

## Article

# A Novel Framework for Shock Filter Using Partial Differential Equations

Chunmei Duan <sup>1,\*</sup> and Hongqian Lu <sup>2,\*</sup><sup>1</sup> School of Management Science and Engineering, Shandong Normal University, Jinan 250014, China<sup>2</sup> School of Electrical and Automation, Qilu University of Technology, Jinan 250353, China

\* Correspondence: cmduan@sdu.edu.cn (C.D.); hqlu@qlu.edu.cn (H.L.); Tel.: +86-531-8618-0509 (C.D.)

Academic Editor: Raúl Alcaraz Martínez

Received: 31 December 2016; Accepted: 22 March 2017; Published: 26 March 2017

**Abstract:** In dilation or erosion processes, a shock filter is widely used in signal enhancing or image deburring. Traditionally, sign function is employed in shock filtering for reweighting of edge-detection in images and decides whether a pixel should dilate to the local maximum or evolve to the local minimum. Some researchers replace sign function with tanh function or arctan function, trying to change the evolution tracks of the pixels when filtering is in progress. However, analysis here reveals that only function replacement does usually not work. This paper revisits first shock filters and their modifications. Then, a fuzzy shock filter is proposed after a membership function in a shock filter model is adopted to adjust the evolve rate of image pixels. The proposed filter is a parameter tuning system, which unites several formulations of shock filters into one fuzzy framework. Experimental results show that the new filter is flexible and robust and can converge fast.

**Keywords:** signal enhancing; fuzzy logic; partial differential equation (PDE); shock filter

## 1. Introduction

Image enhancement is a method for improving the quality or sharpening certain details of an image and is widely used in many fields such as fingerprint recognition [1], medical image processing [2–4], remote sense image processing [5], and underwater image processing [6]. There is an enormous amount of different approaches to performing image enhancing, which can broadly be divided into domain transform [2,5,7,8], histogram equalization [3,9], and feature-oriented filtering [10–13]. As a feature-oriented filtering approach, shock filters are easily implemented and are effective in edge enhancing in image enhancing tasks.

In the past few years, there has been a growing amount of research concerning partial differential equations (PDEs) for computer vision and image processing [8,14–19]. PDEs have come into a general framework for enhancement purposes and are developed in a coupled or independent manner. Shock filters are such a PDE-based approach for the enhancement of image signals proposed by Osher and Rudin [12], which is attractive for many other applications such as motion deblurring [20], geometry processing [13], and image restoration [21]. In contrast to Fourier- or wavelet-based methods, or linear approaches, shock filters avoid Gibbs phenomena in spatial domains.

The first shock filter was proposed by Kramer and Bruckner in 1975 [22], where the Laplacian of a pixel was calculated first and the sign of the Laplacian was employed to decide which influence zone the pixel belongs to. If the Laplacian is positive, the pixel is considered to be in a maximum influence zone and will perform a dilation process, while if the Laplacian is negative, the pixel belongs to a minimum influence zone and will perform an erosion process. The dilation and erosion processes are iterated until the borderlines between the maximum and minimum zones are sharply discontinued to an appropriate level. Kramer and Bruckner formulated their shock filters in a discrete way. In contrast

to their discrete shock filter, Osher and Rudin introduced a continuous shock filter based on PDEs [12], which most of the current shock filter formulations are based on for modification.

Shock filters are popular in many applications [23–25] because of several advantages: Firstly, they create sharp discontinuities at edges in images and flat signals in other area. Second, they do not change the total variation of an image, so they are stable. However, although the conventional shock filters are attractive in many fields, especially in image enhancing, they are noise sensitive and will enhance the noise when they enhance the signals.

The goal of this paper is to propose a novel shock filter called a fuzzy shock filter, which is an adapted version of the shock filters. It provides a general framework for the different function that estimates the flow direction of the pixels; namely, we can select different functions or different parameters to sharpen or flatten a pixel in the same framework. The proposed fuzzy shock filter is more flexible and robust.

The paper is organized as follows. In Section 2, we revisit the formulation of the conventional shock filter proposed by Osher and Rudin and then other modified formulations are provided and discussed. In Section 3, we start by analyzing the value of edge response of images and then propose our fuzzy shock filter. Section 4 shows a number of experimental results of the proposed method, and some comparisons are provided. Section 5 concludes the paper.

## 2. Shock Filter and Its Modifications

The formulation of the conventional shock filter is described as Equation (1). Considering a continuous image  $I(x, y) : \mathbb{R}^2 \rightarrow \mathbb{R}$ . Then, a sequence of sharpened images  $\{I(x, y, t) | t \geq 0\}$  evolve by the PDE:

$$\begin{cases} \frac{dI(x, y, t)}{dt} = -\text{sign}(\Delta I(x, y, t)) \|\nabla I(x, y, t)\| \\ I(x, y, 0) = I(x, y) \end{cases} \quad (1)$$

where  $\nabla I$  is the gradient of  $I$ ,  $I(x, y, 0)$  is the original image,  $\text{sign}(x)$  is the sign function, and  $\Delta I$  is Laplacian of  $I$  (as shown in Figure 1).

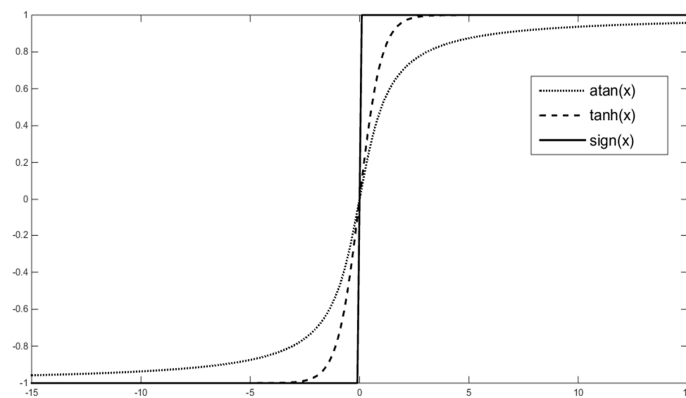


Figure 1.  $\text{sign}(x)$ ,  $\tanh(x)$ , and  $\text{atan}(x)$ .

A number of modifications of the shock filter have been proposed in recent years. The second derivative of an image has been used as an edge detector instead of a Laplacian operator, and better performances are reported in other papers [13,26,27]. For instance, more recently, it was mentioned in [13] that  $(\nabla I_t)^T H_t (\nabla I_t)$  with  $H_t$ , the Hessian of  $I_t$ , is adopted as an edge detector. To make the filters more robust, the edge detector can even take the convolution with Gaussian in [26]. Taking into account of the modification of edge detectors and note  $I(x, y, t)$  as  $I_t$ , then shock filter can be described as

$$\begin{cases} \frac{dI_t}{dt} = -\text{sign}(L(I_t)) \|\nabla I_t\| \\ I_0 = I(x, y) \end{cases} \quad (2)$$

where  $L(I_t)$  is an edge detector such as Laplacian or the second derivative of  $I_t$  along the gradient direction.

Another way to improve the shock filter is to replace the sign function with other functions such as tanh (hyperbolic tangent function) and atan (arctan function) [28], as shown in Figure 1. In Equations (1)–(3), sign, tanh, and atan functions are defined, respectively, as

$$\text{sign}(x) = \begin{cases} 1, & x > 0; \\ 0, & x = 0; \\ -1, & x < 0. \end{cases}$$

$$\tanh(x) = (e^x - e^{(-x)}) / (e^x + e^{(-x)})$$

$$\text{atan}(x) = \frac{2\arctan(x)}{\pi}.$$

In the shock filter, the sign function is employed to find the sign of the edge response value of an image pixel and different sign will be further used to decide which influence zone (maximum zone or minimum zone) the pixel belongs to. The extended shock filter is given by

$$\begin{cases} \frac{dI_t}{dt} = -F(L(I_t)) \|\nabla I_t\| \\ I_0 = I(x, y) \end{cases} \quad (3)$$

where the F function can be  $\text{sign}(x)$ ,  $\tanh(x)$ , or  $\text{atan}(x)$ .

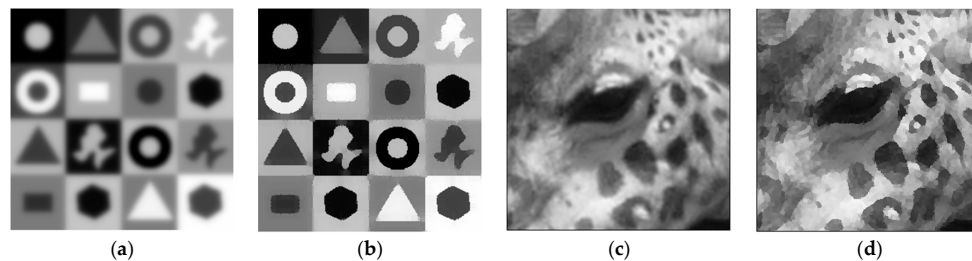
### 3. Fuzzy Shock Filter

#### 3.1. Edge Response Value Analysis

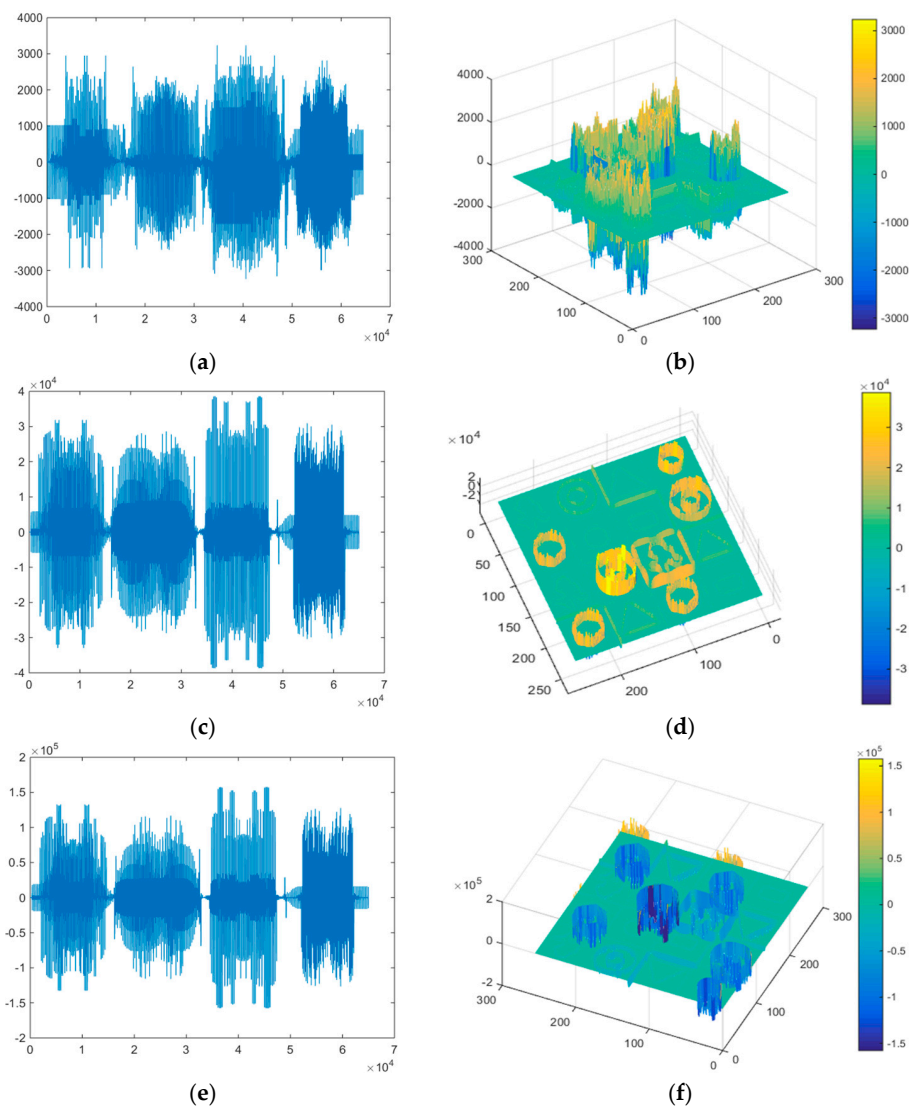
In Equation (3),  $L(I_t)$  can be any second-order edge detector. We adopt

$$L(I) = \frac{1}{\|\nabla I\|^2} (I_x^2 I_{xx} + 2I_x I_y I_{xy} + I_y^2 I_{yy}) \quad (4)$$

which corresponds to the second derivative of image  $I$  in the direction of the normal to the isophotes. Take Figure 2a (Shape) as the test image for instance: We iterated Equation (3) several times. Figure 3 shows the changes in  $L(I)$  values at different iteration times when  $F(s) = \tanh(s)$ . Situations are similar when  $F(s) = \text{atan}(s)$ . Figure 3 shows that, with the increment in iteration times, the values of  $L(I)$  keep increasing either when  $F(s) = \tanh(s)$  or when  $F(s) = \text{atan}(s)$ . However, once the value of  $L(I)$  falls out of a certain scope, the values of  $\text{sign}(L(I))$ ,  $\tanh(L(I))$ , and  $\text{atan}(L(I))$  are very similar according to the function curves in Figure 1. It is not difficult to find that most of the values of  $L(I)$  are out of the span of  $[-10, 10]$  in Figure 3, which are on the edge area of the image especially observed from the mesh figure in Figure 3. Therefore, we believe that simply replacing the sign function with  $\tanh(x)$  or  $\text{atan}(x)$  cannot improve the performance of the conventional shock filter theoretically.

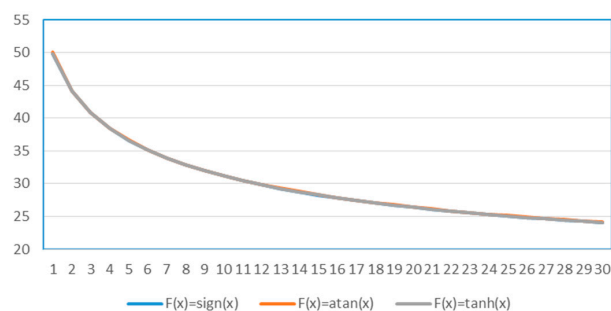


**Figure 2.** Input data (Shape and Giraffe) and filtered data obtained using our fuzzy shock filter. (a,c) The initial images; (b,d) The filtered images using the proposed filter with  $\sigma = 5$  and  $c = 0.6$ .



**Figure 3.** Statistics of  $L(v)$  value when  $F(x) = \tanh(x)$ . (a) After 1 iteration; (c) after 5 iterations; (e) after 10 iterations; (b,d,f) Corresponding mesh data.

In the experiments, we tested Model (3) using  $F(s) = \tanh(s)$ ,  $F(s) = \operatorname{atan}(s)$ , and  $F(s) = \operatorname{sign}(s)$ , respectively. Figure 4 shows the values of the peak signal-to-noise ratio (PSNR) at different iteration times during the filtering. It is obvious in Figure 4 that the trends of the PSNR coincide when the  $F$  function is  $\operatorname{sign}$ ,  $\tanh$ , or  $\operatorname{atan}$ . Therefore, it is easy to deduce again that to merely replace  $\operatorname{sign}(x)$  with  $\tanh(x)$  or  $\operatorname{atan}(x)$  will not change the filtering result.



**Figure 4.** The trend of the peak signal-to-noise ratio (PSNR) at different iteration times.



### 3.2. Fuzzy Shock Filter

The F function in the shock filter is used to decide which influence zone a pixel belongs to. If a pixel belongs to an influence zone of a local maximum, the pixel will be enhanced (the value of it will be increased in proportion to its gradient); otherwise, it will be flattened (the value of it will be decreased in proportion to its gradient). Noise can also be enhanced if they are in a maximum zone during the shock filter. Thus, we introduce a fuzzy membership function gaussmf into shock filter and modify it to

$$\begin{cases} \frac{dI_t}{dt} = -\|\nabla I_t\| \text{gaussmf}(|L'(I_t)|, \sigma, c) \text{sign}(L(I_t)) \\ L'(I_t) = \text{Norm}(L(I_t)) \\ I_0 = I(x, y) \end{cases} \quad (5)$$

where  $\text{gaussmf}(x, \sigma, c) = e^{-\frac{(x-c)^2}{2\sigma^2}}$  is employed as a membership function, Norm is normalization function and  $|*|$  is the absolute value of \*. Tuning the parameter pair  $(\sigma, c)$  in Equation (5) to different values can obtain different filters. Figure 5 shows that, when  $(\sigma, c) = (5, 1)$ , the function curve of gaussmf is very similar to that of the tanh function. Namely, different  $(\sigma, c)$  results in different change track of a pixel in the image, and different filters are therefore obtained.

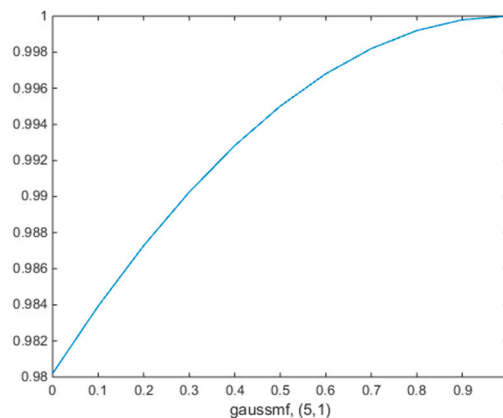


Figure 5. Curve of gaussmf with  $(\sigma, c) = (5, 1)$ .

Figure 6 shows the PSNR values at different iteration times for different fuzzy filters where  $(\sigma, c) = (0.2, 0.7), (5, 1), (0.3, 1)$ , together with a conventional shock filter. It is easy to find that the PSNR for the shock filter is coincide with that of fuzzy filter with  $(\sigma, c) = (5, 1)$ . We can also deduce from Figure 6 that certain fuzzy shock filters have better performances than conventional shock filters as far as PSNR concerned.

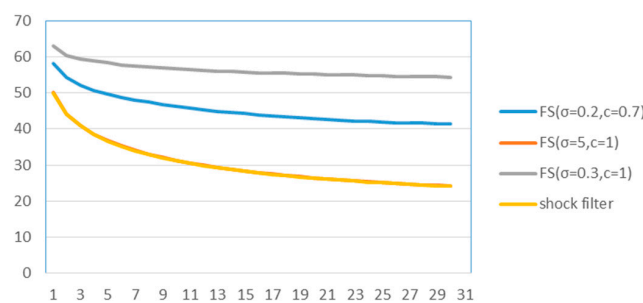
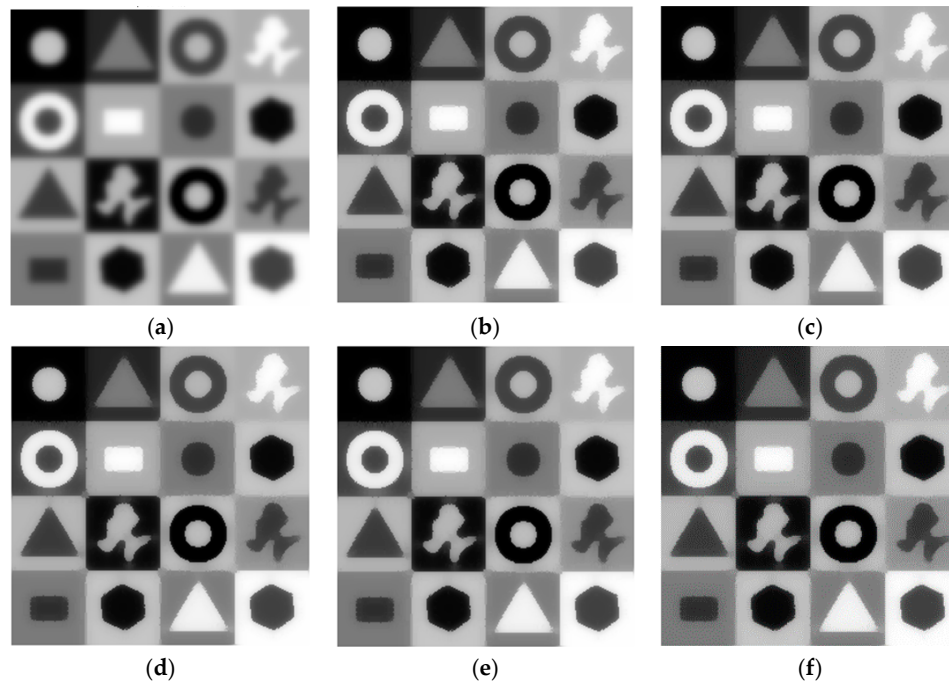


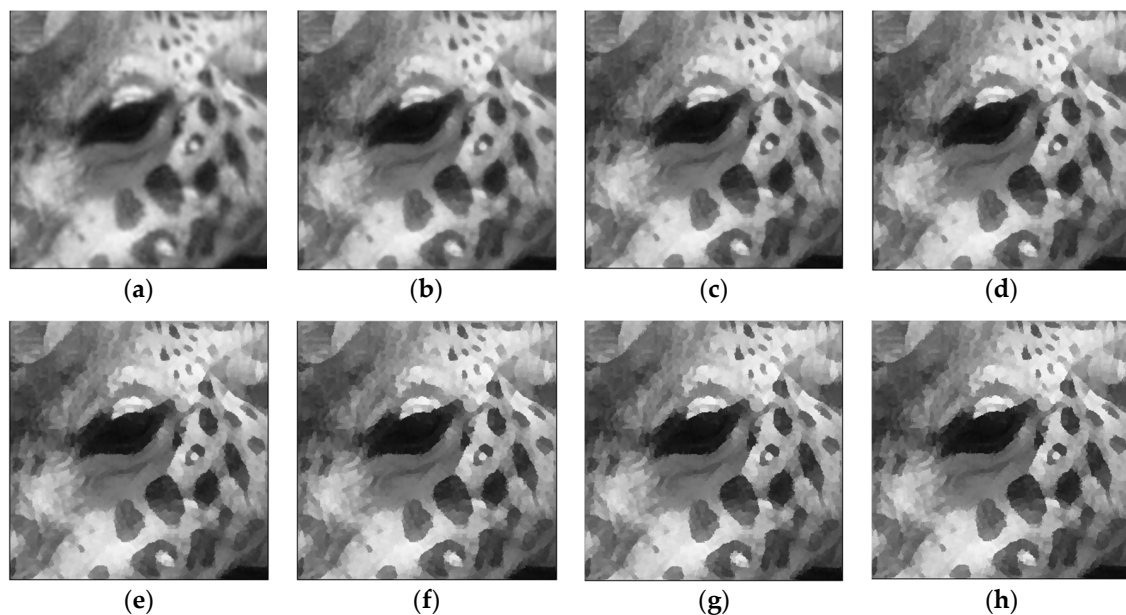
Figure 6. PSNR at different iteration times for different filters.

#### 4. Experiments

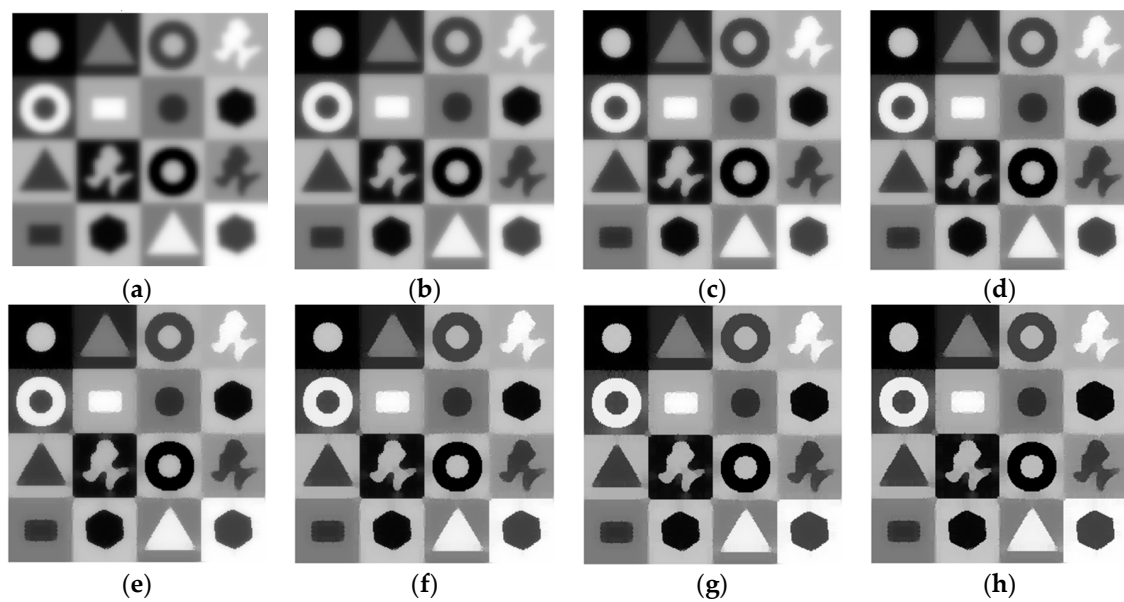
We demonstrate the effectiveness and robustness of our method using a variety of images, as shown in Figures 7–10. We start our experiment with comparing conventional shock filter and the proposed fuzzy shock filter as shown in Figure 7. Several filters are applied to the “Shape” dataset. We observe that the filtering results of different filters are similar in a visual way.



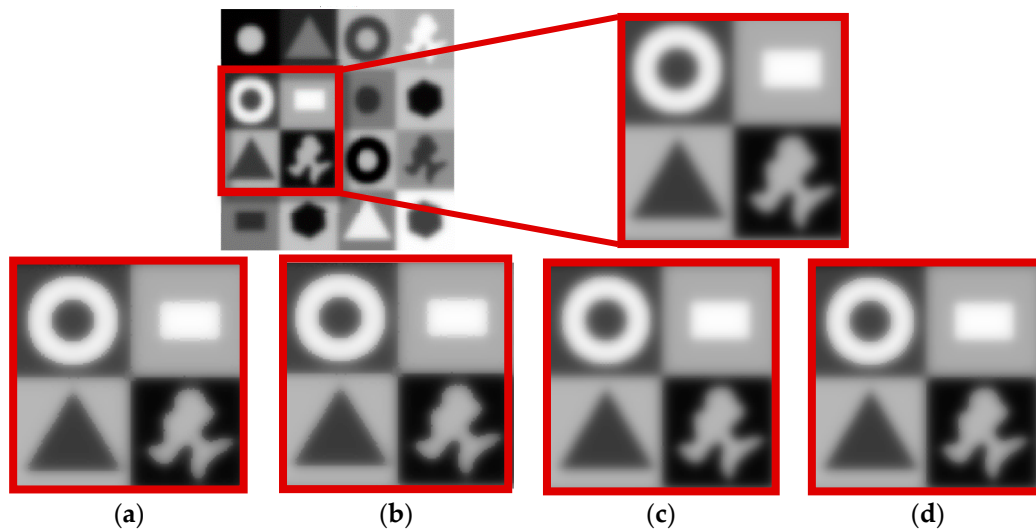
**Figure 7.** Comparison of our method to other filters. (a) The initial image; (b)  $F(x) = \text{sign}(x)$ , namely the conventional shock filter; (c)  $F(x) = \tanh(x)$ ; (d)  $F(x) = \text{atan}(x)$ ; (e) fuzzy shock filter with  $\sigma = 5$  and  $c = 0.6$ ; (f) fuzzy shock filter with  $\sigma = 0.3$  and  $c = 0.6$ .



**Figure 8.** Comparison of the Giraffe image evolution using the proposed fuzzy shock filter for different iteration times ( $t$ ). (a–d)  $t = 0, 3, 7, 10$ ; (e–h)  $t = 14, 20, 25, 30$ .



**Figure 9.** Comparison of the Shape image evolution using the proposed fuzzy shock filter for different iteration times ( $t$ ). (a–d)  $t = 0, 3, 7, 10$ ; (e–h)  $t = 14, 20, 25, 30$ .



**Figure 10.** Comparison of the proposed fuzzy shock filter with different  $(\sigma, c)$  parameters when iteration times  $t = 5$ . (a)  $\sigma = 5$  and  $c = 0.6$ ; (b)  $\sigma = 5$  and  $c = 1$ ; (c)  $\sigma = 0.2$  and  $c = 0.7$ ; (d)  $\sigma = 0.3$  and  $c = 0.6$ .

Figures 8 and 9 show the filtering results of two dataset (Giraffe and Shape) at different iteration times using the proposed fuzzy shock filter with  $(\sigma, c) = (5, 0.6)$ . It is obvious to find that our filter converges fast. Figure 10 show the comparison of the proposed fuzzy shock filter with different  $(\sigma, c)$  parameters when iteration times  $t = 5$ .

On the other hand, the peak signal-to-noise ratio (PSNR) and structural similarity index (SSIM) [29] are two widely used objective standard for image quality evaluation:

$$\text{PSNR} = 10 \times \log\left(\frac{(2^b - 1)^2}{\text{MSE}}\right)$$

where  $b$  is image bit, MSE describes the Mean Square Error between the original image and the processed image. The larger the value of PSNR is, the less distortion the processed image has. SSIM is a standard to evaluate the structure similarity between two images, which is introduced firstly by Laboratory for Image and Video Engineering at University of Texas at Austin. Given two images  $x$  and  $y$ , the SSIM of them is calculated by

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}$$

where  $\mu_s$  is the mean value of  $s$ ,  $\sigma_s^2$  is the variance of  $s$ , and  $\sigma_{xy}$  is the standard deviation of  $x$  and  $y$ .  $c_1 = (k_1L)^2$ ,  $c_2 = (k_2L)^2$  are constants for stability, where  $L$  is a dynamic range of pixel value. In our experiments,  $k_1 = 0.01$ ,  $k_2 = 0.03$ . It can be deduced that the value range of SSIM is from  $-1$  to  $1$ , and the SSIM of the two same images is  $1$ . Table 1 shows the PSNR and SSIM of the Shape data at different iteration times using different filters, where sign, atan, and tanh note three filters described in Equation (3), FS( $s, t$ ) means the proposed fuzzy shock filter with  $(\sigma, c) = (s, t)$ .

**Table 1.** Evaluation using PSNR and the structural similarity index (SSIM).

Test Images	Iteration Times	Quantity	sign	atan	tanh	FS(0.2, 0.7)	FS(5, 1)	FS(0, 1)	FS(0.3, 0.6)
Shapes	5	PSNR	26.3889	26.4506	26.4017	42.8651	26.4969	55.1689	35.3540
	5	SSIM	0.8839	0.8842	0.8832	0.9980	0.8865	0.9999	0.9870
	20	PSNR	36.6110	36.6484	36.6189	49.5914	36.7211	58.2816	43.0428
	20	SSIM	0.9863	0.9866	0.9864	0.9996	0.9866	0.9999	0.9978

## 5. Conclusions

We studied the shock filter and its modifications and have proposed a fuzzy shock filter using a fuzzy membership function. The proposed fuzzy shock filter presented the shock filter in a new way whereby several shock filter formulations can be united into one framework through the proposed membership function. The new fuzzy shock filter has provided a new parameter tuning shock filter model, which is flexible and robust, and can converge in short iteration times.

**Acknowledgments:** This research was supported by the National Nature Science Foundation of China (61502284, 61602282) and the Opening Foundation of Engineering Research Center of Digital Media Technology, Ministry of Education (2015AA0001), China.

**Author Contributions:** Chunmei Duan and Hongqian Lu conceived and designed the study. Chunmei Duan and Hongqian Lu conceived and designed the experiments. Hongqian Lu performed the experiments; Chunmei Duan analyzed the data. Chunmei Duan wrote the paper.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Hong, L.; Wan, Y.; Jain, A.K. Fingerprint image enhancement: Algorithm and performance evaluation. *IEEE Trans. Pattern Anal. Mach. Intell.* **1998**, *20*, 777–789. [[CrossRef](#)]
2. Chen, B.; Chen, Y.; Shao, Z.; Tong, T.; Luo, L. Blood vessel enhancement via multi-dictionary and sparse coding: Application to retinal vessel enhancing. *Neurocomputing* **2016**, 110–117. [[CrossRef](#)]
3. Zhang, Y.; Wu, X.; Lu, S.; Wang, S. Smart detection on abnormal breasts in digital mammography based on contrast-limited adaptive histogram equalization and chaotic adaptive real-coded biogeography-based optimization. *Simulation* **2016**, *92*, 873–885. [[CrossRef](#)]
4. Coulon, O.; Arridge, S.R. Dual echo MR image processing using multi-spectral probabilistic diffusion coupled with shock filters. In Proceedings of the MIUA'2000 British Conference on Medical Image Understanding and Analysis, London, UK, July 2000.
5. Demirel, H.; Anbarjafari, G. Discrete Wavelet Transform-Based Satellite Image Resolution Enhancement. *IEEE Trans. Geosci. Remote Sens.* **2011**, *49*, 1997–2004. [[CrossRef](#)]

6. Schettini, R.; Corchs, S. Underwater image processing: State of the art of restoration and image enhancement methods. *EURASIP J. Adv. Signal Process.* **2010**. [[CrossRef](#)]
7. Starck, J.L.; Murtagh, F.; Candes, E.J.; Donoho, D.L. Gray and color image contrast enhancement by the curvelet transform. *IEEE Trans. Image Process.* **2003**, *12*, 706–717. [[CrossRef](#)] [[PubMed](#)]
8. Kuo, C.; Yang, N.; Liu, C.; Tseng, P.; Chang, C. An effective and flexible image enhancement algorithm in compressed domain. *Multimed. Tools Appl.* **2016**, *75*, 1177–1200. [[CrossRef](#)]
9. Arici, T.; Dikbas, S.; Altunbasak, Y. A Histogram Modification Framework and Its Application for Image Contrast Enhancement. *IEEE Trans. Image Process.* **2009**, *18*, 1921–1935. [[CrossRef](#)] [[PubMed](#)]
10. Zhan, K.; Teng, J.; Shi, J.; Li, Q.; Wang, M. Feature-linking model for image enhancement. *Neural Comput.* **2016**, *28*, 1072–1100. [[CrossRef](#)] [[PubMed](#)]
11. Gilboa, G.; Sochen, N.; Zeevi, Y.Y. Image enhancement and denoising by complex diffusion processes. *IEEE Trans. Pattern Anal. Mach. Intell.* **2004**, *26*, 1020–1036. [[CrossRef](#)] [[PubMed](#)]
12. Osher, S.; Rudin, L.I. Feature-oriented image enhancement using shock filters. *SIAM J. Numer. Anal.* **1990**, *27*, 919–940. [[CrossRef](#)]
13. Prada, F.; Kazhdan, M. Unconditionally stable shock filters for image and geometry processing. *Comput. Gr. Forum* **2015**, *34*, 201–210. [[CrossRef](#)]
14. Perona, P.; Malik, J. Scale-space and edge detection using anisotropic diffusion. *IEEE Trans. Pattern Anal. Mach. Intell.* **1990**, *12*, 629–639. [[CrossRef](#)]
15. Elmoataz, A.; Lezoray, O.; Bouteux, S. Nonlocal discrete regularization on weighted graphs: A framework for image and manifold processing. *IEEE Trans. Image Process.* **2008**, *17*, 1047–1060. [[CrossRef](#)] [[PubMed](#)]
16. Ta, V.T.; Elmoataz, A.; L  zoray, O. Nonlocal pdes-based morphology on weighted graphs for image and data processing. *IEEE Trans. Image Process.* **2011**, *20*, 1504–1516. [[PubMed](#)]
17. Kornprobst, P.; Deriche, R.; Aubert, G. Image sequence analysis via partial differential equations. *J. Math. Imaging Vis.* **1999**, *11*, 5–26. [[CrossRef](#)]
18. Sochen, N.; Kimmel, R.; Bruckstein, A.M. Diffusions and confusions in signal and image processing. *J. Math. Imaging Vis.* **2001**, *14*, 195–209. [[CrossRef](#)]
19. Aubert, G.; Kornprobst, P. *Mathematical Problems in Image Processing: Partial Differential Equations and the Calculus of Variations*; Springer: Berlin/Heidelberg, Germany, 2006; Volume 147.
20. Kim, T.H.; Lee, K.M. Segmentation-free dynamic scene deblurring. In Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 2766–2773.
21. Prasath, V.S.; Vorotnikov, D. On a system of adaptive coupled PDEs for image restoration. *J. Math. Imaging Vis.* **2014**, *48*, 35–52. [[CrossRef](#)]
22. Kramer, H.P.; Bruckner, J.B. Iterations of a non-linear transformation for enhancement of digital images. *Pattern Recognit.* **1975**, *7*, 53–58. [[CrossRef](#)]
23. Sadi, A.; Elmoataz, A.; Toutain, M. Nonlocal PDE morphology: A generalized shock operator on graph. *Signal Image Video Process.* **2016**, *10*, 439–446. [[CrossRef](#)]
24. Zhou, X.; Mateos, J.; Zhou, F.; Molina, R.; Katsaggelos, A.K. Variational Dirichlet Blur Kernel Estimation. *IEEE Trans. Image Process.* **2015**, *24*, 5127–5139. [[CrossRef](#)] [[PubMed](#)]
25. Zang, Y.; Huang, H.; Zhang, L. Guided adaptive image smoothing via directional anisotropic structure measurement. *IEEE Trans. Vis. Comput. Gr.* **2015**, *21*, 1015–1027. [[CrossRef](#)] [[PubMed](#)]
26. Alvarez, L.; Mazorra, L. Signal and image restoration using shock filters and anisotropic diffusion. *SIAM J. Numer. Anal.* **1994**, *31*, 590–605. [[CrossRef](#)]
27. Weickert, J. Coherence-Enhancing Shock Filters. *Lect. Notes Comput. Sci.* **2003**. [[CrossRef](#)]
28. Gilboa, G.; Sochen, N.A.; Zeevi, Y.Y. Regularized shock filters and complex diffusion. In *Proceedings of the 7th European Conference on Computer Vision*; Springer: London, UK, 2002; pp. 399–413.
29. Wang, Z.; Bovik, A.C.; Sheikh, H.R.; Simoncelli, E.P. Image quality assessment: From error visibility to structural similarity. *IEEE Trans. Image Process.* **2004**, *13*, 600–612. [[CrossRef](#)] [[PubMed](#)]

