A NOVEL CONTENT-ADAPTIVE VIDEO DENOISING FILTER

Tai-Wai Chan, Oscar C. Au, Tak-Song Chong, Wing-San Chau

Department of Electrical and Electronic Engineering
Hong Kong University of Science and Technology
Email: eedavid@ust.hk, eeau@ust.hk, eects@ust.hk, eecws@ust.hk

ABSTRACT

In this paper, we propose a simple non-linear content-adaptive filter that is efficient in removing noise from a video. The proposed filter is called spatiotemporal varying filter (STVF) and is able to produce optimal results in the sense that it minimizes the weighted least square error. STVF combines the advantages of conventional denoising filters that enable it to decrease the noise variance in smooth areas but at the same time retains the sharpness of edges in object boundaries. Simulation results show that STVF outperforms the conventional denoising methods like low-pass filtering, median filtering and wiener filtering.

I. INTRODUCTION

Over the past years, with the advance of signal processing and networking technologies, applications that utilize digital video have been increased dramatically. Famous examples included video conferencing, digital TV broadcasting and other multimedia services. Various digital video processing technologies have been therefore developed to tackle different associated problems. One of the techniques is video denoising, which is a process to remove noise from a digital video.

There are many ways noise could get into a digital video, typical ways are through the acquisition system and in the process of transmission over networks. Noise in a digital video is undesirable not only because it scarifies the perceptual quality but also increases the entropy of that digital video and decreases the compression efficiency of a predictive video encoder for the video. Thus, video denoising is important as it could increase the perceptual quality and at the same time decrease the entropy of a digital video.

Many denoising techniques have been developed in the literatures [1-7]. Those algorithms involved using traditional stochastic processing skills and modern digital wavelet domain image processing techniques. In [4], a filter combining Kalman and Wiener estimates is introduced, temporal and spatial redundancies are exploited by Kalman and Wiener filters respectively. In [7], methods on wavelet image denoising are discussed. These algorithms would usually depend on heavy computational power for sorting, calculating variances, expectations and doing motion compensation etc. In addition, they might also require a large amount of memory storage, for instance, storing sub-band image information. Besides these complicated methods, other simple algorithms like low-pass filtering are also available; however, the results are not satisfactory. For example, low-pass filtering would blur the object boundaries and create sawtooth effect.

In the proposed video denoising algorithm, while obtaining satisfactory results, we could also maintain low computation and low memory requirement. The remaining of this paper is organized as follow. In section II, we would introduce the proposed algorithm, and in section III, the simulation results are shown. Finally, conclusions would be drawn in section IV.

II. THE PROPOSED ALGORITHM

Noise could easily get into a digital video through the acquisition system or from the networks of transmission and is usually with random pattern. Pixel values in a digital video should be highly correlated within a small region, both in spatial and temporal domains. We would exploit this correlation in a simple and effective way.

The proposed algorithm is divided into three sections, which are called noise detection, adaptive filtering and pixel regulating as shown in Fig.1.

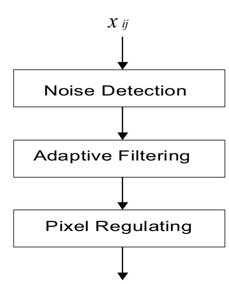

Fig.1 Overall algorithm flow

A noisy digital video is processed on a frame-by-frame and pixel-by-pixel base. Each original noisy pixel value $\boldsymbol{x_{ij}}$ would be processed one by one and a reconstructed pixel value $\boldsymbol{z_{ij}}$ would be generated to replace the original noisy pixel value in order to form the denoised video.

A. Noise Detection

The first step in the proposed algorithm is to detect whether the noisy pixel value input $\boldsymbol{x_{ij}}$ itself is corrupted by an impulsive noise or not. The denoising model shown in Fig.2 is applied for the noise detection.

	p_{ij}	

Filtered frame t-1

	$x_{i-1,j}$	
$x_{i,j-1}$	x_{ij}	$x_{i,j+1}$
	$x_{i+1,j}$	

Current frame t

Fig.2 Denoising Model Layout

The pixel value of the current pixel $\boldsymbol{x_{ij}}$ would be compared to that of the neighboring pixels $\boldsymbol{x_{i-1,j}}$, $\boldsymbol{x_{i+1,j}}$, $\boldsymbol{x_{i,j-1}}$, $\boldsymbol{x_{i,j+1}}$ and co-located pixel $\boldsymbol{p_{ij}}$ in the previous filtered frame. If all the differences are greater than a predefined threshold $\boldsymbol{T_1}$, that means the current pixel value $\boldsymbol{x_{ij}}$ is quite different from all of its neighboring pixel values, in both spatial and temporal domains, and is therefore possibly corrupted by an impulsive noise. We could then mark it as impulsive noise corrupted, otherwise we would mark it as non-impulsive noise corrupted. This classification information would be used in the following adaptive filtering and pixel regulating processes. Mathematical description is as below.

$$
\begin{aligned}
&\textit{if } \forall x \in S, |x_{ij} - x| > T_1 \\
&\textit{then} \\
&\quad \textit{mark } x_{ij} \textit{ as impulsive noise corrupted} \\
&\textit{else} \\
&\quad \textit{mark } x_{ij} \textit{ as non-implusive noise corrupted} \\
&\textit{where} \\
&\quad S = \{x_{i-1,j}, x_{i+1,j}, x_{i,j-1}, x_{i,j+1}, p_{i,j}\}
\end{aligned}
$$

B. Adaptive Filtering

After noise detection in the first step, a filtered value $\boldsymbol{y_{ij}}$ would be assigned to each noisy pixel in this step. We would mainly use a spatiotemporal varying filter (STVF) in Eq. (2) to generate the filtered value $\boldsymbol{y_{ij}}$. If the current pixel is marked as impulsive noise corrupted, then the correlation between the noisy pixel value $\boldsymbol{x_{ij}}$ and its original pixel value should be small and we would assign its corresponding filtered value as the weighted average of its neighboring pixel values $\boldsymbol{x_{i-1,j}}$, $\boldsymbol{x_{i+1,j}}$, $\boldsymbol{x_{i,j-1}}$ and $\boldsymbol{x_{i,j+1}}$ in the current frame by using Eq. (1) for further process.

$$y_{ij} = \frac{x_{i-1,j} + x_{i+1,j} + x_{i,j-1} + x_{i,j+1}}{4} \tag{1}$$

If the current pixel is marked as non-impulsive noise corrupted, then the correlation between the noisy pixel value $\boldsymbol{x_{ij}}$ and its original pixel value should be large. We would like to generate a filtered value $\boldsymbol{y_{ij}}$ that is close to the noisy pixel value. The filtered value of the current pixel would be calculated depending on its local characteristic using Eq. (2) and Eq. (3).

$$y_{ij} = \left\{ \frac{\sum_{x \in S} f(|x_{ij} - x|) * x}{\sum_{x \in S} f(|x_{ij} - x|)} \right. \tag{2}$$

where

$$S = \{p_{ij}, x_{ij}, x_{i-1,j}, x_{i+1,j}, x_{i,j-1}, x_{i,j+1}\}$$

$$f(i) = \begin{cases} 2^{(\lfloor T_1/8 \rfloor - \lfloor i/8 \rfloor)} & i < T_1 \\ 0 & \textit{else} \end{cases} \tag{3}$$

In order to approximate the original pixel value of current pixel, we assign its filtered value as a weighted average of its noisy pixel value and its neighboring pixel values. The weight of each pixel is given by Eq. (3) which depends on the pixel value difference between the current noisy pixel and its neighboring pixel. It is assumed that the current noisy pixel value is highly correlated to its original pixel value as it is not impulsive noise corrupted. And the pixel value differences between the noisy pixel and its neighboring pixels should reflect the correlations between its original pixel and its neighboring pixel values. The smaller the pixel value difference is, the larger the correlation is. Therefore, we would like the filtered value to be generated in a way that the pixel value differences are minimized. In the other words, we need to minimize the weighted least square error (WLSE) between the filtered value and the pixel values $\boldsymbol{x_{ij}}$, $\boldsymbol{x_{i-1,j}}$, $\boldsymbol{x_{i+1,j}}$, $\boldsymbol{x_{i,j-1}}$, $\boldsymbol{x_{i,j+1}}$ and $\boldsymbol{p_{ij}}$, and this constraint is formulated in Eq. (4)

The filtering result of the STVF is optimal as we could see that the optimal solution in Eq. (5) has the same form as Eq. (2). In the other words, STVF could generate the best candidate value to replace the noisy value of current pixel by exploiting the corrections of its neighboring pixel values within a small region and taking optimal weights of them.

$$Minimize\, WLSE = \sum w_i (x_i - y)^2 \tag{4}$$

$$Optimal\ solution\ y = \frac{\sum w_i x_i}{\sum w_i} \tag{5}$$

STVF in fact combines the advantages of low-pass filter and edge-preserving filter. In high-texture areas or object boundaries, STVF would maintain the sharpness of edges as it exploits the edge information by taking into account the pixel value differences between the current processing pixel and its neighboring pixels. It is because if one of the neighboring pixels belongs to other object, then the pixel value difference between the current pixel and that neighboring pixel should be large and STVF would give a small weight for that pixel. While in smooth areas, as all the pixel value differences should be similar, STVF would give similar weights to all the neighboring pixels and perform as well as low-pass filter to decrease the noise variance of smooth areas effectively.

C. Pixel Regulating

The final procedure in the proposed algorithm is to obtain the reconstructed pixel value $\boldsymbol{z_{ij}}$ from the filtered value $\boldsymbol{y_{ij}}$. If the current pixel is not marked as impulsive noise corrupted, then the original pixel value should be highly correlated to the corrupted noisy pixel value $\boldsymbol{x_{ij}}$, therefore we try to keep a small deviation between the reconstructed pixel value and the noisy pixel value using a predefined threshold $\boldsymbol{T_2}$ and Eq. (6). If the current pixel is marked as impulsive noise corrupted, then the correlation between current noisy pixel value and its original pixel value is small, we then set the reconstructed pixel value equal to the filtered pixel value, i.e. setting $\boldsymbol{z_{ij}} = \boldsymbol{y_{ij}}$.

$$z_{ij} = \begin{cases} y_{ij} & |y_{ij} - x_{ij}| \le T_2 \\ x_{ij} - T_2 & y_{ij} < x_{ij} - T_2 \\ x_{ij} + T_2 & y_{ij} > x_{ij} + T_2 \end{cases} \tag{6}$$

Lastly, the denoised video could be obtained by replacing all the noisy pixel values with the reconstructed pixel values.

III. SIMULATION RESULTS

Simulations have been done on several video sequences of size 352x288 pixels. Gaussian noise of zero mean with variances 9 and 16 are added to the video sequences, and then the noisy video sequences are denoised using various denoising algorithms. In each sequence, PSNR is calculated based on the denoised video with respect to the original one. Both the PSNR and perceptual quality of the denoised video sequences are compared. We could see from Table.1 and Table.2 that the proposed algorithm obtained the highest PSNR in all cases and is able to obtain up to 8dB gain compared to that of Wiener filter. There is gain of STVF because it could remove noise from a video but not blurring the edges. From Fig.1 to Fig.6, we also could see that the proposed algorithm results in the best perceptual quality among all the denoising algorithms; this could be easily seen by looking at the words like “EBEL” as shown in the pictures.

IV. CONCLUSION

A simple and yet effective non-linear video denoising filter called STVF is introduced. The proposed filter could obtain an optimal filtering result in the sense that it minimizes the weighted least square error. STVF is not only able to remove noise effectively but at same time retaining image details. Simulation results show that the proposed filter could outperform conventional denoising algorithms both in term of perceptual quality and PSNR, and could obtain up to 8dB gain compare to that of Wiener filter.

V. ACKNOWLEDGEMENT

This work has been supported in part by the ITF grant ITS/122/03 and the grant DAG03/04.EG10 of the Hong Kong Special Administrative Region, China.

VI. REFERENCES

[1] C. Taswell, “The what, how, and why of wavelet shrinkage denoising”, *IEEE Computational Science and Engineering*, Vol. 2 , Issue 3 , pp.12 - 19,May-June 2000

[2] D.D. Muresan and T.W. Parks, “Adaptive principal components and image denoising”, *Proc. of IEEE Int. Conf. on Image Processing(ICIP)*, Vol. 1, pp.101-104 ,14-17 Sept. 2003

[3] M.E. Zervakis and A.N. Venetsanopoulos, “Linear and nonlinear image restoration under the presence of mixed noise”, *IEEE Trans. on Circuits Syst. Video Technol.*, Vol. 38, Issue 3 , pp:258 - 272, Mar. 1991

[4] R.Dugad, N. Ahuja, “Video denoising by combining Kalman and Wiener estimates”, *Proc. of IEEE Int. Conf. on Image Processing(ICIP)*, Vol. 4, pp. 152 - 156, 24-28 Oct. 1999.

[5] S. Xiaodan and L. Fei and L. Yupin, “A region merging based algorithm for total variation denoising” *Proc. of IEEE Int. Conf. on Signal Processing*, Vol. 1 , pp.861 - 864, 26-30 Aug. 2002

[6] W.-K. Ling, P.K.S.Tam, ” Video denoising using fuzzy-connectedness principles” *Proc. of IEEE Int. Symposium on Intelligent Multimedia, Video and Speech Processing*, pp.531 - 534, 2-4 May 2001

[7] Z.-X. Lin, G.-X. Song and W. Xue, "Two improved methods on wavelet image denoising", *Proc. of IEEE Int. Conf. on Machine Learning and Cybernetics*, Vol. 5 , pp.2979 - 2983, 2-5 Nov. 2003

Fig.1 Part of enlarged 1st frame of original “stefan”

Fig.2 Part of enlarged 1st frame of noisy “stefan” with Gaussian noise with variance 16

Fig.3 Part of enlarged 1st frame of denoised “stefan” using low-pass filter

Fig.4 Part of enlarged 1st frame of denoised “stefan” using median filter

Fig.5 Part of enlarged 1st frame of denoised “stefan” using wiener filter

Fig.6 Part of enlarged 1st frame of denoised “stefan” using proposed filter

Sequence	Lowpass	Median	Wiener	Proposed
akiyo	36.0769	36.2063	40.086	41.9359
coastguard	30.6388	29.493	30.8974	38.1736
foreman	33.6656	34.1092	36.9508	40.337
mother	37.6666	37.621	39.7038	42.2166
silent	33.7325	33.6773	34.7288	39.0913
stefan	28.3676	27.2374	29.7946	38.3777

Table.1 Average PSNR(dB) of denoised sequences using different denoising algorithms for noise with variance 9

Sequence	Lowpass	Median	Wiener	Proposed
Akiyo	35.817	35.8082	39.3619	40.1332
coastguard	30.5601	29.372	30.7945	37.0555
Foreman	33.5144	33.8368	36.5756	38.9512
Mother	37.2868	37.0396	39.0944	40.3688
Silent	33.5804	33.4123	34.5176	37.9775
Stefan	28.3186	27.1697	29.6893	37.0843

Table.2 Average PSNR(dB) of denoised sequences using different denoising algorithms for noise with variance 16