A New Video Denoising Method using Texture Metric and Adaptive Structure Variance

Yiwei Zhang [1], Ge Li [*2], Xiaoqiang Guo [3], Wenmin Wang [4], Ronggang Wang [5]

School of Electronic and Computer Engineering, Peking University Shenzhen Graduate School
Lishui Road 2199, Nanshan District, Shenzhen, Guangdong Province, China 518055
[1] yuriyzhang@pku.edu.cn [2] gli@pkusz.edu.cn [4] wangwm@pkusz.edu.cn [5] rgwang@pkusz.edu.cn
Academy of Broadcasting Science, SAPPRET, China
Fuxingmen Outer Street 2, Xicheng District, Beijing, China, 100866
[3] guoxiaoqiang@abs.ac.cn

***Abstract*—In this paper, an innovated method is proposed for video denoising. The method consists of two major procedures. First, a new adaptive superpixel video texture metric is proposed. This video texture metric is calculated, which relates to different parts of a video stream. Then a new adaptive structure variance is estimated by adopting fine and coarse structures. Finally, a noise filter based on the estimated weights of different structures in a video stream is applied. By comparison, the proposed method outperforms traditional state-of-art methods, especially in block artifacts reduction.**

***Index Terms*—video denoising, preprocessing, texture metric, structure variance, superpixel**

I. INTRODUCTION

Denoising technique is a well explored topic in the field of image and video pre-processing. It is the actual foundation for a plenty of applications, such as object detection, behavior analysis, video codec and computer vision, etc. Over the past few decades, the performance of image denoising has been efficiently improved by employing a much more elaborate model of natural images. By now, many good denoising algorithms [1], [2], [3], [4], [5], [6] have been proposed, and [7] presents a comprehensive comparison of them. Even though, the original purpose of denoising is to remove unexpected noise from a corrupted image and video. However the artifacts caused by these algorithms have posted a severe effect on the quality of image and video. Thus a good denoising method should introduce as few artifacts as possible. So far BM3D [4] and BM4D [5] are two of the state-of-art methods for image and video denoising. Grouping similar 2-D blocks to a 3-D group and performing transformation are the major contribution in BM3D. The method adopts a collaborative filter to perform image reconstruction, which retains better details. Deriving from BM3D, BM4D groups similar 3-D spatiotemporal volumes to a 4-D group. Those two methods performs well in the noise reduction. However, block artifacts introduced by BM3D and BM4D are still need to be improved, as shown in Fig.1 (b). In this paper, an innovated method is proposed which can reduce noise well, and in the mean time, shows less blocking artifacts.

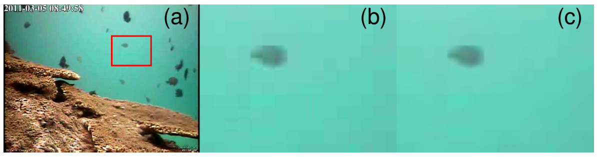

Fig. 1. (a) the first frame of Crowded3. (b) denoising results of BM4D. (c) denoising results of our method.

To achieve better denoising performance, the approach is composed of two stages: obtainment of adaptive superpixel texture metric; combination of fine and coarse structure. First, we utilize superpixel and Single Value Decomposition (SVD) to obtain a texture metric ρ for each path of each frame. By grouping all of path metric ρ, we obtain a brand new video texture metric P. And also, P is utilized to estimate proportion of fine structure and coarse structure in the video stream. Second, a new adaptive structure variance is obtained by adopting fine and coarse structure based on P , which can be fit well for different scenarios. Finally, the reconstructed video stream is obtained by applying a noise filter based on the estimated weights of different structures.

Contribution: two major procedures are proposed in this paper. First, an innovated adaptive video texture metric based on superpixel is introduced, which is related to different parts of a video stream. Second, a new adaptive structure variance is obtained by utilizing fine and coarse structures to perform filtering. By comparison, our proposed method outperforms traditional state-of-art methods, especially in block artifact reduction, as shown in Fig.1 (c).

The remainder of this paper is organized as follows. Section II gives the details of our method. Section III gives the experimental results of our approach and comparisons with two state-of-art algorithms. Finally, the paper is concluded in section IV.

This project was supported by Shenzhen Peacock Plan (20130408-183003656), Science and Technology Planning Project of Guangdong Province, China (No. 2014B090910001) and China 863 project under Grant(No. 2015AA015905).

II. THE PROPOSED METHOD

The aim of the proposed method in this paper is to remove noise from contaminated video, retain much more texture details and produce fewer blocking artifacts. In many literatures, a plenty of noise models are applied in video or image processing. In this study, we adopt an additive white Gaussian noise (AWGN) model.

Consider an observed video stream as a noised image sequence $z : X \times T$ defined as

$$z(\boldsymbol{x}, t) = y(\boldsymbol{x}, t) + \eta(\boldsymbol{x}, t), \boldsymbol{x} \in X, t \in T \tag{1}$$

where $y(\cdot, \cdot)$ is a matrix representing the original (unknown) video, $\eta(\cdot, \cdot) \sim N(0, \sigma^2)$ is i.i.d. AWGN. And $(\boldsymbol{x}, t)$ is the 3-D spatiotemporal coordinate in gray space or the 4-D spatiotemporal coordinate in color space. For simplicity, only video in the gray space is studied, thus $X \subset \mathbb{Z}^2$, $T \subset \mathbb{Z}$. To obtain $y(\cdot, \cdot)$, we propose a method designed as

$$y(\boldsymbol{x}, t) = P. * y_f(\boldsymbol{x}, t) + (1 - P). * y_c(\boldsymbol{x}, t) \tag{2}$$

where P is a texture metric matrix containing overall characters of superpixels for a video stream, derived in section II-A; $y_f(\cdot, \cdot)$ and $y_c(\cdot, \cdot)$ are outputs of noise filters with structure variance σ_f^2 and σ_c^2 respectively, as shown in section II-B.

A. Adaptive Video Texture Metric Based on Superpixel

During video denoising filtering, some texture details may be smooth-filtered causing unwanted artifacts. To overcome this defect, we first adopt the metric P to estimate video texture. Inspired by [8], P is calculated by followings.

1) Calculation of Orientation Gradient: When gradients of neighbours are isotropic, the relative texture is smooth; when gradients are anisotropic, the texture is poignant. So we adopt gradient operator Sober (3) and (4) as the basic operators for texture estimation.

$$D_h = \frac{1}{2}\begin{bmatrix} 0 & 0 & 0 \\ -1 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}, D_v = \frac{1}{2}\begin{bmatrix} 0 & -1 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \tag{3}$$

$$D_h = \frac{1}{8}\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, D_v = \frac{1}{8}\begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \tag{4}$$

where D_h and D_v are horizontal and vertical Sober filter respectively. For the i-th frame I_i in a video, we obtain its gradient as

$$\left[G_{i,h}, G_{i,v}\right] = \left[D_h \odot I_i, D_v \odot I_i\right], 1 \leq i \leq N_f \tag{5}$$

where $G_{i,h}$ and $G_{i,v}$ are horizontal and vertical gradients of I_i respectively, N_f is the number of video frames.

2) Superpixel Segmentation: There exists many superpixel algorithms to segment an image. In this paper, we use SLIC [9] to split the frames into many paths. For the frame I_i, we divide it into N_i paths. Then we obtain superpixel label of I_i, defined as

$$L_i = \{1, \ldots, N_i\} \tag{6}$$

And the flag index of each superpixel path is defined as

$$F_{I_{i,k}} = k, k \in L_i \tag{7}$$

where $I_{i,k}$ represents the k-th superpixel path of I_i.

3) Label of Superpixel Gradient Path: Here we define $G_{i,k,h}$ and $G_{i,k,v}$ as the horizontal and vertical gradients of $I_{i,k}$ respectively, shown in equation (8).

$$\left[G_{i,k,h}, G_{i,k,v}\right] = \left[G_{i,h}(x, y), G_{i,v}(x, y)\right], (x, y) \in X_{I_{i,k}} \tag{8}$$

where $X_{I_{i,k}}$ denotes the set of coordinates of $I_{i,k}$.Then the gradient $G_{i,k}$ of $I_{i,k}$ is obtained by jointing $G_{i,k,h}$ and $G_{i,k,v}$.

$$G_{i,k} = \begin{bmatrix} \vdots & \vdots \\ G_{i,k,h}(m) & G_{i,k,v}(m) \\ \vdots & \vdots \end{bmatrix}, m \in \{1, \ldots, \omega_{i,k}\} \tag{9}$$

where $\omega_{i,k}$ represents the scale size of $I_{i,k}$. $G_{i,k}$ is a $\omega_{i,k}$-by-2 matrix. $\left[G_{i,k,h}(m), G_{i,k,v}(m)\right]$ denotes the gradients of $I_{i,k}$ at the coordinate (x_m, y_m).

4) Adaptive Superpixel SVD: We define a correlative covariance matrix C as

$$C = G_{i,k}^T G_{i,k} \tag{10}$$

For simplicity, we use G to replace $G_{i,k}$. After SVD of G, it is represented as

$$G = U \begin{bmatrix} s_1 & 0 \\ 0 & s_2 \end{bmatrix} V_T \tag{11}$$

$$U = \begin{bmatrix} \vdots & \vdots \\ u_{n,1} & u_{n,2} \\ \vdots & \vdots \end{bmatrix}_{\omega_{i,k} \times 2}, V = \begin{bmatrix} v_{11} & v_{12} \\ v_{21} & v_{22} \end{bmatrix} \tag{12}$$

where U and V are both orthonormal matrix, defined in equation (12). s_1 and s_2 are the principal eigenvalue of matrix G. Combining (11) and (12), we obtain C as

$$C = \begin{bmatrix} v_{11}^2 s_1^2 + v_{21}^2 s_2^2 & v_{11} v_{21}(s_1^2 - s_2^2) \\ v_{11} v_{21}(s_1^2 - s_2^2) & v_{21}^2 s_1^2 + v_{11}^2 s_2^2 \end{bmatrix} \tag{13}$$

However, G in practice is interfered by noise, thus we use $\hat{G}$ to represent the real gradient of superpixel path and $\hat{C}$ to represent the correlative covariance. As a result, equation (11) and (13) are modified as (14) and (15).

$$\hat{G} = \hat{U} \begin{bmatrix} \hat{s_1} & 0 \\ 0 & \hat{s_2} \end{bmatrix} \hat{V}^T \tag{14}$$

$$\hat{C} = \begin{bmatrix} \hat{v_{11}}^2 \hat{s_1}^2 + \hat{v_{21}}^2 \hat{s_2}^2 & \hat{v_{11}} \hat{v_{21}}(\hat{s_1}^2 - \hat{s_2}^2) \\ \hat{v_{11}} \hat{v_{21}}(\hat{s_1}^2 - \hat{s_2}^2) & \hat{v_{21}}^2 \hat{s_1}^2 + \hat{v_{11}}^2 \hat{s_2}^2 \end{bmatrix} \tag{15}$$

Here we set the gradient of superpixel-noise path G_η

$$\hat{G} = G + G_\eta \tag{16}$$

$$\hat{C} = G^T G + G^T G_\eta + G_\eta^T G + G_\eta^T G_\eta \tag{17}$$

Since C is unknown, so we use $E(\hat{C})$ to estimate $E(C)$ as

$$E(C) \simeq E(\hat{C}) = E(G^T G + G^T G_\eta + G_\eta^T G + G_\eta^T G_\eta) \tag{18}$$

Assuming noise is AWGN, G and G_η satisfy the formula as

$$\begin{cases} E(G^T G_\eta) = 0 \\ E(G_\eta^T G) = 0 \\ E(G_\eta^T G) = \begin{bmatrix} \xi\omega_{i,k}\sigma^2 & 0 \\ 0 & \xi\omega_{i,k}\sigma^2 \end{bmatrix} \end{cases} \tag{19}$$

where $\xi = 1/2$ when using operator (3), $\xi = 3/16$ when using operator(4). Thus, the relationship between original and realistic single value is obtained as

$$\begin{cases} \hat{s_1}^2 - \hat{s_2}^2 = \alpha(s_1^2 - s_2^2) \\ \hat{s_1}^2 + \hat{s_2}^2 = \beta(s_1^2 + s_2^2) + \xi\gamma\omega_{i,k}\sigma^2 \end{cases} \tag{20}$$

where

$$\alpha = \frac{v_{11}v_{21}}{\hat{v_{11}}\hat{v_{21}}}, \beta = \frac{v_{11}^2 + v_{21}^2}{\hat{v_{11}}^2 + \hat{v_{21}}^2}, \gamma = \frac{2}{\hat{v_{11}}^2 + \hat{v_{21}}^2} \tag{21}$$

Inspired by [10], [11], we use

$$\rho = \frac{\hat{s_1}^2 - \hat{s_2}^2}{\hat{s_1}^2 + \hat{s_2}^2} \tag{22}$$

to characterize the noise-texture of the superpixel path. Here $\alpha, \beta, \gamma, \xi, \omega_{i,k}$ are constants in a fixed path respectively, the ρ only changes with s_1,s_2 and σ^2. Now, suppose σ^2 is constant, ρ will be larger when there is a prominent texture ($s_1 \gg s_2$) in the path. As a result, we define

$$\rho = \frac{s_1^2 - s_2^2}{s_1^2 + s_2^2} \tag{23}$$

as a texture metric for the superpixe path.

5) Obtainment Of Video Texture Metric: Considering ρ varying with superpixel paths, we use $\rho_{i,k}$ to denote the texture metric of $I_{i,k}$. For more explicitly, we introduce superpixel texture metric path $\hat{I_{i,k}}$ corresponding to $I_{i,k}$, defined as

$$\hat{I_{i,k}} = \{(x,y) | V(x,y) = \rho_{i,k}, (x,y) \in X_{i,k}\} \tag{24}$$

where $V(x,y)$ denotes the value of $\hat{I_{i,k}}$ at coordinate (x,y). Then P is obtained by aggregating all $\hat{I_{i,k}}$, described as

$$P = \{\hat{I_{i,k}} | i \in N_f, k \in L_i\} \tag{25}$$

B. Combination of Structure

In order to reconstruct the video stream with good denoising performance and less block artifacts, we adopt a new adaptive structure variance. This variance is estimated by adopting fine and coarse structures. Then a noise filter based on the estimated weights of different structure is applied. The details are described by followings.

1) Obtaining Structure Variance: First, We use the calculated $\rho_{i,k}$ as a test statistic to decide whether a superpixel path has a stronger texture. By introducing eigenvalues and condition numbers of random metrics [12], we obtained the possibility density $f_p(\rho_{i,k})$ of $\rho_{i,k}$, described as

$$f_p(\rho_{i,k}) = (\omega_{i,k} - 1)\rho_{i,k}(1 - \rho_{i,k}^2)^{(\omega_{i,k}-2)/3} \tag{26}$$

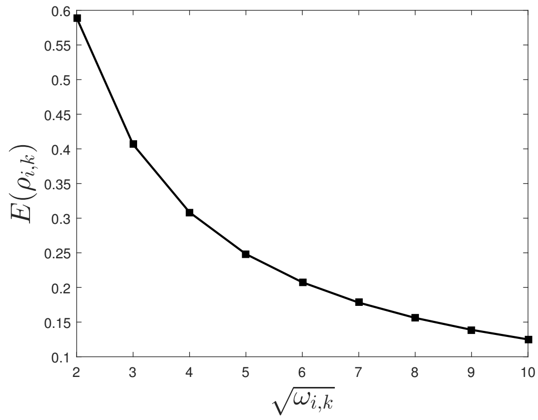

Fig. 2. Expected value of $\rho_{i,k}$

Thus we obtain the expected value of $\rho_{i,k}$ described in equation (27). Fig.2 gives the more intuitional description of $E(\rho_{i,k})$.

$$E(\rho_{i,k}) = \begin{cases} \dfrac{(\omega_{i,k}-1)!!}{\omega_{i,k}!!}, & \omega_{i,k} \ \ is \ \ an \ \ odd \\ \dfrac{(\omega_{i,k}-1)!!}{\omega_{i,k}!!} \cdot \dfrac{\pi}{2}, & \omega_{i,k} \ \ is \ \ an \ \ even \end{cases} \tag{27}$$

Then we set a threshold τ equal to $E(\rho_{i,k})$, and mark $I_{i,k}$ as fine structure path if $\rho_{i,k} > \tau$ and coarse structure path if $\rho_{i,k} < \tau$. Finally, variances of all $I_{i,k}$ are calculated, which are labeled as $\sigma_{i,k,f}^2$ corresponding to fine structure and $\sigma_{i,k,c}^2$ corresponding to coarse structure.

2) Filtering with Weights: First, fine structure variance σ_f^2 and coarse structure variance σ_c^2 of video stream are obtained respectively as

$$\begin{cases} \sigma_f^2 = w_f * \min\limits_{i \in N_f}\{\operatorname*{mean}\limits_{k \in L_i}(\sigma_{i,k,f}^2)\} \\ \sigma_c^2 = w_c * \min\limits_{i \in N_f}\{\operatorname*{mean}\limits_{k \in L_i}(\sigma_{i,k,c}^2)\} \end{cases} \tag{28}$$

where w_f and w_c are utilized to adjust σ_f^2 and σ_c^2 , of which the general formula w is described as

$$w = \frac{1}{1 + ab^{\sqrt{10(c-\Delta)}}} \tag{29}$$

where

$$\Delta = \min\limits_{i \in N_f}\{\operatorname*{mean}\limits_{k \in L_i}(\sigma_{i,k,c}^2 - \sigma_{i,k,f}^2)\} \tag{30}$$

a, b and c are constants. Then the fine structure video stream $y_f(\cdot,\cdot)$ and the coarse structure video stream $y_c(\cdot,\cdot)$ are obtained respectively as

$$\begin{cases} y_f(\boldsymbol{x},t) = Filter\{z(\boldsymbol{x},t), \sigma_f^2\} \\ y_c(\boldsymbol{x},t) = Filter\{z(\boldsymbol{x},t), \sigma_c^2\} \end{cases} \tag{31}$$

where $Filter\{z(\boldsymbol{x},t), \sigma_f^2\}$ and $Filter\{z(\boldsymbol{x},t), \sigma_c^2\}$ denotes the process of filtering with σ_f^2 and σ_c^2 for $z(\cdot,\cdot)$. Otherwise, we observe $\rho_{i,k}$ can estimate the probability of different structures. Thus we utilize P as a weight for filtering. Finally, the reconstructed video stream is obtained by equation (2).

TABLE I
COMPARISON OF PSNR

Method	Video / σ	Tennis	coastguard	gbicycle	gbus	gflower	gforeman	gsalesman	gmissa
BM3D	10	33.34	34.23	37.52	32.73	32.28	37.41	36.75	39.27
	20	**29.67**	31.58	34.03	28.76	28.29	34.60	**33.45**	37.57
	30	**27.07**	**29.82**	31.62	26.75	25.98	32.78	**31.27**	36.23
BM4D	10	34.17	35.04	37.77	32.98	32.37	37.67	36.64	39.72
	20	28.80	31.50	34.30	29.08	28.43	**34.79**	32.67	**37.42**
	30	25.84	29.46	**32.03**	27.01	26.14	33.04	30.37	35.84
Ours	10	**34.19**	**35.14**	**37.97**	**33.48**	**32.66**	**37.82**	**36.78**	**39.73**
	20	29.08	**31.60**	**34.34**	**29.12**	**28.45**	33.47	32.63	37.22
	30	26.04	29.48	31.98	**27.08**	**26.25**	**33.34**	30.57	34.57

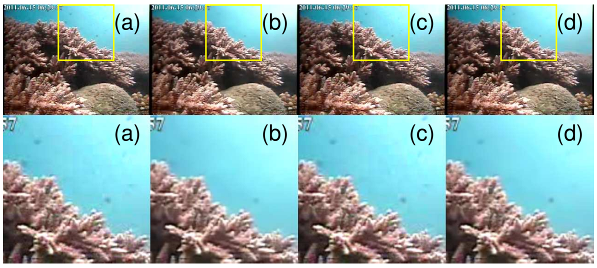

Fig. 3. Denoising results for video stream named Standard2. (a) is the first frame of the video. (b) to (d) respectively denotes the filtering results of different algorithms as BM3D, BM4D, and ours.

III. EXPERIMENT RESULTS

Simulation results are given in comparison of BM3D and BM4D. All the methods are tested and compared with 40 grayscale video stream under various noise levels and 13 natural video stream[1]. Section III-A and section III-B are subjective and objective assessment respectively. The major parameters of the proposed method are set as TABLE II.

TABLE II
MAJOR PARAMETERS OF THE PROPOSED METHOD

Parameters						
δ	$\omega_{i,k}$	w_f		w_c		c
		a	**b**	**a**	**b**	
0.15	81	0.75	0.6	0.79	0.63	4.9

A. Subjective Assessment

As shown in Fig.3 (b-c), our method can adaptively adjust the proportion between fine structure and coarse structure to retain detailed texture and reduce block artifacts on the process of video denoising. As shown in Fig.3 (d), our method retains as much detailed texture as BM4D and produces fewer block artifacts than BM4D.

B. Objective Assessment

The performance comparison is shown in TABLE I.

[1] http://groups.inf.ed.ac.uk/f4k/index.html

IV. CONCLUSION

In this paper, we present an innovative denoising method combining a new adaptive texture metric based on superpixel and a new structure variance. By utilizing the video texture metric to weight the fine and coarse video stream, major artifacts produced by traditional methods are eliminated dramatically. Experiment results show that the proposed method can achieve better performance compared with two state-of-art algorithms.

REFERENCES

[1] H. Talebi and P. Milanfar, "Global image denoising," *Image Processing, IEEE Transactions on*, vol. 23, no. 2, pp. 755–768, 2014.

[2] W. Dong, G. Li, G. Shi, X. Li, and Y. Ma, "Low-rank tensor approximation with laplacian scale mixture modeling for multiframe image denoising," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 442–449.

[3] H. Yue, X. Sun, J. Yang, and F. Wu, "Image denoising by exploring external and internal correlations," *Image Processing, IEEE Transactions on*, vol. 24, no. 6, pp. 1967–1982, 2015.

[4] K. Dabov, A. Foi, and K. Egiazarian, "Video denoising by sparse 3d transform-domain collaborative filtering," in *Signal Processing Conference, 2007 15th European*. IEEE, 2007, pp. 145–149.

[5] M. Maggioni, G. Boracchi, A. Foi, and K. Egiazarian, "Video denoising, deblocking, and enhancement through separable 4-d nonlocal spatiotemporal transforms," *Image Processing, IEEE Transactions on*, vol. 21, no. 9, pp. 3952–3966, 2012.

[6] Z. Liu, L. Yuan, X. Tang, M. Uyttendaele, and J. Sun, "Fast burst images denoising," *ACM Transactions on Graphics (TOG)*, vol. 33, no. 6, p. 232, 2014.

[7] L. Shao, R. Yan, X. Li, and Y. Liu, "From heuristic optimization to dictionary learning: a review and comprehensive comparison of image denoising algorithms," *Cybernetics, IEEE Transactions on*, vol. 44, no. 7, pp. 1001–1013, 2014.

[8] L. Ding, G. Li, R. Wang, and W. Wang, "Video pre-processing with jnd-based gaussian filtering of superpixels," in *IS&T/SPIE Electronic Imaging*. International Society for Optics and Photonics, 2015, pp. 941 004–941 004.

[9] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Susstrunk, "Slic superpixels compared to state-of-the-art superpixel methods," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 34, no. 11, pp. 2274–2282, 2012.

[10] L. Li, R. Wang, W. Wang, and W. Gao, "A low-light image enhancement method for both denoising and contrast enlarging," in *Image Processing (ICIP), 2015 IEEE International Conference on*. IEEE, 2015, pp. 3730–3734.

[11] X. Zhu and P. Milanfar, "Automatic parameter selection for denoising algorithms using a no-reference measure of image content," *Image Processing, IEEE Transactions on*, vol. 19, no. 12, pp. 3116–3132, 2010.

[12] A. Edelman, "Eigenvalues and condition numbers of random matrices," *SIAM Journal on Matrix Analysis and Applications*, vol. 9, no. 4, pp. 543–560, 1988.