Gradient Histogram-Based Anisotropic Diffusion

Hae Yong Kim
Universidade de São Paulo, Escola Politécnica, Dept. Eng. Sistemas Eletrônicos,
Av. Prof. Luciano Gualberto, tr. 3, 158, 05508-900, São Paulo, SP, Brazil, e-mail: hae@lps.usp.br.

Abstract

The well-known anisotropic diffusion (a.k.a. Perona-Malik equation, nonlinear diffusion, or diffusion partial differential equation – PDE) is widely used in image segmentation, filtering and edge detection. The behavior of the diffusion depends highly on the appropriate choice of the gradient thresholding scale parameter K. However, it seems that no clear relationship between the parameter K and the output image has ever been established, and hence the choice of K is a guesswork. This paper proposes the Histogram Gradient-Based Anisotropic Diffusion (GHAD). In GHAD, the user specifies the desired number ν of edge-elements (edgels) of the filtered image. Let us define that the frontier between two neighboring pixels (p, q) is an edgel if $|I(p) - I(q)| > \tau$, where $I(p)$ is the image intensity at p and τ is a constant. From the specified ν, an appropriate parameter K is automatically computed in every diffusion iteration, so that the final filtered image has almost exactly ν edgels. Using this approach, the diffusion converges to a nontrivial piecewise constant image, whenever a feasible parameter ν is specified.

1. Introduction

Linear scale space is a theory introduced by Witkin [1] and used to process an image in multiple resolutions. In this theory, Gaussian low-pass filters process the original fine-scale image, generating simplified coarse-scale images. Unfortunately, coarse-scale images generated by Gaussian filters present blurry edges that do not spatially match the original edges.

In order to keep important edges sharp and spatially fixed, while filtering noise and small edges, Perona and Malik has introduced the nonlinear scale-space [2]. This theory uses the anisotropic diffusion to simplify the original image. Recently, the relations between the anisotropic diffusion and the robust statistics have been established, leading to the robust anisotropic diffusion (RAD) that preserves the boundaries sharper than previous techniques [3].

The behavior of the anisotropic diffusion depends on two parameters: the artificial time parameter t and the gradient thresholding parameter K. The appropriate selection of both parameters is still subjects of ongoing researches and no definitive solution seems to be available [4, 5]. The aim of this paper is to contribute to the elucidation of this problem.

Perona-Malik's anisotropic diffusion converges to an image with constant graylevel when $t \to \infty$. On the other hand, RAD usually converges to a piecewise smooth image after a sufficient number of iteration steps, and consequently the image filtered by RAD depends practically only on K, provided that the diffusion is iterated a suitable number of times. Thus, for RAD, the filtered image practically depends only on the choice of K. Though, it seems that no clear relationship between the parameter K and the output image has ever been established, and hence the choice of K is a guesswork.

Some papers have proposed to automatically choose K [3, 4]. We think that such an automated strategy can be applied only to some specific application, because the amount of desired filtering is a user's choice and no automated system can predict the user's mind.

This paper proposes the Histogram Gradient-Based Anisotropic Diffusion (GHAD). GHAD can only be conceived as a spatio-temporally discretized anisotropic diffusion process. There is no version of GHAD for spatially or temporally continuous images.

In GHAD, the user specifies the desired number ν of edge-elements (edgels) of the filtered image. Let us define that the frontier between two neighboring pixels (p, q) is an edgel if $|I(p) - I(q)| > \tau$, where $I(p)$ is the image intensity at p and τ is a non-negative constant (for quantized images, τ is usually zero). The quantity ν is roughly equivalent to the sum of all edge lengths in the filtered image (distance unit in pixels). From the specified ν, an appropriate parameter K is automatically computed in every diffusion iteration, so that the final filtered image has ν edgels. Note that in each iteration, a different K must be estimated and used. Using this approach, the diffusion converges to a nontrivial piecewise constant image, whenever a feasible parameter ν

is specified.

2. Anisotropic Diffusion

Perona and Malik defined their anisotropic diffusion as [2]:

$$\frac{\partial I(x,y,t)}{\partial t} = \operatorname{div}\left[g\left(\|\nabla I(x,y,t)\|\right)\nabla I(x,y,t)\right] \quad (1)$$

using the original image $I(x,y,0) : \mathbb{R}^2 \rightarrow \mathbb{R}^+$ as the initial condition, where t is an artificial time parameter and g is an "edge-stopping" function. They suggested using one of the two edge-stopping functions below (all edge-stopping functions $g_i(x)$ presented in this paper have been dilated and scaled so that $g_i(0) = 1$ and their "influence functions" $\psi_i(x) = xg_i(x)$ have local maxima at $x = K$):

$$g_1(x) = \left[1 + \frac{x^2}{K^2}\right]^{-1}, \quad g_2(x) = \exp\left[-\frac{x^2}{2K^2}\right]. \quad (2)$$

The right choice of the edge-stopping function g can greatly affect the extent to which discontinuities are preserved. Indeed, using the function g_1, the diffusion process converges to an image with constant graylevel, where all edges are suppressed. The function g_2 preserves the edges better than g_1. However, g_2 also ends up in an image with constant graylevel.

Black et al. [3] used the robust estimation theory to choose a better edge-stopping function, called Tukey's biweight:

$$g_3(x) = \begin{cases} \left[1 - \frac{x^2}{5K^2}\right]^2, & \frac{x^2}{5} \leq K^2 \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

The diffusion that uses this edge-stopping function is called robust anisotropic diffusion (RAD) and this is the edge-stopping function adopted in this paper.

Perona and Malik [2] discretized spatio-temporally their anisotropic diffusion equation (1) as:

$$I(s,t+1) = I(s,t) + \frac{\lambda}{|\eta_s|}\sum_{p\in\eta_s} g(|\nabla I_{s,p}(t)|)\nabla I_{s,p}(t) \quad (4)$$

where $I(s,t)$ is a discretely sampled image, s denotes the pixel position in a discrete 2-D grid, $t \geq 0$ now denotes discrete time steps, the constant λ determines the rate of diffusion (usually, $\lambda = 1$), and η_s represents the set of spatial neighbors of pixel s. For 1-D signals, usually two neighbors are considered: *left* and *right*, except at signal boundaries where only one neighbor must be considered. For 2-D images, usually 4-neighborhood is used: *north*, *south*, *west* and *east*, except at the image boundaries. Perona and Malik approximated the image gradient magnitude in a particular direction at iteration t as:

$$\nabla I_{s,p}(t) = I(p,t) - I(s,t), p \in \eta_s. \quad (5)$$

Black et al. [3] have noticed that the value K_m where the influence function $\psi_i(K_m) = K_m g_i(K_m)$ is maximal determines the threshold between homogeneous regions and edges. If $\|\nabla I_{s,p}(t)\| > K_m$, the frontier between pixels s and p is considered by the diffusion as an edgel to be preserved. And if $\|\nabla I(x,y)\| < K_m$, pixels s and p are considered to belong to the same homogeneous region and the difference between them (possibly due to noises) is gradually suppressed by the diffusion.

3. Automated Selection of K

We list below some of criteria described in the literature to choose an appropriate parameter K:

1. Set K by hand at some fixed value [2].

2. Use the "noise estimator" described by Canny [6]. The histogram of the absolute values of the gradient throughout the image is computed. Then K is set equal to the 90% value of its integral at every iteration [2].

3. Use tools from robust statistics to automatically estimate the "robust scale" σ_e of an image I [3]: $K = \sigma_e = 1.4826\,\text{median}_I[|\,\|\nabla I\| - \text{median}_I(\|\nabla I\|)\,|]$. It seems that K is evaluated once at the beginning of the diffusion process and kept fixed, although it would be easy to evaluate a different K at every iteration.

4. Estimate the noise at every iteration using the difference between the average intensities of images filtered by morphological opening and closing [4]: $K = \text{average}(I \circ e) - \text{average}(I \bullet e)$, where e is a structuring element.

5. Use the p-norm of the image to estimate the noise [4]: $K = (\sigma\|I\|_p)/m$, where σ is a constant proportional to the image average intensity and m is the number of image pixels.

All above criteria try to find a parameter K to appropriately separate edges from noises. However, we argue that the classification of a pair of neighboring pixels in "edge" or "noise" depends not only on the image I to be filtered, but also on the desired scale of the filtered image. If the user wants to obtain a coarse-scale image with only the key edges, a large K should be specified. And if the user wants to obtain a fine-scale image with all the detailed edges, a small K should be chosen. Evidently, no automated process can guess the user's mind.

Voci et al. [4] noticed that anisotropic diffusions with a fixed value of K (as in criteria 1 and 3) had shortcomings. Indeed, any discretized anisotropic diffusion based on equation 4 with fixed K is unable to suppress noise, whenever the gradient of noise is larger than the gradient of edges to be preserved. They suggested decreasing K with the progress of the diffusion process (as in criteria 2, 4 and 5). Anisotropic diffusions with adaptive selection of K can suppress even noise with magnitude larger than edges. This fact will be illustrated in subsection 5.1.

4. Gradient Histogram-Based Diffusion

4.1. The Proposed Technique

We propose a new approach, named Gradient Histogram-based Anisotropic Diffusion (GHAD). GHAD establishes the connection between the parameter K and the number ν of edgels of the filtered image. Indeed, GHAD is based on the criterion 2, where K was arbitrarly set equal to the 90% value of the integral of the gradient histogram. Although this criterion was proposed in the original Perona and Malik's paper [2], it seems that many of its consequences remain unexplored.

In GHAD, the user specifies the desired number ν of edgels in the final filtered signal/image. Alternatively, the user can specify the proportion $\varpi = \nu/n$, where n is the number of possible edgels of the image.

In each diffusion iteration, GHAD computes the absolute gradient values throughout the image. These values are sorted in increasing order, generating the ordered sequence $\vec{v} = (v_0, ..., v_{n-1})$.

In order to make the output image to have ν edgels, we set $K = v_{n-1-\nu}$ in every iteration. With this setting, the diffusion considers ν frontiers between neighboring pixels as edgels to be preserved, and $n - \nu$ frontiers as homogeneous regions where the noise must be suppressed.

The sequence $\vec{v}$ does not need to be completely sorted. It is enough to know the value of the element $v_{n-1-\nu}$ of the increasing sequence. This value can be efficiently computed in average time $O(n)$ using the algorithm that computes the k-th smallest element of a sequence, derived from the quicksort. The description of this algorithm can be found in many introductory books on algorithms, for example, in [7].

Usually, the anisotropic diffusion uses 4-neighborhood for 2-D images and 6-neighborhood for 3-D volumes. Some care must be taken in 2-D/3-D GHAD implementation, to assure that each frontier between neighboring pixels is computed once and only once in $\vec{v}$. For example, a 3×3 image has 12 possible edgels, and each one of 12 gradient magnitudes must appear once and only once in $\vec{v}$.

The image filtered by GHAD has almost exactly the specified number ν of edgels, after a sufficient number of diffusion process iterations. This property established an explicit connection between the specified parameter and the output image. It seems that no previous techniques have this sort of property.

A remark on our implementation: We always normalize the original grayscale signal/image from integer intensity range $[0, ..., 255]$ to floating-point range $[0, 1]$ before performing the diffusion. The diffusion is processed using floating-point operations and variables, in order to minimize rounding errors. The obtained filtered floating-point signal/image is converted back to the integer range and saved.

5. Experimental Results

5.1. Restoration of Piecewise Constant Images

As we said before, any discretized anisotropic diffusion based on equation 4 with constant K is unable to suppress noise, whenever the gradient of noise is larger than the gradient of edges to be preserved. GHAD and the former methods with adaptive selection of K can suppress even noise with magnitude larger than edges. This subsection illustrates this property. Note that, although GHAD and the former adaptive methods can suppress noise with magnitude larger than edges, only GHAD has the flexibility to specify the desired scale (that is, the desired number of edgels) of the filtered image.

Figure 1 depicts the restoration of a piecewise constant image corrupted by Gaussian noise, using GHAD and RAD with fixed K. The original image 1(a) has background graylevel 128, with circles with intensities 89 and 166. This image was converted to floating-point range $[0, 1]$, and the zero-mean Gaussian noise with standard deviation 0.04 was added, resulting in figure 1(b).

In this particular case, the best GHAD's parameter can be obtained. The original image 1(a) has $\varpi = 0.54\%$ of edgels and consequently this is the best restoration parameter. Filtering the noisy image 1(b) by GHAD with $\varpi = 0.54\%$ (number of iterations $t = 1000$), the image 1(c) was obtained (PSNR = 46.1 dB).

Filtering the noisy image 1(b) by RAD with K fixed at 0.04 ($t = 1000$), the image 1(e) was obtained (PSNR = 39.2 dB). RAD could neither completely eliminate noise nor preserve edges sharp. RAD with another K will not do a better restoration because if a larger K is chosen (e.g., $K = 0.05$, figure 1(f)), the edges will become blurrer; and if a smaller K is chosen (e.g., $K = 0.03$, figure 1(d)), more noise will remain unfiltered. Note that the poor filtering behavior was expected.

Figure 2 depicts the restoration of a piecewise constant image highly corrupted by Gaussian noise ($\sigma = 0.08$, figure 2(a)).

Filtering the noisy image 2(a) by GHAD with $\varpi = 0.54\%$ (number of iterations $t = 1000$), the image 2(b) was obtained (PSNR = 40.1 dB).

Filtering figure 2(a) by RAD with K fixed at 0.04 ($t = 1000$), the image 2(d) was obtained (PSNR = 24.7 dB). Again, RAD with another K will not do a better restoration because if a larger K is chosen (e.g., $K = 0.05$, figure 2(e)), the edges will become blurrer; and if a smaller K is chosen (e.g., $K = 0.03$, figure 2(c)), more noise will remain unfiltered.

5.2. Filtering Natural Images with GHAD

Figure 3 depicts "Lenna" image 3(a) filtered by RAD and GHAD. RAD's parameters $K = 0.022$ and GHAD's parameter $\varpi = 5\%$ were chosen so that the final filtered images 3(c) and 3(e) have roughly the same number of edgels. Although figure 3(c) generated by RAD has more edgels than figure 3(e) generated by GHAD, image 3(c) consists of one large area (with tiny details) while image 3(e) consists of medium-sized regions.

RAD eliminates many important edges (red circles in figure 3(b)) when iterated until convergence (figure 3(c), $t = 30000$). This happens because the grayscale intensity leaks continously through smooth edges (blue circles in figure 3(b)). The anisotropic diffusion with fixed K cannot deal appropriately with slowly varying edges. A smooth edge will be either eliminated or preserved unaltered.

GHAD, on the contrary, preserves important edges sharp (red circles in figure 3(d)) even when iterated until convergence (figure 3(e), $t = 40000$). GHAD converts, if necessary, smooth edges (blue circles in figure 3(d)) in sharp edges in order to clearly delimit constant regions. Interestingly, the resulting image 3(e) has 4.9946% of edgels.

Unfortunately, the present implementation of GHAD shows preference for creating new horizontal and vertical edges over new diagonal edges (or edges in other angles). This is probably due to using 4-neighborhood spatial discretization (equation 4). We are researching spatial discretizations that do not have preferences for any particular direction.

Figure 4 shows the images obtained by RAD using different values of K, when iterated until convergence. Note that there is no value of K that makes RAD to generate images similar to those generated by GHAD.

Figure 5 shows the images obtained by GHAD using different values of ϖ, when iterated until convergence. GHAD creates artificial edges in order to not mix two regions delimited with slowly varying edges. RAD does not have this property. Also note that the images generated by GHAD has number of edgels almost exactly equal to the specified parameter ϖ.

6. Conclusions and Future Works

This paper has introduced a new variety of anisotropic diffusion named GHAD (Gradient Histogram-based Anisotropic Diffusion). In this technique, the user specified the desired number ν of edge-elements of the filtered image. From the specified ν, GHAD computes an adequate scale parameter K of the anisotropic diffusion in each iteration step. When iterated until convergence, GHAD ends up producing a piecewise constant image with the number of edge-elements almost exactly equal to ν. We have shown that GHAD outperforms the conventional robust anisotropic diffusion (RAD) in the restoration of a piecewise constant image corrupted by Gaussian noise. We have also shown that the conventional RAD cannot appropriately filter smooth edges, while GHAD converts these edges into sharp ones. In this conversion, the current implementation of GHAD shows preference for creating new horizontal/vertical edges over diagonal ones. We are researching spatial discretizations that do not have preferences for any particular direction.

7. Acknowledgements

This work was supported in part by FAPESP under grant 2003/13752-9 and by CNPq under grant 305065/2003-3.

References

[1] A. P. Witkin, "Scale-Space Filtering," in *Proc. 8th Int. Joint Conf. Art. Intelligence*, vol. 2, pp. 1019–1022, 1983.

[2] P. Perona and J. Malik, "Scale-Space and Edge Detection Using Anisotropic Diffusion," *IEEE. Trans. Patt. Anal. and Machine Intell.*, vol. 12, no. 7, pp 629–639, 1990.

[3] M. J. Black, G. Sapiro, D. H. Marimont and D. Hegger, "Robust Anisotropic Diffusion," *IEEE Trans. Image Processing*, vol. 7, no. 3, pp. 421–432, Mar. 1998.

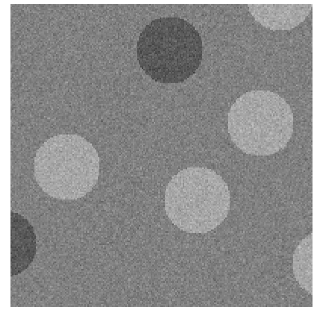
[4] F. Voci, S. Eiho, N. Sugimoto and H Sekiguchi, "Estimating the Gradient Threshold in the Perona-Malik Equation," *IEEE Signal Processing Magazine*, pp. 39–46, May 2004.

[5] V. Solo, "A Fast Automatic Stopping Criterion for Anisotropic Diffusion," in *Proc. ICASSP*, vol. 2, pp. 1661–1664, 2002.

[6] J. Canny, "A Computational Approach to Edge Detection," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-8, pp. 679–698, 1986.

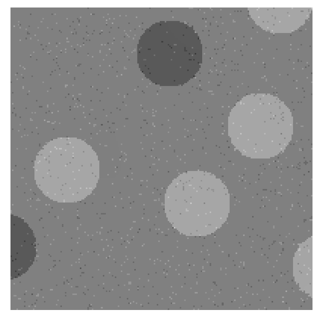
[7] R. Sedgewick, *Algorithms in C++*, Addison-Wesley, 1992.

(a) Original image.

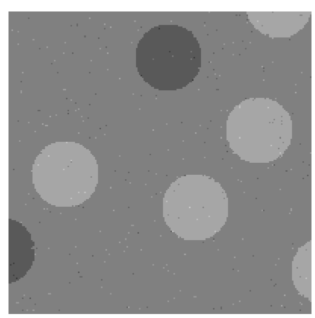


(b) Corrupted by Gaussian noise $\sigma = 0.04$, PSNR = 27.9dB.

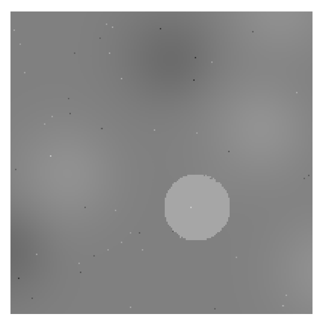
(c) Restored by GHAD, PSNR = 46.1 dB.



(d) Restored by RAD, $K = 0.03$, PSNR = 34.9 dB.

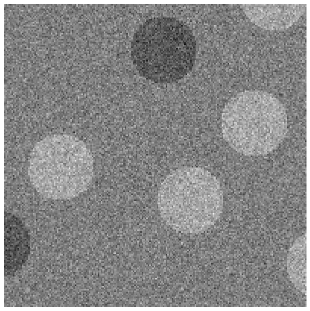


(e) Restored by RAD, $K = 0.04$, PSNR = 39.2 dB.

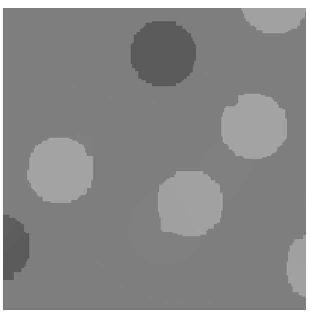


(f) Restored by RAD, $K = 0.05$, PSNR = 30.7 dB.

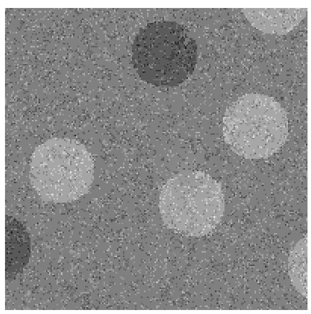
Figure 1. Restoration of a piecewise constant image corrupted by Gaussian noise with standard deviation 0.04.



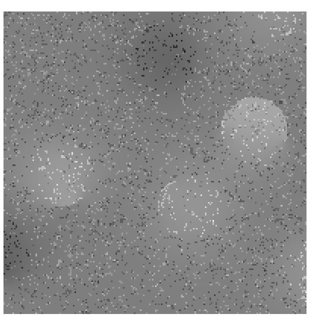
(a) Corrupted by Gaussian noise $\sigma = 0.08$, PSNR = 21.9dB.



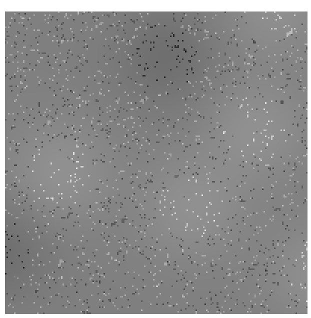
(b) Restored by GHAD, PSNR = 40.1dB.



(c) Restored by RAD, $K = 0.03$, PSNR = 23.8dB.



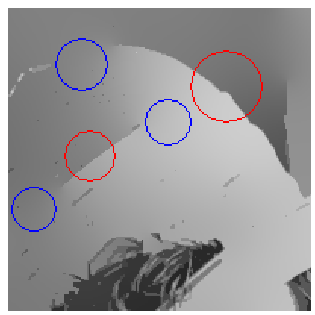
(d) Restored by RAD, $K = 0.04$, PSNR = 24.7dB.



(e) Restored by RAD, $K = 0.05$, PSNR = 25.4dB.

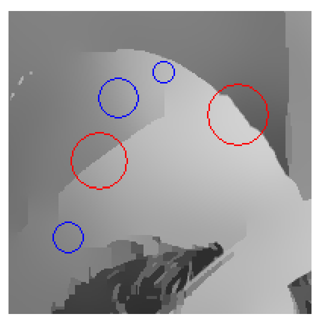
Figure 2. Restoration of a piecewise constant image corrupted by Gaussian noise with standard deviation 0.08.

(a) Original "Lenna" image.



(b) Filtered by RAD, $t = 1000$.

(c) Filtered by RAD, $t \to \infty$. Actual number of edgels = 6.29%.



(d) Filtered by GHAD, $t = 500$.

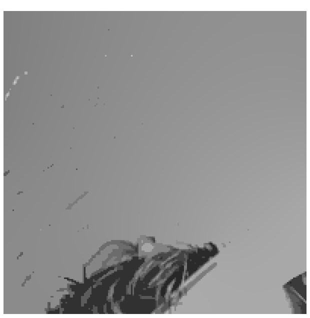
(e) Filtered by GHAD, $t \to \infty$. Actual number of edgels = 4.9946%.

Figure 3. Image "Lenna" processed by RAD ($K = 0.022$) and GHAD ($\varpi = 5\%$).

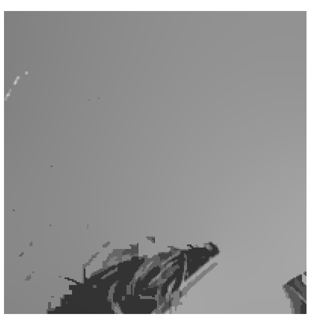
(a) $K = 0.005$

(b) $K = 0.010$

(c) $K = 0.015$



(d) $K = 0.020$



(e) $K = 0.025$

Figure 4. Image "Lenna" processed by RAD with different values K and $t \to \infty$.

(a) $\varpi = 20\%$. Actual number of edgels = 19.8179%.

(b) $\varpi = 15\%$. Actual number of edgels = 14.9515%.

(c) $\varpi = 10\%$. Actual number of edgels = 9.9967%.

(d) $\varpi = 2\%$. Actual number of edgels = 1.9978%.

(e) $\varpi = 1\%$. Actual number of edgels = 0.9991%.

Figure 5. Image "Lenna" processed by GHAD with different values ϖ and $t \to \infty$.