International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 1, Issue 3, November 2014

Video Denoising Using Sparse and Redundant Representations

B.Anandhaprabakaran[1], S.David[2]

[1, 2]Assistant Professor
Department of Electronics and Communication Engineering,
Sri Ramakrishna Engineering College, [2]Sri Eashwar Engineering College, Coimbatore, Tamil Nadu, India

Abstract*:* The quality of video sequences is often reduced by noise, usually assumed white and Gaussian, being superimposed on the sequence. When denoising image sequences, rather than a single image, the temporal dimension can be used for gaining in better denoising performance, as well as in the algorithms' speed. Such correlations are further strengthened with a motion compensation process, for which a Fourier domain noise-robust cross correlation algorithm is proposed for motion estimation. This algorithm relies on sparse and redundant representations of small patches in the images. Three different extensions are offered, and all are tested and found to lead to substantial benefits both in denoising quality and algorithm complexity, compared to running the single image algorithm sequentially.

Keywords: Cross Correlation (CC), Motion estimation, K-SVD, OMP, Sparse representations, Video denoising.

I. INTRODUCTION

VIDEO SIGNALS are often contaminated by noise during acquisition and transmission. Removing/reducing noise in video signals (or video denoising) is highly desirable, as it can enhance perceived image quality, increase compression effectiveness, facilitate transmission bandwidth reduction, and improve the accuracy of the possible subsequent processes such as feature extraction, object detection, motion tracking and pattern classification.
Video denoising algorithms may be roughly classified based on two different criteria: whether they are implemented in the spatial domain or transform domain and whether motion information is directly incorporated. The high degree of correlation between adjacent frames is a "blessing in disguise" for signal restoration. On the one hand, since additional information is available from nearby frames, a better estimate of the original signal is expected. On the other hand, the process is complicated by the presence of motion between frames. Motion estimation itself is a complex problem and it is further complicated by the presence of noise.One suggested approach that utilizes the temporal redundancy is motion estimation [4]. The estimated trajectories are used to filter along the temporal dimension, either in the wavelet domain [5] or the signal domain [4]. Spatial filtering may also be used, with stronger emphasis in areas in which the motion estimation is not as reliable. A similar approach described in [6] detects for each pixel weather it has undergone motion or not. Spatial filtering is applied to each image, and for each pixel with no motion detected, the results of the spatial filtering are recursively averaged with results from previous frames.

A different approach to video denoising is treating the image sequence as a 3-D volume, and applying various transforms to this volume in order to attenuate the noise. Such transforms can be a Fourier-wavelet transform, an adaptive wavelet transform [7], or a combination of 2-D and 3-D dual-tree complex wavelet transform [8].

A third approach towards video denoising employs spatiotemporal adaptive average filtering for each pixel. The method described in [9] uses averaging of pixels in the neighborhood of the processed pixel, both in the current frame and the previous one. The weights in this averaging are determined according to the similarity of 3 3 patches around the two pixels matched. The method in [10] extends this approach by considering a full 3-D neighborhood around the processed pixel (which could be as large as the entire sequence), with the weights being computed using larger patches and, thus, producing more accurate weights. The current state-of-the-art reported in [11] also finds similar patches in the neighborhood of each pixel; however, instead of a weighted averaging of the centers of these

patches, noise attenuation is performed in the transform domain.

In this paper, we explore a method that utilizes sparse and redundant representations for image sequence denoising, [1] In the single image setting, the K-SVD algorithm, is used to train a sparsifying dictionary for the corrupted image, forcing each patch in the image to have a sparse representation describing its content. Put in a maximum *a posteriori* probability (MAP) framework, the developed algorithm in [1] leads to a simple algorithm with state-of-the-art performance for the single image denoising application.

In this paper, a video denoising method based on a spatiotemporal model of motion compensated frames is proposed. By considering 3-D (spatio-temporal) patches, a propagation of the dictionary over time, and averaging that is done on motion compensated neighboring patches both in space and time. As the dictionary of adjacent frames (belonging to the same scene) is expected to be nearly identical, the number of required iterations per frame can be significantly reduced It is also important to be aware that video signals are more than simple 3-D extensions of 2-D static image signals, where the major distinction is the capability of representing motion information [12]. If the motion between frames is properly compensated, then the temporal correlations between the patches can be enhanced. Effective motion compensation relies on reliable motion estimation. we opt to use global motion estimation methods based on cross correlation (CC) [13]-[15]. The challenge here is that the motion information must be estimated from noisy video signals rather than the noise free original signals. To address this issue, another important aspect of our paper was the and incorporation of a novel noise-robust motion estimation algorithm, which has not been carefully examined in previous video denoising algorithms. All of these modifications lead to substantial benefits both in complexity and denoising performance.

The structure of the paper is as follows. In Section II, we describe the principles of sparse and redundant representations and their deployment to single image denoising. Section III discusses the generalization to video, discussing various options of using the temporal dimension with their expected benefits and drawbacks. Details on the motion estimation algorithm will be given in Section IV. Each proposed extension is experimentally validated in section V. A performance comparison of other methods and the one introduced in this paper is given. Section VI summarizes and concludes the paper.

II. Image Denoising Using Sparsity And Redundancy

A method of denoising images based on sparse and redundant representations is developed and reported in [1] and [2]. In this section, we provide a brief description of this algorithm, as it serves as the foundation for the video denoising we develop in Section III.

A noisy image **Y** results from noise v superimposed on an original image **X** . We assume the noise to be white, zero mean
Gaussian noise, with a known standard deviation **σ**

$$\mathbf{Y=X+V \text{ where } V \sim(0,\sigma^2 I)} \quad (1)$$

The basic assumption of the denoising method developed in [1] and [2] is that each image patch (of a fixed size) can be represented as a linear combination of a small subset of patches (*atoms*), taken from a fixed *dictionary*. Using this assumption, the denoising task can be described as an energy minimization procedure. The following functional describes a combination of three penalties to be minimized

$$f_{still}\left(\{\alpha_{ij}\}_{ij}, X\right) = \lambda||X-Y||_2^2 + \sum_{ij\varepsilon\Omega} ||D\alpha_{ij} - R_{ij}X||_2^2 + \sum_{ij\varepsilon\Omega} \mu_{ij}||\alpha_{ij}||_0 \quad (2)$$

The first term demands a proximity between the measured image, Y, and its denoised (and unknown) version X. The second term demands that each patch from the reconstructed image (denoted by $R_{ij}X$) can be represented up to a bounded error by a dictionary D, with coefficients α_{ij} . The third part demands that the number of coefficients required to represent any patch is small. The values μ_{ij} are patch-specific weights. Minimizing this functional with respect to its unknowns yields the denoising algorithm.

The choice of D is of high importance to the performance of the algorithm. In [1] it is shown that training can be done by minimizing (2) with respect toD as well (in addition to X and α_{ij}). The proposed algorithm in [1] is an iterative block-coordinate relaxation method, that fixes all the unknowns apart from the one to be updated, and alternates between the following update stages.

1) *Update of the sparse representations* $\{\alpha_{ij}\}$: Assuming that D and X are fixed, we solve a set of problems of the form

$$\widehat{\alpha_{ij}} = \underset{\alpha}{argmin} ||D\alpha - R_{ij}X||_2^2 + \mu||\alpha||_0 \quad (3)$$

per each location [i,j]. This means that we seek for each patch in the image the sparsest vector to describe it using atoms from D. In [1], the orthogonal matching pursuit (OMP) algorithm is used for this task .

2) Update *the dictionary* D: In this stage, we assume that X is fixed, and we update one atom at a time in D, while also updating the coefficients in $\{\alpha_{ij}\}_{ij}$ that use it. This is done via a rank-one approximation of a residual matrix, as described in [16].

3) *Update the estimated image* X : After several rounds of updates of $\{\alpha_{ij}\}_{ij}$ and D ,the final output image is computed by fixing these unknowns and minimizing (2) with respect to X . This leads to the quadratic problem

$$\hat{X} = \underset{X}{argmin}\ \lambda||X - Y||_2^2 + \sum_{ij} ||D\alpha_{ij} - R_{ij}X||_2^2 \quad (4)$$

which is solved by a simple weighting of the represented patches with overlaps, and the original image Y .

The improved results obtained by training a dictionary based on the noisy image itself stem from the dictionary adapting to the content of the actual image to be denoised. An added benefit is that the K-SVD algorithm has noise averaging built into it, by taking a large set of noisy patches and creating a small, relatively clean representative set.

III. Extension To Video Denoising

In this section, we describe, step by step, the proposed extensions to handling image sequences. Training a single dictionary for the entire sequence is problematic; The scene is expected to change rapidly, and objects that appear in one frame might not be there five or ten frames later. This either means that the dictionary will suit some images more than others, or that it would suit all of the images but only moderately so. Obtaining state-of-the-art denoising results requires better adaptation. An alternative approach could be proposed by defining a locally temporal penalty term, that on one hand allows the dictionary to adapt to the scene, and on the other hand, exploits the benefits of the temporal redundancy.

$$f_{Video}^{(t)}\left(\{\alpha_{ij}\}_{ij}, X_t, D_t\right) = \lambda||X_t - Y_t||_2^2 + \sum_{ij\varepsilon\Omega} ||D_t\alpha_{ij} - R_{ijt}X||_2^2 + \sum_{ij\varepsilon\Omega} \mu_{ij}||\alpha_{ij}||_0 \quad (5)$$

Defined for t=1,2,…..,T.

The temporal repetitiveness of the video sequence can be further used to improve the algorithm. As consecutive images X_t and X_{t-1} are similar, their corresponding dictionaries are also expected to be similar. This temporal coherence can help speed-up the algorithm. Fewer training iterations are necessary if the initialization for the dictionary D_t is the one trained for the previous image. The number of training iterations should not be constant, but rather depend on the noise level. It appears that the less noisy the sequence, the more training iterations are needed for each image. In higher noise levels, adding more iterations hardly results in any denoising improvement.

Only patches centered in the current image are used for training the dictionary and cleaning the image. A compromise between temporal locality and exploiting the temporal redundancy is again called for. This compromise is achieved by using motion compensated patches of the image currently denoised, both in training and cleaning. Introducing this into the penalty term in (5) leads to the modified version

$$f_{Video}^{(t\pm)}\left(\{\alpha_{ijk}\}_{ijk}, X_t, D_t\right) = \lambda||X_t - Y_t||_2^2 + \sum_{ij\varepsilon\Omega}\sum_{k=t-\Delta t}^{t+\Delta t} \mu_{ijk}||\alpha_{ijk}||_0 + \sum_{ij\varepsilon\Omega}\sum_{k=t-\Delta t}^{t+\Delta t} ||D_t\alpha_{ijk} - R_{ijk}X||_2^2 \quad (6)$$

Defined for t=1,2,…..,T.

(i.e., the patches are taken from three frames).

To gauge the possible speed-up and improvement achieved by temporally adaptive and propagated dictionary, we test the required number of iterations to obtain similar results to the non propagation alternative. Several options for the number of training iterations that follow the dictionary propagation are compared to the non propagation (using 15 training iterations per image) option. Dictionary is trained after propagating the dictionary (from image #10), using 4 training iterations for each image. This comparison shows

that propagation of the dictionary leads to a cleaner version with clearer and sharper texture atoms. These benefits are attributed to the memory induced by the propagation. Indeed, when handling longer sequences, we expect this memory feature of our algorithm to further benefit in denoising performance.

At high noise levels, the redundancy factor (the ratio between the number of atoms in the dictionary to the size of an atom) should be smaller. At high noise levels, obtaining a clean dictionary requires averaging of a large number of patches for each atom. This is why only a relatively small number of atoms is used. At low noise levels, many details in the image need to be represented by the dictionary. Noise averaging takes a more minor role in this case. This calls for a large number of atoms, so they can represent the wealth of details in the image.

IV. NOISE-ROBUST MOTION ESTIMATION

One of the challenges in the implementation of the above algorithm is to estimate motion in the presence of noise. Here, we propose a simple but reliable noise-robust CC method[3] for global motion estimation at integer pixel precision. The limitation of using global motion estimation is that it cannot account for rotation, zooming and local motion. Let $F_1(v)$ and $F_2(v)$ represent two image frames, where v is a spatial integer index vector for the underlying 2-D rectangular image lattice. A classical approach to estimating a global motion vector between the two frames is the cross correlation method which is based on the observation that when $F_2(v)$ is a shifted version of $F_1(v)$, the position of the peak in the CC function between $F_1(v)$ and $F_2(u)$ corresponds to the motion vector. Despite the simplicity of the idea, the computation of the CC function is often costly. An equivalent but more efficient approach is to use the Fourier transform method: Let $F(\omega) = F\{f(v)\}$ represents the 2-D Fourier transform of an image frame, where F denotes the Fourier transform operator. Then, the CC function can be computed as

$$k_{cc}(v) = f^{-1}\{Y(\omega)\} \quad (7)$$

$$Y(\omega) = F_1(\omega)F_2^*(\omega) \quad (8)$$

The estimated motion vector is given by

$$V_{pot} = \underset{v}{argmax}\, K_{cc}(V) \quad (9)$$

An interesting variation of this approach is the phase correlation (PC) method where the Fourier spectrum is normalized in the frequency domain to have unit energy across all frequencies. The phase correlation function is given by

$$K_{pc}(v) = f^{-1}\left\{\frac{Y(\omega)}{|Y(\omega)|}\right\} \quad (10)$$

To have a close look, let us assume that $F_2(v)$ is simply a shifted version of $F_1(v)$, i.e., $F_2(\mathbf{v}) = F_1(\mathbf{v}-\Delta\mathbf{v})$. Based on the shifting property of the Fourier transform, we have $F2(\boldsymbol{\omega}) = F1(\boldsymbol{\omega})\exp\{-j\boldsymbol{\omega}\boldsymbol{T}\Delta\mathbf{v}\}$ and $Y(\boldsymbol{\omega}) = |F1(\boldsymbol{\omega})|2\exp\{j\boldsymbol{\omega}^T\Delta\mathbf{v}\}$, and thus

$$K_{pc}(v) = f^{-1}\{\exp\{jw^T\Delta u\}\} = \delta(v+\Delta v) \quad (11)$$

Which creates an impulse at the true motion vector position and is zero everywhere else. Both CC and PC methods were designed with the assumption that there is no noise in the images. With the presence of noise, their performance degrades. Our noise-robust cross correlation (NRCC) function is defined as

$$k_{nrcc}(v) = f^{-}\{Y(w)\left\{1-\frac{|N(w)|^2}{|Y(w)|}\right\}\} \quad (12)$$

Where $|N(\omega)|^2$ is the noise power spectrum (in the case of white noise, $|N(\omega)|^2$ is a constant). To better understand this, it is useful to formulate the three approaches (PC, CC, and NRCC) using a unified framework. In particular, each method can be viewed a specific weighting scheme in the Fourier domain

$$K(v) = f^{-1}\{W(w)\exp\{jw^T\Delta v\}\} \quad (13)$$

where the differences lie in the definition of the weighting function $W(\boldsymbol{\omega})$

$$W_{pc}(w) \equiv 1$$
$$W_{cc}(w) = |F_1(w)|^2$$
$$W_{nrcc}(w) = |F_1(w)|^2 - |N(w)|^2 \quad (14)$$

The PC method assigns uniform weights to all frequencies, the CC method assigns the weights based on the total signal power (which is the sum of signal and noise power), while the NRCC method assigns the weights

proportional to the signal power only (by removing the noise power part). It converges to the CC method when the images are noise-free.

V. VALIDATION

The simulation results of the paper have been discussed here. All video sequences are in YUV 4:2:0 format, but only the denoising results of the luma channel are reported here for algorithm validation. Two objective criteria, namely the PSNR and the SSIM, were employed to provide quantitative quality evaluations of the denoising results.

PSNR is defined as

$$PSNR = 10\log\left(\frac{L^2}{MSE}\right) \quad (15)$$

here L is the dynamic range of the image (for 8 bits/pixel images, L = 255) and MSE is the mean squared error between the original and distorted images.

SSIM is first calculated within local windows using

$$SSIM(x,y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \quad (16)$$

here **x** and **y** are the image patches extracted from the local window from the original and distorted images, respectively. Mx, σ_x^2, and σ_{xy} are the mean, variance, and cross-correlation computed within the local window, respectively. The overall SSIM score of a video frame is computed as the average local SSIM scores.

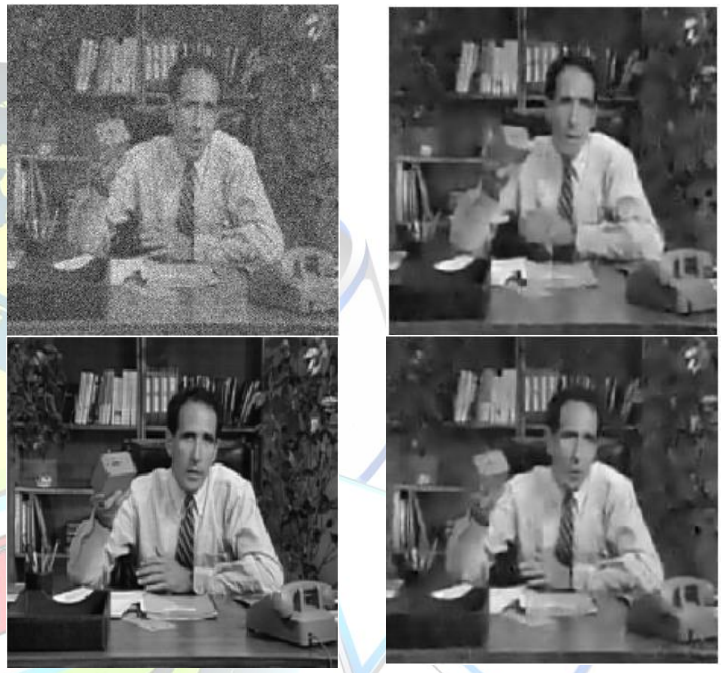
Besides the running time of this method can fill the requirements of the real time processing. To testify the advantage of our method, the comparison between present and K_SVD method is shown in Fig. 1.

TABLE I
PSNR COMPARISONS WITH LATEST VIDEO DENOISING ALGORITHMS

Noise std(σ)	10	15	20	50
K_SVD[1]	35.89	33.30	31.80	24.12
K_SVD[2]	37.95	35.17	33.33	26.03
VBM3D	38.33	36.6	35.12	28.49
Proposed	38.01	35.98	34.23	27.31

VI. CONCLUSION

In this paper an image sequence denoising algorithm based on sparse and redundant representations is used for denoising process. Applying motion estimation/motion compensation is effective in enhancing the correlations and thus, improving the performance of the denoising process. Three extensions were used: the use of spatio-temporal (3-D) atoms, dictionary propagation coupled with fewer training iterations, and an extended patch-set for dictionary training and image cleaning. The experimental results show the effectiveness of this method.



(a1) (a2)

Fig.1 Denoising results of Frame 10 in Salesman sequence corrupted with noise standard deviation σ = 20. (a1)–(a4) Image frames in the original, noisy, and K_SVD[1],Proposed one denoised sequences.

REFERENCES

[1]. M. Elad and M. Aharon, "Image denoising via sparse and redundant representation over learned dictionaries," *IEEE Trans. Image Process.*, vol. 15, no. 12, pp. 3736–3745, Dec. 2006.

[2]. M. Protter and M. Elad, "Image sequence denoising via sparse and redundant representations," *IEEE Trans. Image Process.*, vol. 18, no. 1, pp. 27–36, Jan. 2009.

[3]. Gijesh Varghese and Zhou Wang" Video Denoising Based on a Spatiotemporal Gaussian Scale Mixture Model"in Proc.IEEE *Int. Sym. Circuits Syst,* vol.20,*July 2010.*

[4]. S. M. M. Rahman, M. O. Ahmad, and M. N. S. Swamy, "Video denoising based on inter-frame statistical modeling of wavelet

coefficients," IEEE Trans. Circuits Syst. Video Tech., vol. 17, no. 2, pp. 187–198, Feb. 2007.

[5]. V. Zlokolica, A. Pizurica, and W. Philips, "Wavelet-domain video denoising based on reliability measures," *IEEE Trans. Circuits Syst. Video Tech.*, vol. 16, no. 8, pp. 993–1007, Aug. 2006.

[6]. A. Pizurica, V. Zlokolica, and W. Philips, "Combined wavelet domain and temporal video denoising," in Proc. IEEE Conf. Adv. Video Signal-*Based Surveillance*, Jul. 2003, pp. 334–341.

[7]. I. Selesnick and K. Y. Li, "Video denoising using 2D and 3D dual-tree complex wavelet transforms," presented at theWavelet

Applications in Signal and Image Processing X (SPIE), San Diego, CA, Aug. 2003.

[8]. N. M. Rajpoot, Z. Yao, and R. G. Wilson, "Adaptive wavelet restoration of noisy video sequences," presented at the Proc. Int. Conf. Image Processing, Singapore, Oct. 2004.

[9]. V. Zlokolica, A. Pizurica, and W. Philips, "Video denoising using multiple class averaging with multiresolution," in *Proc. Lecture Notes in Compute Science (VLVB03)*, 2003, vol. 2849, pp. 172–179.

[10]. A. Buades, B. Coll, and J. M. Morel, "Denoising image sequences does not require motion estimation," in *Proc. IEEE Conf. Advanced Video* and Signal Based Surveillance, Sep. 2005, pp. 70–74.

[11]. K. Dabov, A. Foi, and K. Egiazarian, "Video denoising by sparse 3D transform-domain collaborative filtering," presented at the Eur. Signal Processing Conf., Poznan, Poland, Sep. 2007.

[12]. Z. Wang and Q. Li, "Video quality assessment using a statistical model of human visual speed perception," *J. Opt. Soc. Am. A*, vol. 24, pp. B61–B69, Dec. 2007.

[13]. T. S. Huang and R. Y. Tsai, "Motion estimation," in *Image Sequence Analysis*, T. S. Huang, Ed. Berlin, Germany: Springer-Verlag, 1981, pp. 1–18.

[14]. R. Manduchi and G. A. Mian, "Accuracy analysis for correlation-based image registration algorithms," in *Proc. IEEE Int. Sym. Circuits Syst.*, 1993, pp. 834–837.

[15]. A. M. Tekalp, *Digital Video Processing*. Upper Saddle River, NJ: Prentice hall,1995.