A Quantitative Assessment of Edge Preserving Smoothing Filters for Edge Detection

Huseyin Gunduz, Cihan Topal, Cuneyt Akinlar

Department of Computer Engineering
Anadolu University
Eskisehir, TURKEY

{huseyingunduz,cihant,cakinlar}@anadolu.edu.tr

***Abstract* - Edge detection algorithms have traditionally utilized the Gaussian Linear Filter (GLF) for image smoothing. Although GLF has very good properties in removing noise and unwanted artifacts from an image, it is also known to remove many valid edges. To cope with this problem, edge preserving smoothing filters have been proposed and they have recently attracted increased attention. In this paper, we quantitatively compare three prominent edge preserving smoothing filters; namely, Bilateral Filter (BLF), Anisotropic Diffusion (AD) and Weighted Least Squares (WLS) with each other and with GLF in terms of their effects on the final detected edges using the precision/recall framework of the famous Berkeley Segmentation Dataset (BSDS 300). We conclude that edge preserving smoothing filters indeed improve the performance of the edge detectors, and of the filters compared, WLS yields the best performance with AD also outperforming the GLF.**

Index Terms— Edge detection, Canny, Edge Drawing (ED), edge preserving smoothing, bilateral filter, anisotropic diffusion, weighted least squares (WLS).

I. INTRODUCTION

Edge detection is one of the fundamental tools of image processing and computer vision. It is usually performed as the first step of a processing pipeline and is especially used in feature detection and extraction. The importance of this problem have led the researchers to develop many edge detection algorithms in the literature [1, 2, 3, 4].

The first step of any edge detection algorithm is to remove noise and reduce the amount of detail and unwanted artifacts in the image. The easiest and the most widely used filter for this purpose is the Gaussian Linear Filter (GLF) [1]. This filter smooths out every pixel of the input image using the same Gaussian function and thus removes potential edges along with the unwanted image artifacts, which inversely affects the edge detection performance.

To overcome this problem, edge preserving smoothing filters have been proposed in the literature. The main goal of these filters is to remove unwanted artifacts from the image as in GLF while preserving valid edge crossings. These filters can be analyzed in two different categories: (1) Those that work locally and compute the output value for each pixel as some sort of an average of the local neighborhood, (2) Those that formulate a global optimization problem using the image pixel intensity values and smoothing coefficients, and solve this problem to obtain the final filter output.

The most important method in the first category is the Anisotropic Diffusion (AD) proposed by Perona and Malik [5]. In this method, the filter coefficients are variable as opposed to Gaussian Linear Filer (GLF) and changes depending on the structure of the input image. Shortly stated, using an anisotropic diffusion based equation, the smoothing rate is adjusted based on the gradient value at each pixel. While smoothing rate is reduced at the edge crossings, it is increased over the continuous, smooth regions of the image. Although this method works well in practice, it does not perform well especially on noisy images. Konishi proposes making use of different statistical methods in the determination of edge crossings instead of using a gradient operator [6]. In this method, the image is modelled as a random field and the relationships between the pixels are analyzed. In addition to the original AD proposed by Perona and Malik, many different AD variants have been proposed in the literature [7, 8, 9, 10, 11].
Another important and popular edge preserving smoothing filter from the first category is the Bileteral Filter (BLF) proposed by Tomasi and Manduchi [12]. This filter makes use the spatial and range differences around the center pixel to produce a weighted average around the neighborhood. The fact that the filter kernel size is not linear creates computational problems. For this reason, several methods have been proposed to speed up the filter and make it available for applications that require fast computation [13, 14, 15].

An important method from the second category is the Weighted Least Squares (WLS) [16, 17]. The idea with this method is to formulate image smoothing as a global optimization problem and solve a system of linear equations to obtain the output image.

II. IMAGE SMOOTHING FILTERS

In this study, four popular image smoothing filters have quantitatively been compared. Here is a brief description of the filters to be compared.

A. Gaussian Linear Filter (GLF)

Given an input image *f*, and a Gaussian smoothing filter function *g*, the smoothed output image *h* is calculated at each pixel (x, y) by the following convolution equation:

$$h(x,y) = f(x,y)\,g(x,y), \tag{1}$$

Since this filter is based on the Gaussian function, the weights of the pixels closer to the center pixel would be higher than the weights of the pixels far from the center as follows:

$$G(x,y) = \frac{1}{2\pi\sigma^2}\exp\left(-\frac{x^2+y^2}{2\sigma^2}\right). \tag{2}$$

B. Bilateral Filter (BLF)

Contrary to GLF, where the weights of the filter coefficients are the same for each pixel of the input image, the bilateral filter adjusts the weights of the filter coefficients based on the intensity values of the current pixel region, thus trying to preserve boundary crossings. If the difference between the intensity of the center pixel and the intensity of the pixels in the neighborhood is high, then the weights filter coefficients are reduced. Thus after filtering, the intensity of the center pixel is prevented from big changes. In other words, in BLF, the intensity values of the pixels located on edge crossings are mainly determined by the pixels in the same class. The BLF equation is given as:

$$BLF(I)_p = \frac{1}{W_p}\sum_{q\in S} G_{\sigma_s}\left(\| p-q \|\right) G_{\sigma_r}\left(| I_p - I_q |\right) I_q, \tag{3}$$

where σ_s is the standard deviation in the spatial domain, and σ_r is the standard deviation in the spectral domain. W is used for normalization.

C. Anisotropic Diffusion (AD)

Anisotropic Diffusion (AD) works by the application of the heat diffusion equation over the image as follows [5]:

$$\left\{\begin{array}{l} \dfrac{\partial I}{\partial t} = div\left[c\left(|\nabla I|\right)\cdot\nabla I\right] \\ I(t=0) = I_0 \end{array}\right\}, \tag{4}$$

where I_0 is the input image, ∇ is the gradient operator, *div* is the divergence operator, $|\,|$ is the magnitude operator, and *c(x)* is the diffusion coefficient function and can be taken as one of the following:

$$c(x) = \frac{1}{1+(x/k)^2}, \tag{5}$$

and

$$c(x) = \exp\left[-(x/k)^2\right], \tag{6}$$

where *k* is the edge magnitude parameter.

Gradient magnitude is used to identify the edge areas or intensity discontinuities. At pixels where $|\nabla| >> k$, the value of $c(|\nabla|)$ becomes 0. At pixels where $|\nabla| << k$, the value of $c(|\nabla|)$ becomes 1.

The equation (4) can be written in discrete form as:

$$I_s^{t+\nabla t} = I_s^t + \frac{\nabla t}{|\bar{\eta}_s|}\sum_{p\in\bar{\eta}_s} c\left(\nabla I_{s,p}^t\right)\nabla I_{s,p}^t, \tag{7}$$

where I_s^t represents the discretized image, *s* is the location of the pixel, ∇t is the step size, $\bar{\eta}_s$ is the spatial neighborhood of *s*, $|\bar{\eta}_s|$ is the number of pixels in the filter window. We finally obtain:

$$\nabla I_{s,p}^t = I_p^t - I_s^t, \forall p \in \bar{\eta}_s. \tag{8}$$

D. Weighted Least Squares (WLS)

Given an input image *f,* the goal is to obtain the output image u by minimizing the energy function given in (9) using the Weighted Least Squares (WLS) method [17],

$$J(u) = \sum_p\left(\left(u_p - f_p\right)^2 + \lambda\sum_{q\in N(p)} w_{p,q}(f)\left(u_p - u_q\right)^2\right), \tag{9}$$

where *N(p)* represents pixel *p*'s neighbors and λ is a balancing factor. The weight equation $w_{p,q}$ is the similarity between pixels *p* ve *q*. When *J(u)* is set to 0, the minimized *u* is obtained by solving a system of linear equations represented by a sparse matrix:

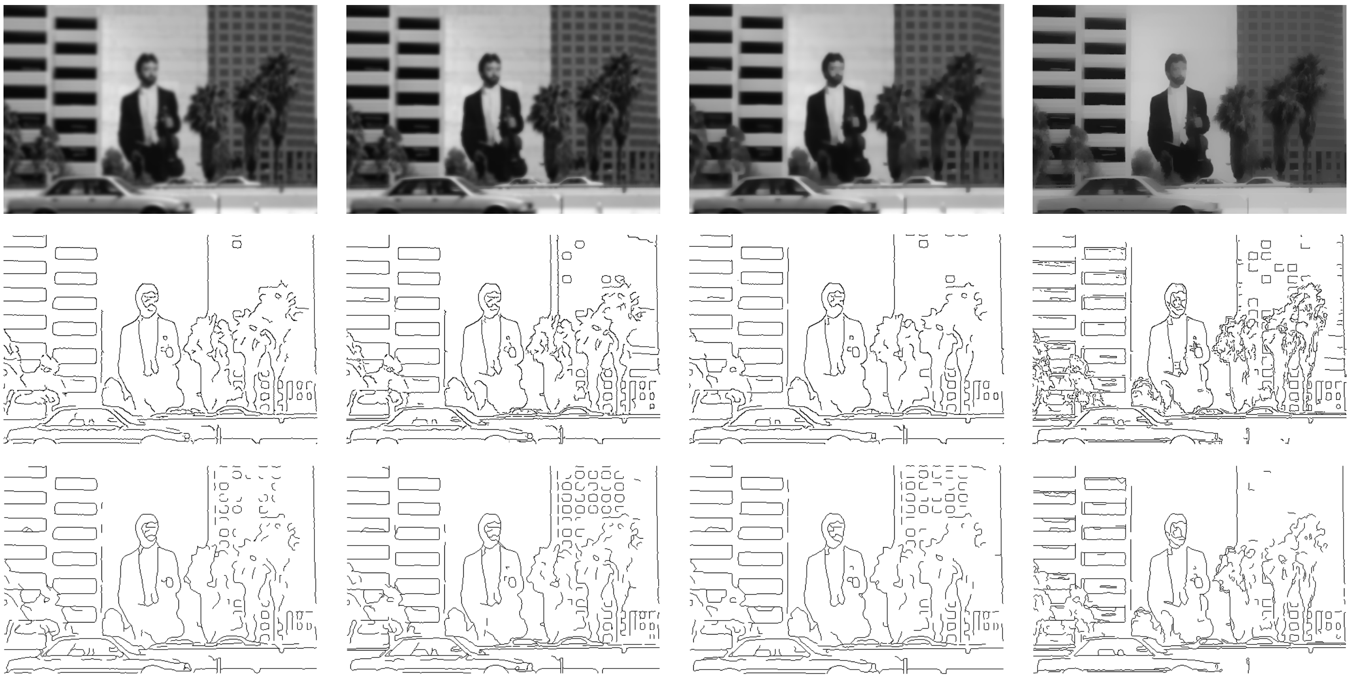
$$(I + \lambda A)u = f, \tag{10}$$

In [16], the authors has expressed the output image, *u*, as a gradient equation as follows:

$$\sum_p\left(\left(u_p - f_p\right)^2 + \lambda\left(a_{x,p}(f)\left(\frac{\partial u}{\partial x}\right)_p^2 + a_{y,p}(f)\left(\frac{\partial u}{\partial y}\right)_p^2\right)\right), \tag{11}$$

$$a_{x,p}(f) = \left(\left|\frac{\partial \ell}{\partial x}\right|^\alpha + \varepsilon\right)^{-1} a_{y,p}(f) = \left(\left|\frac{\partial \ell}{\partial y}\right|^\alpha + \varepsilon\right)^{-1}, \tag{12}$$

where ℓ is the log-luminance channel of the input image, α is the gradient sensitivity, ε is a constant to prevent division by zero at places where *f* is constant.



(a) (b) (c) (d)

Fig. 1. Image 119082 of BSDS. Top to bottom: Smoothed image, Canny edge map having the maximum F-score, ED edge map having the maximum F-score, smoothed by a) GLF b) BLF c) AD d) WLS filters.

IV. EXPERIMENTAL RESULTS

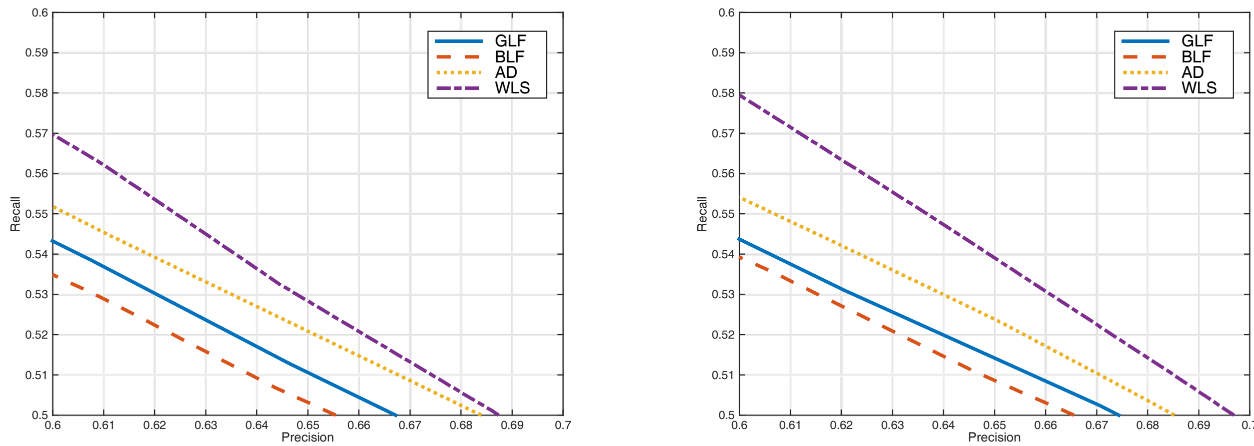
In this section we quantitatively compare three prominent edge preserving smoothing filters; namely, AD, BLF and WLS with each other and with GLF in terms of their effects on the final detected edges. To achieve this goal, we make use of the precision/recall framework of the famous Berkeley Segmentation Dataset (BSDS) [18]. This dataset contains 200 training and 100 test images and is very popular in quantitatively comparing different boundary detection methods [19, 20]. Images in the dataset are of 481×321 or 321×481 pixel resolution and each image has between 5 to 10 human marked segmentations, which are used as the ground truth data. The edge (boundary, contour) detection results from an algorithm is automatically compared with the ground truth data using the precision/recall framework and an overall F-measure score is produced by the BSDS testbed to rate the performance an algorithm. This enables objective comparison of different edge detection algorithms.

To compare the effects of different image smoothing methods, an input image is first passed through the smoothing filter. The output of the filter is then processed by two different edge detectors: The famous Canny edge detector [1] and the recently proposed real-time edge segment detector, Edge Drawing (ED) [2]. Both of these detectors take a smoothed image as input and produce a binary edge map as output, which is then used in the BSDS evaluation testbed to compute an F-score for an image. The higher the F-score, the better the edge map. The edge detection algorithms are run with the same parameters for each input image. Thus, the smoothing filter that maximizes the F-measure score can be said to be the best.

Fig. 1 shows a sample image from the BSDS dataset along with the smoothed images with different smoothing filters. The edge maps having the maximum F-score for each smoothed image obtained by Canny and ED are also presented in Fig. 1. Finally, the precision/recall/F-score values corresponding to each edge map are presented in Table 1.

Fig. 2 shows the precision/recall curves of maximum F-score using the 100 test images in the BSDS dataset for each of the smoothing filters obtained by Canny and ED respectively, and Table 2 gives a summary of the results. Clearly, WLS yields the best performance with AD also outperforming GLF, while BLF shows mixed performance and does not live up to the expectations.

Finally, Table 3 shows the running time for each filter averaged over the 100 test images in the BSDS testbed. Clearly, GLF is two orders of magnitude faster than the other filters, which explains its wide adaption especially in applications that require real-time performance.



(a) (b)

Fig. 2. a) Canny edge detection performance b) ED edge detection performance

Table 1. Detection performance for image 119082 of the BSDS dataset

	Canny			ED		
	Precision	Recall	F-score	Precision	Recall	F-score
GLF	0.688	0.766	0.725	0.675	0.743	0.708
BLF	0.662	0.828	0.735	0. 637	0. 788	0.705
AD	0.699	0.782	0.738	0.656	0.757	0.703
WLS	0.649	0.831	0.729	0.688	0.805	0.742

Table 2. Maximum F-score obtained for each smoothing filter by Canny and ED for the BSDS 300 dataset.

	Maximum Canny F-score	**Maximum ED F-score**
GLF	0.571	0.574
BLF	0.567	0.570
AD	0.578	0.580
WLS	0.585	0.590

Table 3. Average running time of each smoothing filter for the images in the BSDS 300 dataset.

	Time (ms)
GLF	1
BLF	534
AD	395
WLS	764

V. Conclusions

Edge preserving smoothing filters try to remove noise and unwanted artifacts from an image while preserving important edge/boundary crossings. In this paper, we quantitatively compared three edge preserving smoothing filters with each other and with the widely used Gaussian Linear Filter. While BLF did not live up to the expectations, both WLS and AD were observed to produce better results compared to GLF. We conclude that edge preserving filters indeed improve the performance of the edge detectors, and further research needs to be performed to improve the existing filters or to develop new ones. Also, their running times need to be improved to make them suitable for real-time computer vision applications.

REFERENCES

[1] J.F. Canny, “A Computational Approach to Edge Detection,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 8, no. 6, pp. 679-698, 1986.

[2] C. Topal and C. Akinlar, “Edge Drawing: A Combined Real-Time Edge and Segment Detector,” Journal of Visual Communication and Image Representation, vol. 23, no. 6, pp. 862-872, 2012.

[3] D. Marr, E. Hildreth, “Theory of edge detection,” Proceedings of the Royal Society of London, B207, pp. 187–217, 1980.

[4] R. M. Haralick, “Digital step edges from zero crossing of second directional derivatives,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 6, no. 1, pp. 58–68, 1984.

[5] P. Perona and J. Malik, “Scale-space and edge detection using anisotropic diffusion,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 12, no. 7, pp. 629–639, 1990.

[6] S. Konishi, A. L. Yuille, J. M. Coughlan and S. C. Zhu, “Statistical Edge Detection: Learning and Evaluating Edge Cues,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 25, no. 1, pp. 57-74, 2003.

[7] G. Cottet, L. Germain, “Image processing through reaction combined with nonlinear diffusion,” Mathematics of Computation, vol. 61, pp. 659–673, 1993.

[8] J. Weickert, “Anisotropic Diffusion in Image Processing,” ECMI Series, Teubner-Verlag, 1998.

[9] S. Aja-Fernández, C. Alberola-López, “On the estimation of the coefficient of variation for anisotropic diffusion speckle filtering,” IEEE Transactions on Image Processing, vol. 15, no. 9, pp. 2694–2701, 2006.

[10] J. Weickert, “Coherence-enhancing diffusion filtering,” International Journal of Computer Vision, vol. 31, pp. 111–127, 1999.

[11] Y. Yu, S. Acton, “Speckle reducing an isotropic diffusion,” IEEE Transactions on Image Processing, vol. 11, no. 11, pp. 1260–1270, 2002.

[12] C. Tomasi and R. Manduchi, “Bilateral filtering for gray and color images,” in IEEE Int. Conf. on Computer Vision, pp. 839–846, 1998.

[13] F. Durand and J. Dorsey, “Fast bilateral filtering for the display of high dynamic range images,” ACM Trans. Graph., vol. 21, no. 3, pp. 257–266, 2002.

[14] F. Porikli, “Constant time O(1) bilateral filtering,” in IEEE Conf. on Computer Vision and Pattern Recognition, 2008.

[15] S. Paris and F. Durand, “A fast approximation of the bilateral filter using a signal processing approach,” International Journal of Computer Vision, vol. 81, no. 1, pp. 24–52, 2009.

[16] D. Lischinski, Z. Farbman, M. Uyttendaele, and R. Szeliski, “Interactive local adjustment of tonal values,” ACM Transactions on Graphics, vol. 25, no. 3, pp. 646-653, 2006.

[17] D. Min, S. Choi, J. Lu, B. Ham, K. Sohn, and M. N. Do, "Fast Global Image Smoothing Based on Weighted Least Squares", IEEE Trans. Image Process., vol. 23, no. 12, pp. 5638-5653, 2014.

[18] D. Martin, C. Fowlkes, D. Tal, J. Malik, “A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics,” in Proceedings of the Eighth International Conference on Computer Vision, vol. 2, pp. 416–423, 2001.

[19] C. Lopez-Molina, M. Galar, H. Bustince, B. De. Baets, “On the impact of anisotropic diffusion on edge detection,” Pattern Recognition, vol. 47, no. 1, pp. 270-281, 2014.

[20] X. Hou, A. Yuille, C. Koch, “A meta-theory of boundary detection benchmarks,” in Proceedings of the NIPS Workshop on Human Computation for Science and Computational Sustain ability, 2012.