A Detail Preserving Vector Median Filter Based on Texture Analysis

Ying Pan, Shihui Wang

A detail preserving vector median filter based on texture analysis

Ying Pan*, Shihui Wang**

*Information Network Center, Guangxi University, 100 Daxue Road, Nanning City，Nanning 530004，China
** Information Network Center, Guangxi University, 100 Daxue Road, Nanning City，Nanning 530004，China

shwang@gxu.edu.cn

ABSTRACT. A texture detail preserving vector filter is proposed to remove impulsive noise in color images. By computing the texture complexity of the local neighborhood centered at each pixel according to the texture analysis of the color image, the proposed method classifies the local neighborhood into smooth region or texture one. For smooth regions and texture regions, different smoothing strategies are performed. For a smooth region, more smoothing is needed, whereas for a texture region the amount of smoothing should be controlled by its texture complexity. Experimental results show that the proposed method obviously outperforms the classical VMF both in noise suppression and in texture and detail preservation.

KEYWORDS: Color image, vector median filter, impulsive noise, texture analysis

1 Introduction

Images are frequently corrupted by impulse noise, which rapidly reduce image quality, and destroy image structure of texture and details. Median filter has become the most popular method to remove impulse noise in color images (Plataniotis et al., 2000), including vector median filter (VMF) (Astola et al., 1990), basic vector directional filter (BVDF) (Trahanias et al., 1993), and Distance-directional filter (DDF) (Karakos et al., 1997). They sort the vectors of pixels in the filter window using different color distances, such Euclidean distance, angle distance and mixture of Euclidean and angle distance, then select the minimum distance as the output of filter.

These vector median filters is highly effective in removing impulsive noise from color images. However, it tends to smooth image textures and details, especially when noise contamination is high. This is because VMF performs the same smoothing on all pixels without consideration of image local structures. So many improved vector median filters are proposed. Switching vector median filters (SVMF) (Smolka et al. 2005 and Malinski, et al., 2006) and weighted vector median filters (Li et al., 2006 and Jin, et al., 2008) are two typical classes of vector median filters.

This paper improves VMF based on local texture analysis, and implements a texture-presevered vector filter. The proposed method first computes the texture complexity of the local neighborhood centered at each pixel according to the texture analysis of the color image. Then, the local neighborhood is classified into smooth region or texture one. Finally, the central pixel in the local neighborhood is smoothed by traditional VMF or weighted VMF, depending on the local neighborhood is smooth region or texture one. Experimental results show that the proposed method obviously outperforms the classical VMF both in noise suppression and in texture and detail preservation.

2   Weighted vector median filter

Let the filter windows size be N, where N is odd. The pixels inside the filter window are sort from up to down and left to right. Then the pixels inside the filter window centered at pixel $\mathbf{X}_{(N+1)/2}$ are:

$$\Omega = \{\mathbf{X}_1, \mathbf{X}_2, \cdots, \mathbf{X}_{(N+1)/2}, \cdots, \mathbf{X}_N\} \tag{1}$$

For the pixel $\mathbf{X}_k$ in the window Ω, calculate the sum of the vector distances (L_2 norm) from it to other pixels in the windows firstly:

$$D_k = \sum_{i=1}^{N} \|\mathbf{X}_i - \mathbf{X}_k\|_2 \tag{2}$$

In this paper, L_2 norm (Euclidian distance) is used. Then sort D_1, D_2 ,..., D_N values in ascending order and find the smallest one among all the D_i values. Finally, the corresponding pixel $\mathbf{X}_{(1)}$ of the smallest distance $D_{(1)}$ is the output of VMF, that is the central pixel of the window is replaced with $\mathbf{X}_{(1)}$.

Weight vector median filter (WVMF) performs better in noise suppression and detail preservation than classical VMF. Different weights are assigned to pixels in the window, and pixel which are not corrupted or more similar to original pixels should be assigned higher weights.

3   Proposed detail preserving vector median filter

The proposed method performs maximum smoothing in the smooth region and less smoothing in the texture region to realize detail preservation. So the texture complexity of the pixel in a local region needs to be computed firstly. Different texture complexity methods have been developed. This paper employs edge information to measure texture complexity. The more edge pixels in a local region, the larger texture complexity of the region should be. However, due to the vector nature, extraction of edges from color images is more challenging than from grayscale images. There are different methods to detect color image edges. For simplicity, this paper uses the edges in luminance image, since human vision is highly sensitive to

luminance compared to chromaticity (Ruttiger *et al.*, 2000). We use the Canny edge operator to detect edges in luminance image (Canny, 2000). And $e(x,y)$ denotes the edge image:

$$e(x,y)=\begin{cases} 1, & (x,y)\ is\ an\ edge\ point \\ 0, & otherwise \end{cases} \tag{3}$$

Let the size of the local region for computing the texture complexity is $K \times K$. If the pixel (x,y) locate in the texture region, then there are some edges crossing the region K where (x,y) is the central pixel. Typically, there are one or two edges crossing the region. So the texture complexity is computed by

$$\rho(x,y)=\frac{\sum_{(x,y)\in K}e(x,y)}{K} \tag{4}$$

The pixel in edge texture region is smoothed by weighted VMF, and others are smoothed by traditional VMF. In other words, if the pixel located in the smooth region ($\rho(x,y)<1$) or in the texture region ($\rho(x,y)\geq1$) but is not an edge pixel, we use traditional VMF to smooth. Otherwise, for the edge pixels in the texture region, we use weighted VMF to perform. This paper uses Gaussian weighting function in the weight vector median filter:

$$w(x,y)=exp\left(-\frac{x^2+y^2}{2\sigma^2}\right) \tag{5}$$

where the central pixel of the filter window is $(0,0)$ and σ is the mean variance of Gaussian function to adjust the weight level. Given the same size of the filter window, the more value of σ, the more uniform of the Gaussian weights. Then the weighted VMF smooth the noise highly and more details of the image are smoothed. When $\sigma \rightarrow +\infty$, the WVMF degrades into classical VMF. On the contrary, the smaller value of σ is, the greater the difference among the weights. Then the weighted VMF smooth the noise lowly and more details of the image are preserved.

We can use the value of $e(x,y)$ to judge the pixel (x,y) is or not located in the edge. However, there are always offset between the actual edge pixel and the detected edge pixel in the process of edge detection, which offset is always one or two pixels. So this paper denotes the pixel (x,y) is an edge pixel if there is at least one edge pixel in its 5×5 neighbourhood region.

Then output of the proposed detail preserving vector median filter is given by

$$\mathbf{y}^{(Proposed)}(x,y)=\begin{cases} \mathbf{y}^{(WVMF1)}(x,y)\ , & \sum_{(m,n)\in\Re}e(m,n)>0\ and\ \rho(x,y)\geq1\ and\ \rho(x,y)<2 \\ \mathbf{y}^{(WVMF2)}(x,y)\ , & \sum_{(m,n)\in\Re}e(m,n)>0\ and\ \rho(x,y)\geq2 \\ \mathbf{y}^{(VMF)}(x,y)\ , & otherwise \end{cases} \tag{6}$$

where $\Re$ is the 5×5 neighbourhood region that central pixel is (x,y), and $\mathbf{y}^{Proposed}$, $\mathbf{Y}^{VMF}$, $\mathbf{y}^{(WVMF1)}$ and $\mathbf{y}^{(WVMF2)}$ are the outputs of the proposed filter,

traditional VMF, and WVMF with the mean variance of Gaussian function in Eq. [5] σ_1 and σ_2 ($\sigma_2 < \sigma_1$), respectively.

It can be seen from the above equations that in smooth regions the maximum smoothing by traditional VMF will be performed, whereas in texture regions, less smoothing introducing by WVMF, which is determined by parameter σ, is carried out. Thus, texture preservation is implemented.

4 Experimental Results

The objective evaluation measures, **PSNR** (peak signal-to-noise ratio), **MAE** (mean absolute error), and **NCD** (normalized color difference) are used to represent the quality of different filters (Plataniotis *et al.*, 2000). The PSNR and MAE are used to measure the performance of noise suppression and structural content (edges, textures, and fine details) preservation, respectively, whereas the NCD is used to quantify the color distortion.

In the proposed algorithm, four tuning parameters Ω, K, and σ_1 and σ_2 need to be fixed. They are set as follows: Ω (filter window size) is 3x3, K (neighborhood size for computation of texture complexity) is 7x7, and $\sigma_1 = 1$ and $\sigma_2 = 0.8$ (used for WVMF in the Gaussian weighting function). Note that when computation of edges, the color image is preprocessed by 3x3 VMF.

The proposed method has been evaluated by an extensive range of tests. Figure 1 shows two test color images, "Lena" and "Parrots" with size of 256×256. And the performance of the proposed method is compared with VMF, BVDF and DDF. Table 1 and Table 2 list the experimental results of VMF, BVDF, DDF and the proposed method for Lena and Parrots with noise ratio (NR) of 5%、10%、20%、30%, respectively. It is seen that the proposed method significantly outperforms VMF, BVDF, DDF in terms of all the three criteria. It do well in noise suppression (see PSNR measure), and is effectively in texture and detail preservation (see MAE measure), and preserve color hue (see NCD measure).

Figure 2 show the filter images of Parrots with 20% noise ratio which are smoothed by VMF, BVDF, DDF and the proposed method respectively. It can be seen that the proposed method is effective both in noise suppression and in texture and detail preservation.

Furthermore, to get a better look at the filter image details, Figure 3 shows the amplification of a part of the filtered image of Lena with 10% noise. Obviously, the proposed method preserves well the image details, especially the parts of the eyes and the hair.

NR Filters	5%			10%			20			30%		
	PSNR	MAE	NCD	PSNR	MAE	NCD	PSNR	MAE	NCD	PSNR	MAE	NCD
Noisy	23.50	2.51	0.0274	18.64	7.63	0.0826	17.27	10.42	0.1144	15.50	15.71	0.1733
VMF	29.47	2.99	0.0163	29.10	3.22	0.0177	28.35	3.67	0.0206	27.38	4.25	0.0249

	PSNR	MAE	NCD	PSNR	MAE	NCD	PSNR	MAE	NCD	PSNR	MAE	NCD
BVDF	27.48	4.03	0.0194	27.07	4.23	0.0205	26.53	4.57	0.0229	25.43	5.23	0.0274
DDF	29.48	2.98	0.0162	28.82	3.39	0.0183	28.10	3.85	0.0210	27.09	4.43	0.0250
Proposed	32.85	1.93	0.0113	32.01	2.16	0.0128	30.30	2.65	0.0161	28.29	3.34	0.0222

Table 1. The results of VMF、BVDF、DDF and the proposed algorithm for Parrots image with different noise ratio

NR	5%			10%			20			30%		
Filters	PSNR	MAE	NCD	PSNR	MAE	NCD	PSNR	MAE	NCD	PSNR	MAE	NCD
Noisy	23.55	2.49	0.0275	20.41	5.17	0.0581	17.39	10.27	0.1185	15.61	15.50	0.1723
VMF	32.08	2.71	0.0167	31.55	2.97	0.0185	30.47	3.51	0.0220	28.92	4.21	0.0273
BVDF	31.64	2.86	0.0164	30.86	3.21	0.0186	29.06	4.00	0.0231	26.95	5.02	0.0299
DDF	32.06	2.71	0.0165	31.52	2.97	0.0182	30.42	3.53	0.0217	28.84	4.24	0.0270
Proposed	35.19	1.74	0.0112	34.14	1.99	0.0129	31.99	2.56	0.0172	29.33	3.31	0.0239

Table 2. The results of VMF、BVDF、DDF and the proposed algorithm for Lena image with different noise ratio

(a) Lena   (b) Parrots

Fig. 1. Two test color images

(a) Noisy image(20%)   (b) Proposed   (c) VMF   (d) BVDF   (e) DDF

Fig. 2. Filtered image (Parrots with 20% noise)

(a) Original   (b) Noisy image(10%)   (b) Proposed   (c) VMF   (d) BVDF   (e) DDF
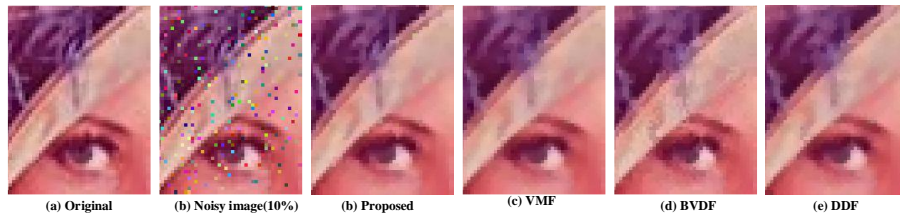
Fig. 3. The amplification of a part of Filtered image (Lena with 10% noise)

5 Conclusions

A detail preserving vector median filter for removal of impulse noise from color images is proposed in this paper. First, by analyzing the edge map of a color image, the original image pixels are classified into texture pixels and non-texture ones. Then, for texture pixels, a weighted VMF is performed, which results in less smoothing in denoising the texture pixels and thus the textures and details are effectively preserved. For non-texture pixels, the traditional VMF is carried out, which ensures that maximum smoothing is performed and thus the smoothness of the non-texture pixels are kept. By this way, the proposed texture-preserved vector filter is implemented. Experimental results indicates the superiority of the proposed method by showing significant performance gains both in noise reduction and detail preservation, compared to the traditional VMF.

6 References

1. Plataniotis K N, Venetsanopoulos A N., "*Color Image Processing and Applications*". Berlin: Springer-Verlag. 2000.
2. Astola J, Haavisto P, Neuvo Y., "*Vector median filters*", vol.78, No.4, pp. 678-689, Proc. IEEE. Proceeding of the IEEE, 1990.
3. Trahanias P E, Venetsanopoulos A N., "*Vector directional filters: a new class of multichannel image processing filters*", vol.2, No.4, pp. 528-534, IEEE Transactions on Image Processing, 1993.
4. Karakos D G, Trahanias P E., "*Generalized multichannel image filtering structures*", vol.6, No.7, pp. 1038-1045, IEEE Transactions on Image Processing, 1997.
5. Smolka B, Chydzinski A, Wojciechowski K., "*Fast Detection and Impulsive Noise Attenuation in Color Images*". vol.11, No.5-6, pp. 389-402, Real-Time Imaging, 2005.
6. Malinski L, Smolka B., "*Fast averaging peer group filter for the impulsive noise removal in color images*". vol.11, No.3, pp. 427-444, Journal of Real-Time Image Processing, 2016.
7. Li Y, Arce G R, Bacca J., "*Weighted Median Filters for Multichannel Signals*", vol.54, No.11, pp. 4271-4281, IEEE Transactions on Signal Processing, 2006.
8. Jin L H, Xiong C Q, Li D H., "*Adaptive center-weighted median filter*", vol.36, No.8, pp. 9-12, Journal Huazhong University of Science and Technology (Natural Science Edition), 2008.
9. Rüttiger L, Lee B B., "*Chromatic and luminance contributions to a hyperacuity task*", vol.40, No.7, pp. 817-832, Vision Research, 2000.
10. Canny J F., "*A computational approach to edge detection*", vol.8, No.6, pp. 679-698, IEEE Transactions on Pattern Analysis and Machine Intelligence, 1986.