structures are preserved. In [10], Gaussian filtering is used for gradient estimation, so the control of the diffusion is more robust to noise. Nevertheless, it remains difficult to distinguish between noise or texture and small objects that need to be preserved by the diffusion process.

In this paper, we present a rotating filter (inspired by [8]) able to detect textures. Secondly, we introduce a new method for anisotropic diffusion which controls accurately the diffusion near edge and corner points and diffuses isotropically inside the textured regions. In particular, our detector provides two different directions on edges or corners, these informations allow an anisotropic diffusion in these directions contrary to [6] where only one direction was considered.

We first present in Section 2 our rotating smoothing filter. A new pixel classification using a bank of filtered images is introduced in Section 3. Our anisotropic diffusion scheme is introduced in Section 4. Section 5 is devoted to experimental results and Section 6 concludes this paper.

2. ROTATING FILTERS

In our method, for each pixel of the original image, we use rotating filters [8] in order to build a signal s in function of a rotation angle θ. Smoothing rotating filters means that the image is smoothed with a bank of rotated anisotropic Gaussian kernels :

$$G_\theta(x,y) = C.H\left(P_\theta\begin{pmatrix} x \\ y \end{pmatrix}\right)e^{-\begin{pmatrix} x & y \end{pmatrix}P_\theta^{-1}\begin{pmatrix} \frac{1}{2\sigma_1^2} & 0 \\ 0 & \frac{1}{2\sigma_2^2} \end{pmatrix}P_\theta\begin{pmatrix} x \\ y \end{pmatrix}}$$

where C is a normalization coefficient, P_θ a rotation matrix of angle θ, x and y are pixel coordinates and σ_1 and σ_2 the standard-deviations of the Gaussian filter.

As we need only the causal part of the filter (illustrated on Fig. 1 (a)), we simply "cut" the smoothing kernel in the middle, it corresponds to the Heaviside function H. By convolution with these rotated kernels (see Fig. 1(b)), we obtain a collection of directional smoothed images $I_\theta = I * G_\theta$.

For computational efficiency, we just proceed in a first time in a rotation of the image at some discretized orientations from 0 to 360 degrees (of $\Delta\theta = 1, 2, 5,$ or 10 degrees,

depending on the precision needed and the smoothing parameters) before applying non rotated smoothing filters with σ_η and σ_ξ the standard-deviations of the Gaussian filter (illustrated on Fig. 1(a)). As the image is rotated instead of the filters, the filtering implementation is quite straightforward. In a second time, we apply an inverse rotation of the smoothed image and obtain a bank of $360/\Delta\theta$ images.
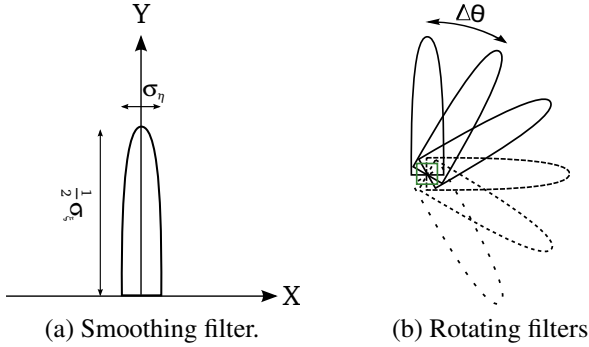

Fig. 1. Description of smoothing rotating filters.

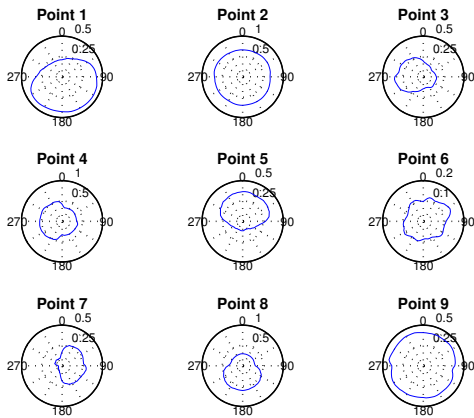## 3. PIXEL CLASSIFICATION

3.1. Pixels signals

Fig. 2. Flat cartesian representation of $s(\theta)$ for points indicated by a cross in Fig. 5(a) with a discretization angle $\Delta\theta$ equal to 5 degrees with $\sigma_\eta = 1.5$ and $\sigma_\xi = 10$.

Applying the rotating filter at one point of an image and making 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 is an example of s-functions measured at 9 points located on the image of Fig. 5(a). Each plot of Fig. 2 represents in polar coordinates the function $s(\theta)$ of a particular point. From these
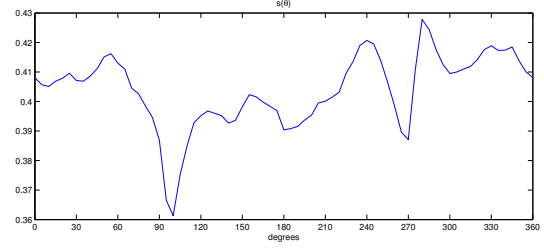

Fig. 3. Flat cartesian representation of $s(\theta)$ at the point 9 which corresponds to a pixels inside texture region.

pixel signals, we now extract the descriptors that will discriminate edges and regions.

In the case of a pixel in a homogeneous region, $s(\theta)$ will be constant (see Fig. 2 point 2). On the contrary, in a textured region, $s(\theta)$ will be stochastic (as illustrated in Fig. 3). In the case where the pixel lies between several different regions, $s(\theta)$ will contain several flat areas (see point 1 on Fig. 5(a), $s(\theta)$ is illustrated in Fig. 2 in polar coordinates and in Fig. 4(a) for a cartesian representation).

3.2. Flat area detection
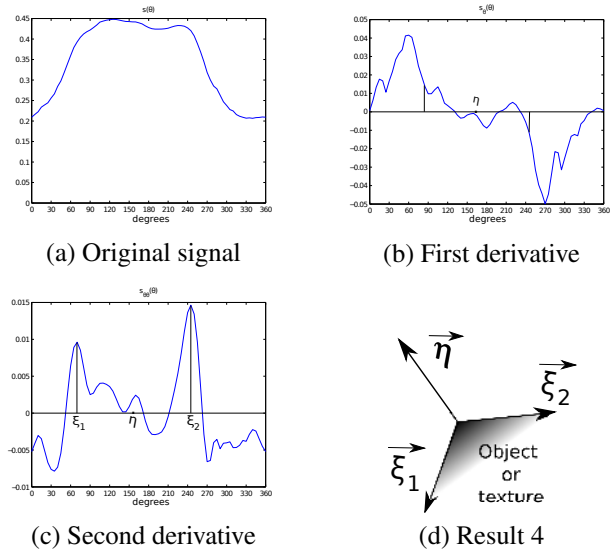
Fig. 4. A signal and its derivatives to compute η, ξ_1 and ξ_2 directions.

The main idea for analyzing a 360 scan signal is to detect significant flat areas, which correspond to homogeneous regions of the image. Fig. 4(a) shows the pixel signal $s(\theta)$ extracted from point 9 of Fig. 5(a). This particular point is located at the limit of two regions (the edge of a bush on the image). After smoothing, the derivative $s_\theta(\theta)$ is calculated and represented on Fig. 4(b), and so is the second derivative

$s_{\theta\theta}(\theta)$ represented on Fig. 4(c). From the derivative $s_{\theta}(\theta)$, flat areas are detected as intervals (i.e. angular sectors) with a null derivative, their median direction noted η corresponds to the gradient direction. From the second derivative $s_{\theta\theta}(\theta)$, we can extract the directions ξ_1 and ξ_2 which delimit the flat area detected. ξ_1 and ξ_2 are calculated as the directions of maximum (or minimum) curvature (see Fig. 4(c)). As illustrated on Fig. 4(d), they are the directions of a smoothed edge curve crossing the considered pixel (entering and leaving directions). They will be used by a simple anisotropic diffusion scheme that will be presented in the next section.

The method will consist to diffuse isotropically inside homogenous (point 2) and textured regions (points 6 and 9) and diffuse anisotropically on directions ξ_1 and ξ_2 at edge and corner points (points 1, 3 and 4).

4. OUR ANISOTROPIC DIFFUSION SCHEME

As stated in the introducing section, we want to design a a smoothing process able to remove texture, preserve edges and smooth homogeneous regions. Like many restoration schemes (for example [6], [10]) which can be interpreted from a geometrical point of view, we are going to define a diffusion scheme which will take into account the pixel classification established in the previous section.

Basically we want to smooth isotropically inside textured and homogeneous regions whereas we want to diffuse anisotropically on and near boundaries. The first idea could be to use a heat equation on textured and homogeneous regions and a Mean Curvature Motion (MCM) [3] scheme on edge points, leading to a diffusion scheme described by equation 1.

$$\frac{\partial I_t}{\partial t} = F(I_0)\Delta I_t + (1 - F(I_0))\frac{\partial^2 I_t}{\partial \xi^2} \qquad (1)$$

where :
  t is the diffusion time,
  ξ is the direction of diffusion (contour tangent),
  I_0 is the original image,
  I_t is the diffused image at time t,
  $F(I_0)$ is our classification function.

But unlike [6], our classification function $F(I_0)$ does not provide us with a precise control on image boundaries, then the MCM scheme here takes an important part and moves corner points according to the curvature of iso-intensity lines. As a consequence, this scheme behaves as the MCM scheme then for example a square is transformed into a circle after some iterations. For minimizing this effect we are going to consider the two directions ξ_1 and ξ_2 provided by our pixel classification process on image boundaries.

The new diffusion process can be now described by the new following equation :

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

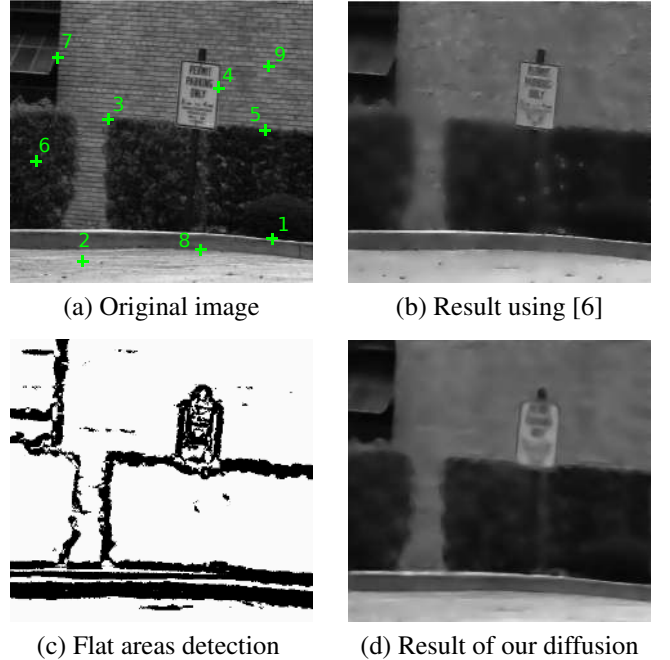on which diffusion is driven by the two directions ξ_1 and ξ_2.

5. RESULTS

(a) Original image   (b) Result using [6]

(c) Flat areas detection   (d) Result of our diffusion

Fig. 5. Result 1.

(a) Edge detection on the original image   (b) Edge detection on the diffused image
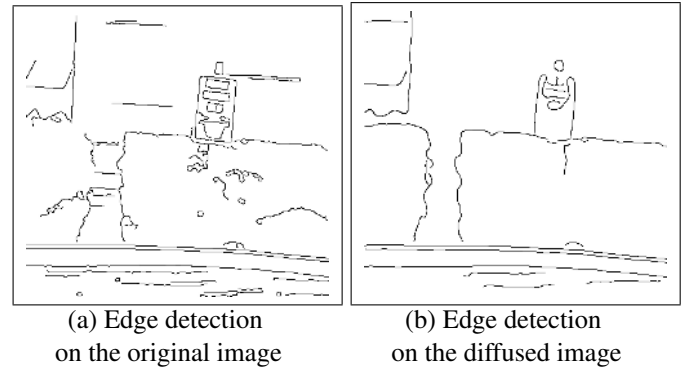
**Fig. 6**. Edges detections comparison.

We present results obtained on real images using our detector and compare them with other methods.

In the first image (Fig. 5(a)), the aim is to smooth the different textures present in the image (wall, bushes) preserving all objects (windows, panel, sidewalk). We used our detector with $\sigma_\xi = 10$, $\sigma_\eta = 1.5$ and a discretization angle of

5 degrees. Fig. 5(c) shows flat areas results : a black pixel matches at least one flat area, whereas white pixels denote that no flat area has been detected. In others words, they correspond to a pixel lying between two regions (contours), between a homogeneous region and texture region or a corner. The result of the anisotropic diffusion is presented in the Fig. 5(d) after 50 iterations. Note that different objects are perfectly visible whereas textures regions are smoothed and some of them have merged.

We compare our result with the method proposed in [6] after 100 iterations and a K parameter equal to 0.03 (illustrated in Fig. 5(b)). We can note that texture has not been completely removed on the wall and that bushes boundaries are not correctly preserved. If we change the K parameter or the iteration number, we will never obtain the desired result.

In order to show the efficiency of our method for texture removal, we compare edge detection on the original image and on the image obtained after the diffusion. Fig. 6 shows the difference between an edge detection [2] on the original image (Fig. 6(a)) and the diffused image (Fig. 6(b)) with the same edge detection parameters (standard deviation of the Gaussian $\sigma = 2$). Edge detection on the diffused image is less noisy than on the original image. Moreover, edges of bushes, panel, sidewalk and windows appear clearly, whereas edge detection on the original image fails detecting some bushes and wall contours.

(a) Original image   (b) Our diffusion result
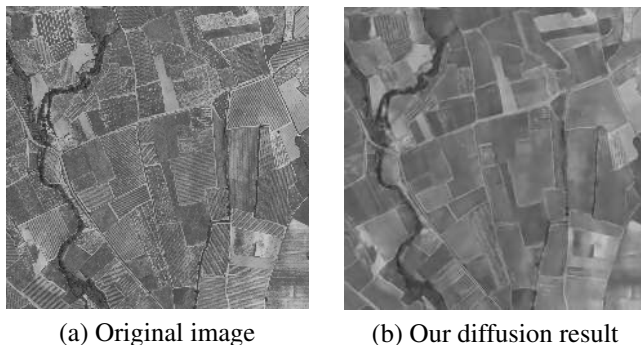
Fig. 7. Result 2.

The second picture (Fig. 7(a)) is an aerial photography of vineyard areas. The aim is to separate vineyard areas from roads. We used our detector with $\sigma_\xi = 5$, $\sigma_\eta = 1$ and a discretization angle of 5 degrees. The result of the anisotropic diffusion is presented in the Fig. 7(b) after 50 iterations. This result achieves correctly our objective which was to smooth vineyard regions while preserving roads.

An image data base with results is available on-line [7].

6. CONCLUSION

We have proposed in this paper a new method to remove texture in images by pixel classification using a rotating smoothing filter. Comparing our results with existing algorithms allows us to validate our method. Our classification method seems very promising as we have been able to classify correctly texture region, homogenous region and edge regions (small ribbon located centered at the edge). Next on our agenda is to extend this approach to color images and enhance this method for image restoration.

7. REFERENCES

[1] P. Arbelaez, M. Maire, C. Fowlkes and J. Malik, "From Contours to Regions: An Empirical Evaluation," in *Computer Vision and Pattern Recognition*. IEEE, 2009, pp. 2294-2301.

[2] Canny, J. F.: A Computaional Approach to Edge Detection. In: IEE Transaction on Pattern Analysis and Machine Intelligence. 8(6), 679–698, (1986)

[3] F. Catté, F. Dibos, and G. Koepfler, "A Morphological Scheme for Mean Curvature Motion and Applications to Anisotropic Diffusion and Motion of Level Sets", *SIAM Journal on Numerical Analysis*, vol. 32(6), pp. 1895-1909, 1995.

[4] R. M. Haralick, "Statistical and structural approaches to texture," Proc. IEEE, 1979, vol. 67(5), pp. 786-804.

[5] B. Julesz, Textons, the elements of texture perception and their interactions, *Nature*, vol. 290, pp. 91-97, 1981.

[6] P. Lions, L. Alvarez and J. M. Morel, "Image selective smoothing and edge detection by nonlinear diffusion," in *SIAM Journal of Numerical Analysis*, vol 29(3), pp 845-866, 1995.

[7] Magnier, B., Montesinos, P.: Texture removal results. http://www.lgi2p.ema.fr/~magnier/ Research/Texture_Removal/Demos.html

[8] P. Montesinos and B. Magnier, "A New Perceptual Edge Detector in Color Images," in *Advanced Concepts for Intelligent Vision Systems*. 2010.

[9] N. Paraagios and R. Deriche "Geodesic active contours for supervised texture segmentation," in *Computer Vision and Pattern Recognition*. IEEE, 1999, pp. 2422-2427.

[10] P. Perona and J. Malik, "Scale space and edge detection using anisotropic diffusion," in *Transactions on Pattern Analysis and Machine Intelligence*. 1990, vol 12(7), pp 629-639.

[11] Weickert, J.: Anisotropic diffusion in image processing, Citeseer, (1998)