COLOR IMAGE EDGE DETECTION BY ROBUST ANISOTROPIC DIFFUSION

Harold I. A. Bustos, Hae Yong Kim.
Dept. Eng. Sistemas Eletrônicos, Universidade de São Paulo.
E-mails: {haroldivan, hae}@lps.usp.br.

ABSTRACT

Edge detection is an important operation in many image-processing applications. Anisotropic diffusion is one of most reliable edge detection methods. There are many anisotropic diffusion techniques for grayscale images. However, there are only few works on the diffusion for color images, and these all use the traditional diffusion function defined by Malik-Perona. Recently, "robust anisotropic diffusion" was proposed for grayscale images. This method is based on Tukey's robust estimator, and it converges much faster than traditional Malik-Perona's diffusion. Consequently, the new technique better preserves edges and attenuates undesirable noises. So, the edge detection becomes even more reliable. In this paper, we propose to use the Tukey's estimator to detect edges in color images. This technique is executed through two independent diffusion processes. In the first, the complex chromaticity function is diffused. The second process diffuses the scalar intensity. The results of these two diffusions are combined to detect edges. We have compared the obtained results with the Malik-Perona's conventional technique. The new method indeed converges faster, yielding sharper edges.

1. INTRODUCTION

The linear scale-space of a grayscale image G is the series of images obtained by filtering G with Gaussian low-pass filters with different standard deviations. The scale-space allows us to observe and process image G in different scales. Some image-processing algorithms are better to execute in a rough scale. Once a rough solution is found, a sharp image is processed to obtain a more accurate solution.

Originally, the anisotropic diffusion was proposed to obtain an alternate scale-space, by using anisotropic diffusion instead of the Gaussian filter. The nonlinear scale-space so obtained presents a very interesting property. Important edges remain sharp even in rough resolutions, while small edges are filtered out. A parameter can adjust which edges must be considered "important" and which must be considered "marginal."

Surprisingly, the anisotropic diffusion used to obtain nonlinear scale-space was found to be one of most reliable edge detectors. Surely, the edge detection is a very important image-processing operation. Thus, many anisotropic diffusion techniques have been proposed for grayscale images [1, 2, 3, 4].

The anisotropic diffusion diffuses grayscale values in homogenous regions, thus attenuating noises and eliminating marginal edges. Meanwhile, it inhibits the diffusion in borders of objects, what sharpens important edges. Consequently, any simple segmentation and/or edge detection algorithms can be used afterwards, because an image pre-processed by anisotropic diffusion will have clear edges and few noises.

In the literature, there are some works on anisotropic diffusion for color and multi-spectral images [5, 6]. These works use the anisotropic diffusion process originally defined by Malik and Perona. Recently, Black et al. [2] discovered that the Tukey's robust estimator converges faster than Malik-Perona's function, yielding a more reliable and faster edge detection for grayscale images. A grayscale image processed by Tukey's diffusion is considerably sharper than the one processed by Malik-Perona's diffusion.

In this paper, we use the anisotropic diffusion based on Tukey's function to detect edges in color images. We compare the results of our method with the results yielded by Malik-Perona's diffusion.

Our method first transforms the color image into HSI color space. Then, hue and saturation is regarded together as a chromaticity complex field, while the intensity is considered a scalar field. We perform two independent anisotropic diffusions: one on the chromaticity complex field and another on the luminosity scale field. The results of two diffusions are merged together to detect edges.

This paper consists of four sections. We present in second section the representation of color images. We describe in section 3 the complex anisotropic diffusion and edge detecting algorithms. In section 4, we present the obtained results. Finally, in section 5 we present our conclusions.

2. REPRESENTATION OF COLOR IMAGE

A color image C is represented in RGB color space as

$$C(x) = [R(x), G(x), B(x)] \in \mathbb{R}^3, \quad x \in \mathbb{Z}^2,$$

where $R(x)$, $G(x)$ and $B(x)$ are respectively red, green and blue channels, normalized between 0 and 1. Many image-processing textbooks (as [7]) describe the conversion technique from RGB color space into HSI. After the conversion into HSI color space, the image C is represented as

$$C(x) = [\vartheta(x), \sigma(x), I(x)] \in \mathbb{R}^3, \quad x \in \mathbb{Z}^2,$$

where $\vartheta(x)$, $\sigma(x)$ and $I(x)$ are respectively hue, saturation and intensity channels. Hue is an angle and it is usually denoted in radians. The saturation and intensity are normalized to range from 0 to 1.

Hue $\vartheta(x)$ and saturation $\sigma(x)$ channels form together the complex chromaticity $k(x)$. Saturation represents modulus and hue represents angle of complex number [5]:

$$k(x) = \sigma(x)\exp(j\vartheta(x))$$

Hue $\vartheta(x)$ is defined in the interval [0, 2π), where 0 means red, $2\pi/3$ means green and $4\pi/3$ means blue. According to the circular nature of hue, the difference between two hues $\vartheta(x_1)$ and $\vartheta(x_2)$ must be defined as:

$$\min\left[\left|\vartheta(x_1)-\vartheta(x_2)\right|,\, 2\pi-\left|\vartheta(x_1)-\vartheta(x_2)\right|\right].$$

3. COLOR ANISOTROPIC DIFFUSION AND EDGE DETECTION

3.1. Robust anisotropic diffusion

In order to perform diffusion, the time variable t is added to definitions. Thus, the complex chromaticity should be defined as:

$$k(x,t) = \sigma(x,t)\exp(j\vartheta(x,t)),$$

where $t \in \mathbb{R}$ represents the time parameter or the number of iterations. The robust anisotropic diffusion of chromaticity $k(x,t)$ is performed through the partial differential equation below [2, 5]:

$$\frac{\partial k(x,t)}{\partial t} = \operatorname{div}\left[c(x,t)\nabla k(x,t)\right], \qquad (1)$$

where *div* and ∇ denote respectively divergent and gradient operators. The function

$$c(x,t) = f\left(\left|\nabla k(x,t)\right|\right)$$

controls the rate of diffusion according to the gradient modulus. The classic diffusion function was defined by Malik-Perona. It decreases monotonically with the gradient magnitude. For chromaticity diffusion, the Malik-Perona's function should be defined as:

$$f_1\left(\left|\nabla k(x,t)\right|\right) = \frac{1}{1+\left(\frac{|\nabla k(x,t)|}{\gamma}\right)^2}, \quad \gamma = 2\delta^2 .$$

where γ is the contrast parameter, that is, the threshold to decide whether the diffusion will be performed or not, and δ is the scale regularization parameter [3]. The gradient of complex chromaticity $k(x,t)$ and its magnitude are defined:

$$\nabla k(x,t) = \left(\nabla\sigma(x,t) + j\sigma\nabla\vartheta(x,t)\right)\exp\left(j\vartheta(x)\right) \text{ and}$$

$$\left|\nabla k(x,t)\right| = \sqrt{\left|\nabla\sigma(x,t)\right|^2 + \sigma^2(x,t)\left|\nabla\vartheta(x,t)\right|^2} .$$

Note that in modulus of chromaticity gradient, the variation of hue is weighted by the saturation. This rule conforms to the fact that the less saturated a color the less hue is psychologically perceived.

Recently, Black et al. [2] proposed the use of Tukey's robust estimator for the grayscale diffusion. For chromaticity diffusion, it can be written as:

$$f_2\left(\left|\nabla k(x,t)\right|\right) = \begin{cases} \dfrac{1}{2}\left[1-\left(\dfrac{|\nabla k(x,t)|}{\delta}\right)^2\right]^2, & \text{if } |\nabla k(x,t)| < \delta \\ 0, & \text{otherwise} \end{cases},$$

We have used the discretization described in [2] to numerical implementation of equation (1):

$$\vartheta_s^{(t+1)} = \vartheta_s^{(t)} + \frac{\lambda}{|\eta_s|}\sum_{p\in\eta_s} f_1\left(\left|\nabla k(x,t)\right|\right)\nabla\vartheta_{s,p} \qquad (2)$$

where $\vartheta_s^{(t)}$ is the hue of pixel s in iteration step (t); λ is the diffusion rate (for example, 1.0); and η_s is the set of neighbors of pixel s (4-neighborhood or 8-neighborhood are the most usual). As saturation and hue are mixed up in the modulus of chromaticity gradient, these two components together determine the rate of diffusion. The parameter $\nabla\vartheta_{s,p}$ represents the gradient of hue at pixel s in relation to neighboring pixels $p\in\eta_s$. Saturation is discretized similarly:

$$\sigma_s^{(t+1)} = \sigma_s^{(t)} + \frac{\lambda}{|\eta_s|}\sum_{p\in\eta_s} f_1\left(\left|\nabla k(x,t)\right|\right)\nabla\sigma_{s,p} \qquad (3)$$

In this case, $\sigma_s^{(t)}$ represents the saturation of pixel s at iteration (t). The parameter $\nabla\sigma_{s,p}$ is the gradient of saturation at pixel s in relation to the neighboring pixels $p\in\eta_s$.

The corresponding equations to be used in Tukey's diffusion are:

$$\vartheta_s^{(t+1)} = \vartheta_s^{(t)} + \frac{\lambda}{|\eta_s|}\sum_{p\in\eta_s} f_2\left(\left|\nabla k(x,t)\right|\right)\nabla\vartheta_{s,p} \qquad (4)$$

$$\sigma_s^{(t+1)} = \sigma_s^{(t)} + \frac{\lambda}{|\eta_s|}\sum_{p\in\eta_s} f_2\left(\left|\nabla k(x,t)\right|\right)\nabla\sigma_{s,p} \qquad (5)$$

The scale regularization parameter δ used both in $f_1(|\nabla k(x,t)|)$ and in $f_2(|\nabla k(x,t)|)$ should be chosen according to the intrinsic properties and domain of values of the two functions, since they have different threshold levels to identify edges.

The diffusion of intensity or luminosity $l(x, t)$ is performed by an analogous process. The intensity is filtered by equation (1), replacing complex chromaticity $k(x, t)$ by scalar intensity $l(x, t)$. The discretization is also similar to that of chromaticity. Actually, the intensity diffusion process is a grayscale anisotropic diffusion, since it is executed completely independent from chromaticity. Finally, the three diffused components are mixed up and the final processed image is obtained.

3.2. Edge detection

The anisotropic diffusion pre-processes the image in order to facilitate the edge detection. However, it does not directly detect edges. To find edges, another algorithm must be executed. We have used a simple edge detection algorithm based on the magnitude of the gradient. If the magnitude is superior to a threshold value, the pixel is considered to belong to an edge. Note that the diffusion was performed in pixels whose gradient magnitudes were inferior to the threshold value. So, pixels with gradient magnitudes above this threshold value must be considered as edges. Consequently, to detect edges after Malik-Perona's diffusion, the following inequalities must be computed:

$$\begin{cases} \nabla l(x,t) > \delta\sqrt{2} \\ \nabla \vartheta(x,t) > \delta\sqrt{2} \\ \nabla \sigma(x,t) > \delta\sqrt{2} \end{cases}$$

where $\delta\sqrt{2}$ is the threshold value. A pixel is considered to belong to edge if at least one of the three inequalities holds. The same procedure was used to Tukey's diffusion, changing the threshold from $\delta\sqrt{2}$ to δ.

4. RESULTS

Tukey's and Malik-Perona's anisotropic diffusions for color images have been implemented and tested to find out the advantages and disadvantages of each method.

Malik-Perona's diffusion was iterated 500 times, using scale parameter δ=1660 for complex chromaticity and δ=80 for scalar intensity. Afterwards, three HSI components were mixed to reconstruct the color image. Finally, the edge detection algorithm was applied.

Tukey's diffusion was also iterated 500 times, but scale parameter was δ=1000 for complex chromaticity and δ=22 for intensity.

Figures 1a, 2a and 3a correspond to the hue, saturation and intensity of original image. Figures 1b, 2b and 3b correspond to the same images processed by Malik-Perona's diffusion. Hue and saturation (figures 1b and 2b) were diffused together in the chromaticity complex field, while intensity (figure 3b) was processed separately. Figures 1c, 2c and 3c are the outputs of hue, saturation and intensity images processed by Tukey's diffusion. Once again, hue and saturation were diffused together, while intensity was diffused independently.

Figures 4a, 4b and 4c correspond respectively to: the original color image; the color image diffused by Malik-Perona's process, obtained by merging figures 1b, 2b and 3b; and the color image diffused by Tukey's process, obtained by merging figures 1c, 2c and 3c.

Note in figures 4b and 4c that Malik-Perona's diffusion blurs edges, no matter if the edge is important or not. Meanwhile, Tukey's diffusion sharpens the principal edges while filtering out marginal edges. Malik-Perona's diffusion can erase important edges if iterated many times, but Tukey's diffusion does not erase any significant edges.

When the average quantity of diffusion gets close to zero, we say that the diffusion process has terminated. The Malik-Perona's diffusion never terminates, what causes edge blurring as the number of iterations (time parameter *t*) increases. The Tukey's diffusion terminates after some iteration, that is, the average quantity of diffusion becomes very close to zero. This statement is confirmed by the average quantity of diffusion in three HSI components (figures 6a, 6b and 6c).

Figure 6a depicts the average quantity of hue diffusion. Tukey's diffusion is depicted in blue and Malik-Perona's in red. In Tukey's method, the diffusion is 0.000000 from iteration 31 on. Meanwhile, for Malik-Perona's method the average diffusion is 3.103196 in iteration 500. This means that the diffusion process has not converged even after 500 iterations.

The convergence of saturation is also faster using Tukey's technique. Figure 6b depicts the average saturation diffusion. Using Tukey's method, the diffusion is 0.000000 from iteration 61 on. Using Malik-Perona's, the average diffusion is 0.566333 even at iteration 500.

Two diffusions apparently converge for the intensity component. However, the convergence is faster using Tukey's technique. Using Tukey, the diffusion is 0.000000 from iteration 46 on. Using Malik-Perona, the diffusion is 0.000000 from iteration 50 on.

From the facts observed above, we can conclude that Tukey's diffusion terminates once edges are detected. Meanwhile, Malik-Perona's diffusion never terminates. Thus, edges are better enhanced using Tukey in all three HSI components, and consequently the edge detection becomes more reliable using Tukey's diffusion.

5. CONCLUSIONS

In this paper, we have presented a new anisotropic diffusion method based on the Tukey's robust estimator for edge detection in color images. The edge detection in color images processed using Tukey's method was faster and trustier than Malik-Perona's.

6. ACKNOWLEDGEMENTS

The authors would like to express their gratitude to FAPESP for the partial financial support of this work under grant 2001/02400-9.

REFERENCES

[1] Hong Shan Neoh and Guillermo Sapiro, "Using Anisotropic Diffusion of Probability Maps for Activity Detection in Block-Design Functional MRI," *Proc. IEEE Int. Conf. Image Processing*, vol. 1, pp. 621-624, 2000.

[2] M. J. Black, G. Sapiro, D. H. Marimont, and D. Heeguer, "Robust Anisotropic Diffusion," *IEEE T. Image Processing*, vol. 7, no. 3, March 1998.

[3] G. Gilboa, Y. Y. Zeevi, N. Sochen, "Image Enhancement Segmentation and Denoising by Time Dependent Nonlinear Diffusion Processes," *Proc. IEEE Int. Conf. Image Processing*, vol. 3, pp. 134-137, 2001.

[4] H. W. Kim, J. H. Jang, K. S. Hong, "Edge-Enhancing Super-Resolution Using Anisotropic Diffusion," *Proc. IEEE Int. Conf. Image Processing*, vol. 3, pp. 130-133, 2001.

[5] L. Lucchese and S. K. Mitra, "Color Segmentation Through Independent Anisotropic Diffusion of Complex Chromaticity and Lightness,'' *Proc.*

IEEE Int. Conf. Image Processing, vol. 2, pp. 746-749, paper 2614, 2001.

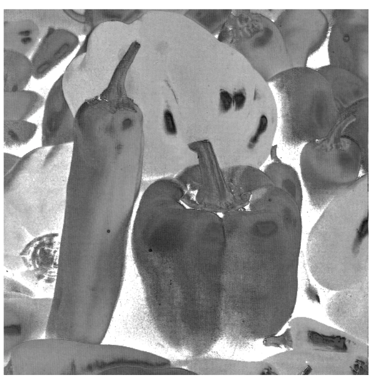
[6] B. Tang and G. Sapiro, "Chromaticity Diffusion," *Proc. IEEE Int. Conf. Image Processing*, vol. 2, pp. 784-790, 2000.

[7] Gonzalez and Woods, *Digital Image Processing*, Addison Wesley Publishing Company, 1992.

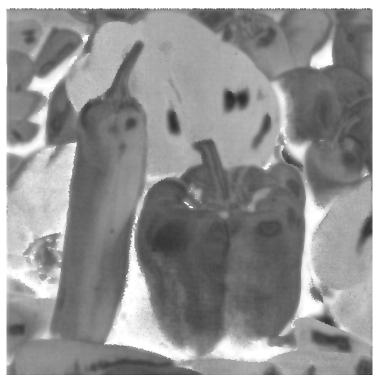
(1a) Hue of original image.

(1b) Hue of image processed with Malik-Perona's diffusion.

(1c) Hue of image processed with Tukey's diffusion.

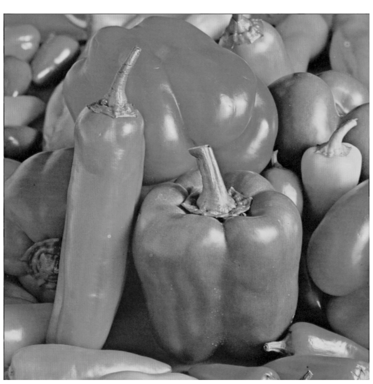


(2a) Saturation of original image.



(2b) Saturation of image processed with Malik-Perona's diffusion.

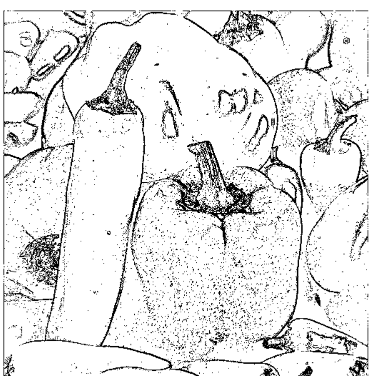
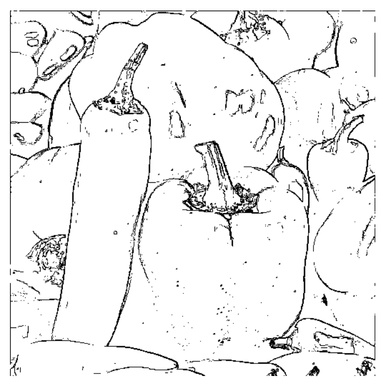
(2c) Saturation of image processed with Tukey's diffusion.

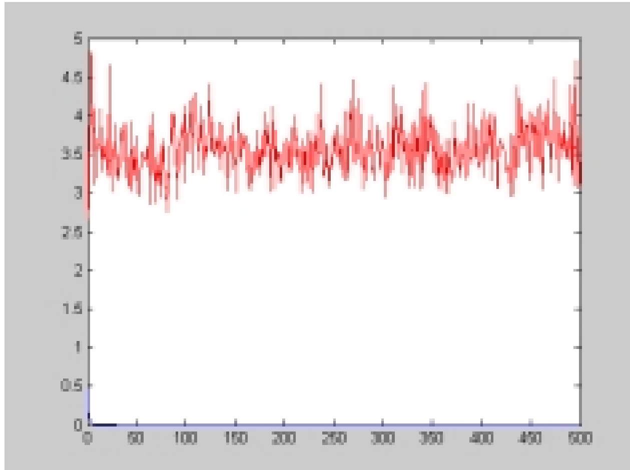


(3a) Intensity of original image.

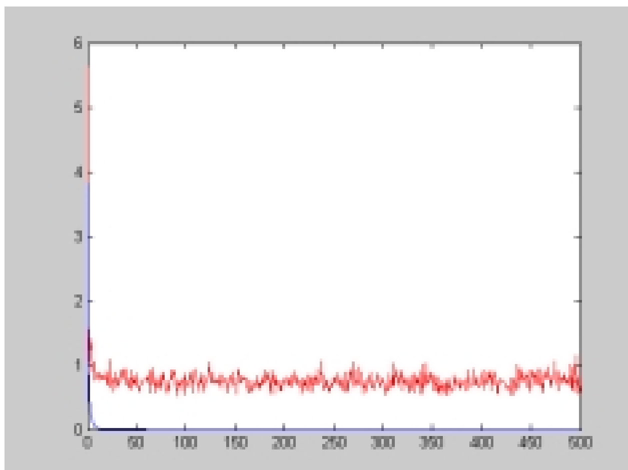
(3b) Intensity of image processed with Malik-Perona's diffusion.

(3c) Intensity of image processed with Tukey's diffusion.

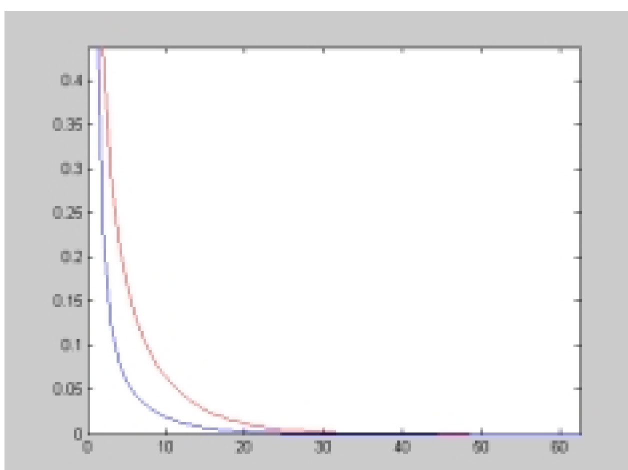
 (4a) Original color image.	 (4b) Image restored using Malik-Perona's diffusion.	 (4c) Image restored using com Tukey's diffusion.
 (5a) Edges of original color image.	 (5b) Edges of image restored by Malik-Perona's diffusion.	 (5c) Edges of image restored by Tukey's diffusion.



(6a) Average of hue diffused versus number of iterations. Tukey in blue and Malik-Perona in red.



(6b) Average of saturation diffused versus number of iterations. Tukey in blue and Malik-Perona in red.



(6c) Average of intensity diffused versus number of iterations. Tukey in blue and Malik-Perona in red.