EE381K-14: Multidimensional Digital Signal Processing, Spring 2003
Instructor: Prof. Brian L. Evans

Recursive Implementation of Anisotropic Filtering
(Final Report)

Zeyun Yu

Department of Computer Science, University of Texas at Austin

Abstract

Gaussian filters are widely used for image smoothing but it is well known that this type of filters blur the image features (e.g., edges). Anisotropic filtering draws more and more attention for its ability of feature-preserving smoothing. However, most existing anisotropic filters are quite time-consuming. In this report I will discuss the combination of anisotropic filtering and recursive implementation for image smoothing such that the anisotropic smoothing could be performed in a fast way.

1. Introduction

Additive noise is commonly seen in many types of images (e.g., biomedical images and remote sensing images). Gaussian low-pass filtering is known to be an efficient and simple way for image smoothing. However, Gaussian filtering blurs image edges while smoothing image noise. The reason is that Gaussian filters are isotropic in the sense that all surrounding pixels affect the center one in a similar fashion regardless their intensity variations. Hence, edges and noise are treated in the same way, which yields noise reduction as well as edge blurring.

To remedy the problem of traditional Gaussian filtering, anisotropic filtering has drawn more and more attention. Several categories of anisotropic filtering have been proposed. *Bilateral Filtering* [7, 12, 13] is a straightforward extension of Gaussian filtering by simply multiplying an additional term in the weighting function. A PDE-based technique, known as *anisotropic heat diffusion,* has also been studied [4, 8] although it was shown that it is in some sense equivalent to the bilateral filtering [11]. Another popular technique for anisotropic filtering is by *wavelet transformation* [18]. The basic idea is to identify and zero out wavelet coefficients of a signal that likely correspond to image noise. By carefully designing the filter, we can smooth image noise while maintaining the sharpness of the edges in an image [19]. Finally, the development of nonlinear median-based filters in recent years has also resulted in promising results. One of those filters is called *mean-median* (*MEM*) filter [20, 21, 22]. This filter, different from the traditional median filter, can preserve fine details of an image while smoothing the noise.

Although various techniques have been proposed for anisotropic filtering in order to preserve image features, they are, however, generally much more time-consuming compared to the simple and efficient implementation of Gaussian filters. Some authors have studied fast implementation of bilateral filtering using FFT technique [12]. Their method, however, is just an approximation of bilateral filtering. In this report I discuss another way to implement anisotropic filtering. The intuition behind my method comes from the implementation of Gaussian filtering using recursive scheme [5, 6], which only requires a small constant number of MACs (multiplications and accumulations) regardless the size of the neighborhood being considered. However, the recursive scheme was originally designed for isotropic Gaussian filtering and little work has been done on combining the recursive scheme and the anisotropic filtering.

This report is organized as follows. Section 2 reviews several recursive techniques that are closely related to the proposed approach. Section 3 gives a detailed description of the proposed method and experimental results and comparisons will be given in Section 4. Finally Section 5 concludes the report.

2. Prior Work

The recursive implementation of several types of filters had been discussed in the literature. In [1, 2, 3], Deriche studied the recursive implementation of filters with exponential weighting functions. However, as pointed out in [5], all the recursive implementations seen in [1, 2, 3] are based on non-Gaussian filtering such that they do not have the impressive properties that Gaussian filters have (e.g., isotropic property). In

[5], the authors proposed a recursive implementation of the Gaussian filter. Their approach is based on a rational approximation of the Gaussian function given by:

$$g(t) = \frac{1}{\sqrt{2\pi}} e^{-t^2/2} = \frac{1}{a_0 + a_1 t^2 + a_4 t^4 + a_6 t^6} + \varepsilon(t), \tag{1}$$

where $a_0 = 2.490895$, $a_2 = 1.466003$, $a_4 = -0.024393$ and $a_6 = 0.178257$. The error $\varepsilon(t)$ is proved to be limited to $|\varepsilon(t)| < 2.7 \times 10^{-3}$. From (1), one can derive the following recursive implementation of Gaussian filtering:

$$\begin{cases} forward: & w_n = B \cdot x_n + \dfrac{b_1 w_{n-1} + b_2 w_{n-2} + b_3 w_{n-3}}{b_0} \\ backward & y_n = B \cdot w_n + \dfrac{b_1 y_{n+1} + b_2 y_{n+2} + b_3 y_{n+3}}{b_0} \end{cases}, \tag{2}$$

where $B = 1 - (b_1 + b_2 + b_3)/b_0$ and b_0, b_1, b_2, b_3 are constants derived from the standard deviation σ of the Gaussian filter. It was claimed in [5] that the above recursive implementation of the Gaussian filter only requires 6 MACs per output pixel.

As we saw in Section 1, Gaussian filtering is isotropic such that it does not preserve the sharpness of image features while smoothing noise. The main goal of this report is recursive implementation of anisotropic filtering. Fortunately there have already been several papers dealing with this issue. The first paper is the one by Alvarez et al. [9, 10], who discussed the recursive realization of the nonlinear version of the exponential filters seen in [1, 2, 3]. The idea is that, instead of using constant parameter α, we can consider a varying parameter α_n, which depends on the norm of the derivative of the input signal.

$$\alpha_n = h(|\frac{\partial x}{\partial t}(t_n)|), \tag{3}$$

where $h(.)$ is a non-decreasing function with $h(0) = \varepsilon > 0$. For simplicity, the authors in [9, 10] chose $h(.)$ as a linear function: $h(s) = \varepsilon + M \cdot s$, where M is a positive constant.

In [15, 16], the authors discussed the recursive implementation of the exact Gaussian filter. The 2D Gaussian function they considered could be anisotropic in the sense that the principal axis of the Gaussian function could be in any orientation and the associated deviations σ_u and σ_v could have any aspect ratio. However, they assumed the Gaussian function to be uniformly applied in everywhere of the image. Therefore, their method can only be used to smooth or detect the lines with a specific orientation. This assumption is certainly invalid for image smoothing with arbitrarily oriented features.

3. Proposed Approach

In this section I will discuss a method to selectively smooth image noise. Before we get into the details, let us consider a simple example: a one-dimensional signal with some noise and a sharp edge (as shown in Fig. 1(a)). Clearly an isotropic filter would give a blurred result as shown in Fig. 1(b). To maintain the sharp edge of this signal, we need to make output signal look like the one shown in Fig.1 (c). If we compare the outputs in Fig.1 (b) and Fig.1 (c), we can see that the anisotropic filtering makes the output closer to the original signal; that is, the output goes up on the left side of the edge and goes down on the right side of the edge. In the following we will use this heuristic information to design our filter. In addition to this information, we will need to compute some local statistics for each pixel, namely, the local minimum, local maximum and local average.

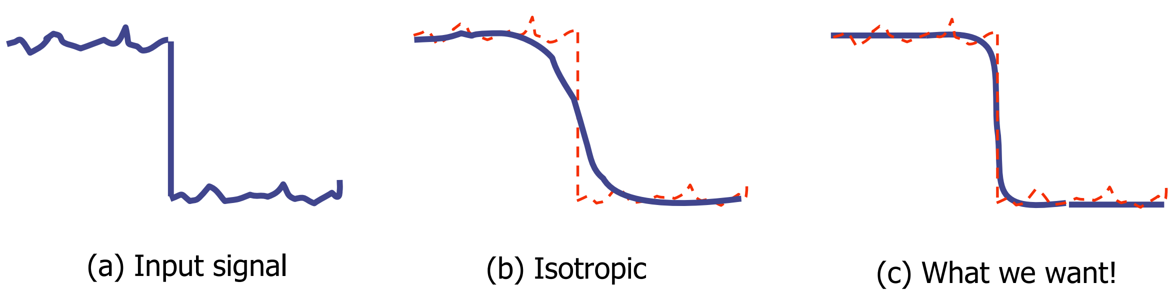


Figure 1 A simple example to illustrate the goal of our anisotropic filtering

3.1. Computing the Local Statistics

The local min/max/avg of a pixel can be simply defined as the minimal, maximal and averaging intensities within a local window with a fixed size. By this definition we can easily compute the local min/max/avg by searching all the pixels within the local window. This method is straightforward to implement but it turns out to have two problems. First, it takes a lot of time to search for the local min/max or compute the local average for each pixel. Secondly, the computed min/max maps by this simple method are not continuously defined over the image domain, which means that the resulting min/max maps may contain lots of block-like artifacts.

In the proposed method, the local statistics are computed in a recursive way. The local average for each pixel is computed in the same way as seen in (2). However the propagation rules for the local min/max are different. Note that the recursive rule in (2) is equivalent to the following rule.

$$\begin{cases} forward: & w_n = x_n + \frac{b_1}{b_0}(w_{n-1} - x_n) + \frac{b_2}{b_0}(w_{n-2} - x_n) + \frac{b_3}{b_0}(w_{n-3} - x_n) \\ backward & y_n = w_n + \frac{b_1}{b_0}(y_{n+1} - w_n) + \frac{b_2}{b_0}(y_{n+2} - w_n) + \frac{b_3}{b_0}(y_{n+3} - w_n) \end{cases} \tag{4}$$

To compute the *local minimum*, we add several additional "*sign*" terms to (4) as follows:

$$\begin{cases} forward: & w_n = x_n + \frac{b_1}{b_0}(w_{n-1} - x_n) \cdot sign(x_n - w_{n-1}) + \frac{b_2}{b_0}(w_{n-2} - x_n) \cdot sign(x_n - w_{n-2}) \\ & \quad + \frac{b_3}{b_0}(w_{n-3} - x_n) \cdot sign(x_n - w_{n-3}) \\ backward & p_n = w_n + \frac{b_1}{b_0}(p_{n+1} - w_n) \cdot sign(w_n - p_{n+1}) + \frac{b_2}{b_0}(p_{n+2} - w_n) \cdot sign(w_n - p_{n+2}) \\ & \quad + \frac{b_3}{b_0}(p_{n+3} - w_n) \cdot sign(w_n - p_{n+3}) \end{cases} \tag{5}$$

Here the function $sign(x)$ is defined as 1 if $x > 0$ and 0 otherwise. Similarly we can compute the *local maximum* using the following recursive rule.

$$\begin{cases} forward: & w_n = x_n + \frac{b_1}{b_0}(w_{n-1} - x_n) \cdot sign(w_{n-1} - x_n) + \frac{b_2}{b_0}(w_{n-2} - x_n) \cdot sign(w_{n-2} - x_n) \\ & \quad + \frac{b_3}{b_0}(w_{n-3} - x_n) \cdot sign(w_{n-3} - x_n) \\ backward & q_n = w_n + \frac{b_1}{b_0}(q_{n+1} - w_n) \cdot sign(q_{n+1} - w_n) + \frac{b_2}{b_0}(q_{n+2} - w_n) \cdot sign(q_{n+2} - w_n) \\ & \quad + \frac{b_3}{b_0}(q_{n+3} - w_n) \cdot sign(q_{n+3} - w_n) \end{cases} \tag{6}$$

3.2. Anisotropic Smoothing

The local statistics computed above can be used to determine the output for each pixel. The following figure shows the local max/min/avg maps of the input signal seen in Fig. 1(a).

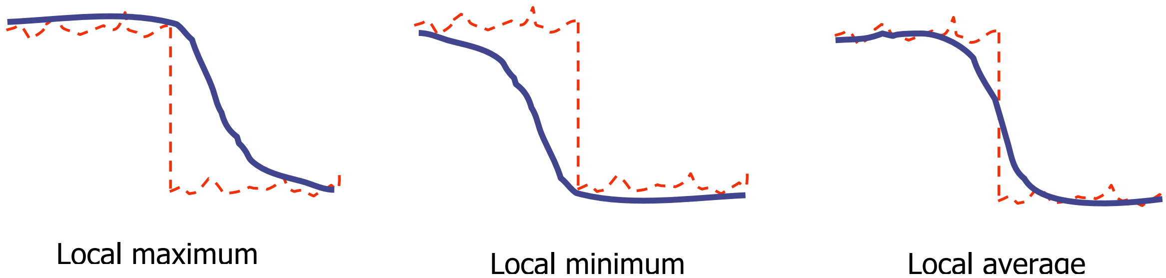


Figure 2 The local statistics of the signal shown in Fig. 1(a)

With the local statistics, we determine the output for each pixel with the following rule:

$$\begin{cases} If \quad in(x) \geq avg(x), \quad then \quad out(x) = l\max(x) \\ If \quad in(x) < avg(x), \quad then \quad out(x) = l\min(x) \end{cases}, \tag{7}$$

where $in(x)$ and $out(x)$ are the input and output signals, respectively. $lmin(x)$, $lmax(x)$ and $avg(x)$ are the local minimum, local maximum and local average at location x, respectively.

3.3. Extension to Higher Dimensions

The extension of the above method to higher dimensions is quite straightforward. We just need to perform the above one-dimensional scheme along each dimension as shown below. We had used a similar idea of the above method for image contrast enhancement [17].

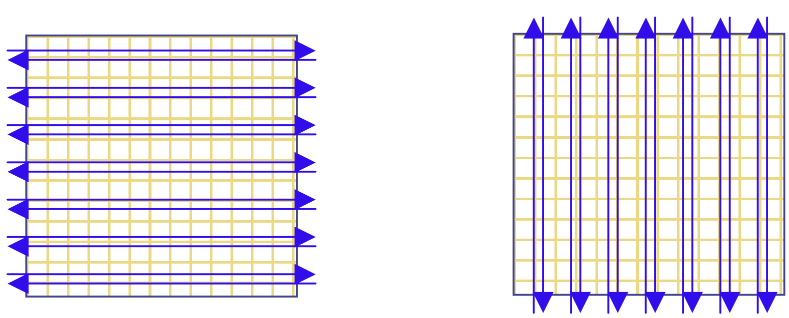
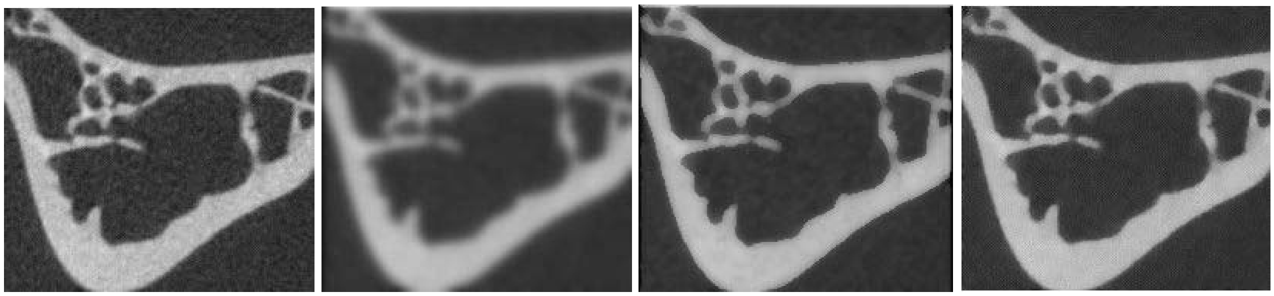


Figure 3 Illustration of the extension of our method to two-dimensional images.

4. Experimental Results and Comparisons

In this section we show several examples of 2D images and the results generated by different methods. The first example is a medical image as shown in Fig. 4(a). The rest pictures in Fig. 4 show the results by different methods. Pictures in Fig. 5 show another example with some blocks and additive noise.



(a) Original image (b) Isotropic filtering (c) Proposed method (d) Perona-Malik method [4]

Figure 4 Example I: a medical image and the comparison between several methods.

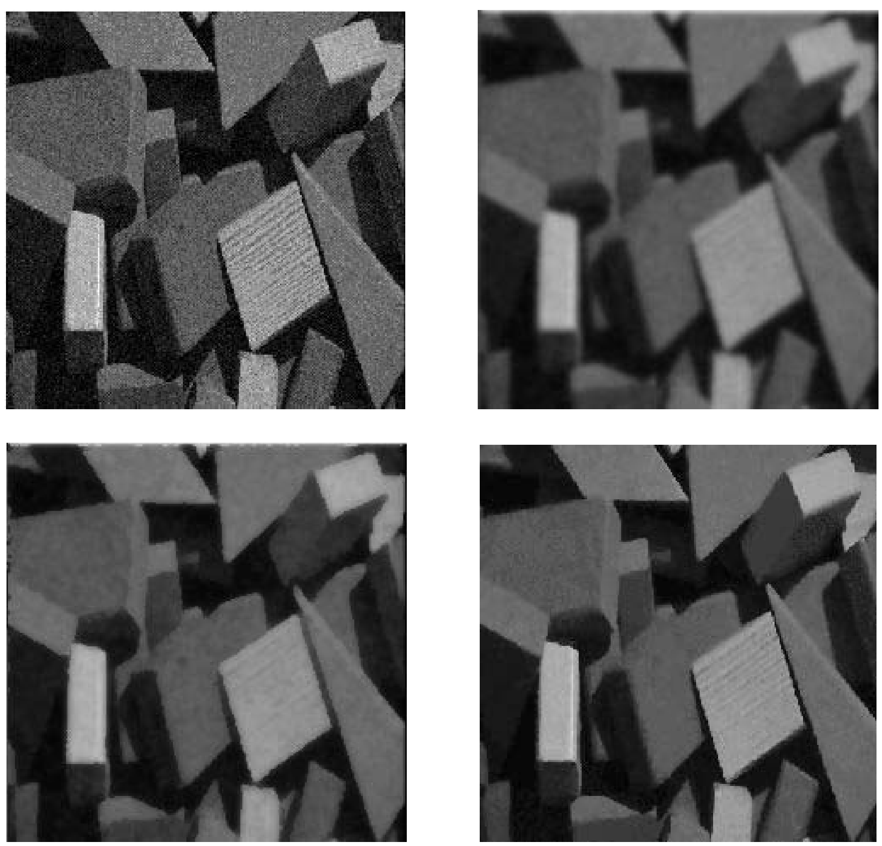


Figure 5 Example II: a block image and the comparison between several methods.

Top-left: original image; top-right: isotropic filtering;

bottom-left: proposed method; bottom-right: Perona-Malik method [4].

In the following figure, we illustrate the comparison of the computational time for three methods. We can see that our method is slower than the isotropic filtering but faster than the Perona-Malik method.

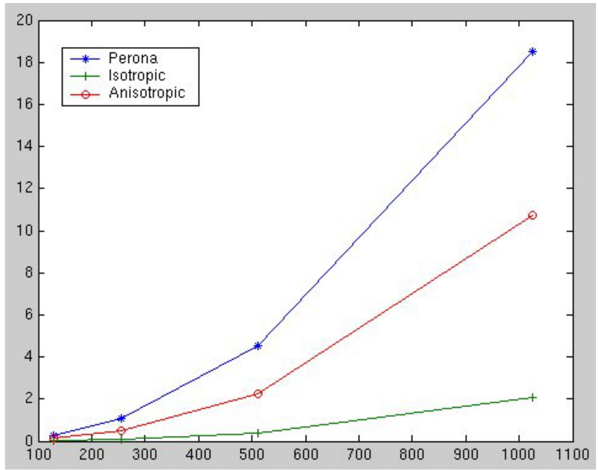


Figure 6 Comparison of the computational time for three methods

5. Conclusion

In this report I describe a recursive method for selectively smoothing images. Our method is better than isotropic filtering for its ability of preserving the sharpness of image edges. Experiments also show that our method is faster than Perona-Malik method and the results by both methods are quite similar.

References

[1] R. Deriche, "Optimal edge detection using recursive filtering", *Proceeding of 1st International Conference on Computer Vision*, London, June 8-12, 1987.

[2] R. Deriche, "Using Canny's criteria to derive a recursively implemented optimal edge detector", *International Journal of Computer Vision*, vol. 1, no. 2, pp. 167-187, May 1987.

[3] R. Deriche, "Fast algorithms for low-level vision", *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 12, no. 1, pp 78-87, 1990.

[4] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 12, no. 7, pp.629-639, 1990.

[5] I.T. Young and L.J. Vliet, “Recursive implementation of the Gaussian filter”, *Signal Processing*, vol. 44, pp. 139-151, 1995.

[6] L.J. Vliet, I.T. Young and P.W. Verbeek, “Recursive Gaussian derivative filters”, *Proceedings of the 14th International Conference on Pattern Recognition*, vol. 1, pp. 509-514, Brisbane, Australia, 16-20 August 1998.

[7] C. Tomasi and R. Manduchi, “Bilateral filtering for gray and color images”, In *Proc. IEEE Int. Conf. on Computer Vision*, pp. 836-846, 1998.

[8] J. Weickert, *Anisotropic Diffusion In Image Processing*, ECMI Series, Teubner, Stuttgart, ISBN 3-519-02606-6, 1998.

[9] L. Alvarez, R. Deriche and F. Santana, "Recursivity and PDE's in Image Processing", Universidad de Las Palmas de G.C. Technical Report (available online at: http://serdis.dis.ulpgc.es/~lalvarez/research/AlDeSa.ps), October 1999.

[10] L. Alvarez, R. Deriche and F. Santana, “Recursivity and PDE's in image processing”, *Proc. International Conference on Pattern Recognition*, vol. 1, pp. 242-248, Barcelona, Spain, September, 2000.

[11] D. Barash, “A fundamental relationship between bilateral filtering, adaptive smoothing and the nonlinear diffusion equation”, *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 24, no. 6, pp. 844-847, June 2002.

[12] F. Durand and J. Dorsey, “Fast bilateral filtering for the display of high-dynamic-range images”, In *Proc. ACM Conference on Computer Graphics (SIGGRAPH)*, pp. 257-266, 2002.

[13] M. Elad, “On the bilateral filter and ways to improve it”, *IEEE Transactions On Image Processing*, vol. 11, no. 10, pp. 1141-1151, October 2002.

[14] I.T. Young, L.J.van Vliet and M.van Ginkel, “Recursive Gabor filtering”, *IEEE Trans. Signal Processing*, vol. 50, No. 11, November 2002.

[15] J.M. Geusebroek, A.W.M. Smeulders, and J. van de Weijer. “Fast anisotropic Gauss filtering”. *Proc. European Conference on Computer Vision* (A. Heyden, G. Sparr, M. Nielsen, and P. Johansen, editors), volume 1, pages 99-112. Springer Verlag (LNCS 2350), Copenhagen, Denmark, 2002.

[16] J.M. Geusebroek, A.W.M. Smeulders, and J. van de Weijer. “Fast anisotropic Gauss filtering”. *IEEE Trans. Image Processing*, (accepted), 2003. (I have this paper but no more details about the publication date.)

[17] Z. Yu and C. Bajaj, “A fast and adaptive method for image contrast enhancement”, Submitted to *International Conference on Computer Vision*, Beijing, China, October 2003.

[18] D.L. Donoho and I.M. Johnson. “Ideal Spatial Adaptation via Wavelet Shrinkage.” *Biometrika*, vol. 81, pp. 425-455, 1994.

[19] Y. Xu, J. B. Weaver, D. M. Healy, Jr., and J. Lu, “Wavelet Transform Domain Filters: A Spatially Selective Noise Filtration Technique”, *IEEE Trans. Image Processing,* vol. 3, no. 6, November 1994.

[20] A. Ben Hamza, P. Luque, J. Martinez, and R. Roman, “Removing noise and preserving details with relaxed median filters”, *Journal of Mathematical Imaging and Vision*, vol. 11, no. 2, pp. 161-177, October 1999.

[21] A. Ben Hamza and Hamid Krim, “Image Denoising: A NonLinear Robust Statistical Approach”, *IEEE Transactions on Signal Processing*, vol. 49, no. 12, pp. 3045-3054, December 2001.

[22] A. Ben Hamza and Hamid Krim, “Nonlinear Image Filtering: Trade-off between Optimality and Practicality”, *Proc. IEEE International Conference on Image Processing*, Greece, October 2001.