7-2016

Recursive Non-Local Means Filter for Video Denoising with Poisson-Gaussian Noise

Redha A. Almahdi
University of Dayton, almahdir1@udayton.edu

Russell C. Hardie
University of Dayton, rhardie1@udayton.edu

Recursive Non-Local Means Filter for Video Denoising with Poisson-Gaussian Noise

Redha Almahdi
Department of Electrical and
Computer Engineering
University of Dayton
300 College Park, Dayton, Ohio 45469
Email: almahdir1@udayton.edu

Russell C. Hardie
Department of Electrical and
Computer Engineering
University of Dayton
300 College Park, Dayton, Ohio 45469
Email: rhardie1@udayton.edu

***Abstract*—In this paper, we describe a new recursive Non-Local means (RNLM) algorithm for video denoising that has been developed by the current authors. Furthermore, we extend this work by incorporating a Poisson-Gaussian noise model. Our new RNLM method provides a computationally efficient means for video denoising, and yields improved performance compared with the single frame NLM and BM3D benchmarks methods. Non-Local means (NLM) based methods of denoising have been applied successfully in various image and video sequence denoising applications. However, direct extension of this method from 2D to 3D for video processing can be computationally demanding. The RNLM approach takes advantage of recursion for computational savings, and spatio-temporal correlations for improved performance. In our approach, the first frame is processed with single frame NLM. Subsequent frames are estimated using a weighted combination of the current frame NLM, and the previous frame estimate. Block matching registration with the prior estimate is done for each current pixel estimate to maximize the temporal correlation. To address the Poisson-Gaussian noise model, we make use of the Anscombe transformation prior to filtering to stabilize the noise variance. Experimental results are presented that demonstrate the effectiveness of our proposed method. We show that the new method outperforms single frame NLM and BM3D.**

I. Introduction

In a broad range of real-world problems, digital image or/video is corrupted by random noise. The characteristic of random noise sources in digital images can generally be accurately modeled as Poisson-Gaussian noise. In many digital image applications such as medical imaging, fluorescence microscopy, and astronomy, only a limited amount of photons can be collected due to various physical constraints such as a light source with low power, short exposure time, and phototoxicity [1]. The two predominant random noise sources in digital image acquisition modalities are the stochastic nature of the photon-counting at the detectors (Poisson), and the intrinsic thermal and electronic fluctuations of the acquisition devices (Gaussian) [1]. A wide variety of denoising algorithms have employed the additive white Gaussian noise modeling to account for the second source of noise, such as [2]–[6]. However, the denoising algorithms for the Poisson-Gaussian noise is a less studied issue. The mixed Gaussian-Poisson noise model was introduced in [7]. Since the Poisson component models the signal-dependent of the errors, it cannot have a constant noise variance, which makes the premise for Poisson denoising very different from the case of additive white Gaussian noise with constant variance [8].

There are two main options to overcome this issue and maintain the denoising of images that have been corrupted by Poisson noise (signal-dependent). The first option is to specifically consider the signal dependent statistics of the noise model, and exploit these properties and observations in designing an effective denoising algorithm [1], [9]. The alternative option utilizes variance stabilization, which is performed through a three-step process. First, the variance stabilizing transforms (VSTs) such as the Anscombe transformation [10]. This procedure allows us to treat the noise as additive Gaussian with unitary variance [11]. Second, the noise is removed using our proposed denoising algorithm for additive white Gaussian noise. Finally, the inverse transformation VST is employed to the desired estimate signal to obtain the final estimate of the signal [11].

Recently, many algorithms have been proposed for denoising. The Non-Local Means (NLM) algorithm has been proposed by Buades et al. in [2], and it has been successfully employed in applications such as video and texture denoising [12]. NLM is employed to denoise images and video that have been corrupted by additive white Gaussian noise(AWGN). However, many authors have proposed methods to improve the NLM for Poisson noise models [12]–[15]. In addition, Block Matching and 3-D filtering (BM3D) has been proposed [6]. BM3D is a state-of-the-art method that uses vector distances between 2-D image blocks. Registered blocks are stacked into a 3-D group (grouping), and noise is reduced by collaborative Wiener filtering. This method has been extended for Poisson-Gaussian noise using the unbiased inverse of the generalized Anscombe transformation (GAT) [11].

In this paper, we present a recursive NLM (RNLM) algorithm for video processing. In our approach, we take advantage of recursion for computational savings, and spatio-temporal correlations for improved performance, compared with the direct 3D NLM. The first frame is processed with single-frame NLM. Subsequent frames are estimated using a weighted sum of pixels from the current frame, and a pixel from the previous frame estimate. Only the best matching patch from the previous estimate is incorporated into the current estimate. This is done to maximize the temporal correlation. We show that the new method outperforms single frame NLM and single frame

BM3D.

The remainder of this paper is organized as follows: the observation model and some benchmark methods are introduced in Section II, the proposed RNLM algorithm is presented in Section III, experimental results are presented in Section IV, and conclusions are presented in Section V.

II. Video Restoration

A. Observation Model

In this section, we present the observation model and notation for our video restoration methods. The degradation model is classified into a Poisson signal-dependent noise and a Gaussian signal-independent noise. The signal z is assumed to follow the Poisson-Gaussian noise model:

$$y_k(i) = \alpha \times P_k(i) + n_k(i), \tag{1}$$

for $i = 1, 2, ..., N$ and $k = 1, 2, ..., K$. Note that $y_k(i)$ represents pixel values in the observed frame in the video sequence. The pixel is obtained through a digital image acquisition device. The pixel is corrupted by Poisson $P_k(i)$ with mean $P((x_k(i))$, scaled by α. The parameter α is a positive scaling factor that is a camera gain parameter, and $n_k(i) \sim \mathcal{N}(0, \sigma_n^2)$ is additive zero-mean Gaussian noise with variance σ_n^2. The parameters of Poisson-Gaussian noise model are α and σ_n^2. The parameter α impacts the Poisson component, and σ_n^2 is the standard deviation of the additive Gaussian noise component. These are camera specific parameters depend on the CCD array in the camera and also on the integration time.

B. Single Frame Non-Local Means Filter

We begin by defining the single-frame NLM (SNLM) [2], upon which our method is built. Processing the frames from an image sequence individually, the SNLM output can be expressed as

$$\hat{x}_k(i) = \frac{1}{W_{k,i}} \sum_{j \in \varepsilon(i)} w_k(i,j) y_k(j), \tag{2}$$

where $\hat{x}_k(i)$ denotes the estimated image at pixel i. The set $\varepsilon(i)$ contains the indices of the pixels within an $M_s \times M_s$ search window centered about pixel i. The variable $w_k(i,j)$ is the weight applied to pixel j, when estimating pixel i in frame k. To normalize the weights, the variable $W_{k,i}$ is used, and this is simply the sum of the individual weights.

The SNLM weights are computed based on patch similarity and spatial proximity. In particular, $w_k(i,j)$ is computed as

$$w_k(i,j) = \exp\left\{ -\frac{\left\|\mathbf{y}_{k,i} - \mathbf{y}_{k,j}\right\|^2}{2\sigma_y^2} - \frac{d(i,j)^2}{2\sigma_d^2} \right\}, \tag{3}$$

and $W_{k,i}$ can be expressed as

$$W_{k,i} = \sum_{j \in \varepsilon(i)} w_k(i,j). \tag{4}$$

Note that the variable $\mathbf{y}_{k,i}$ is a vector in lexicographical form containing pixels from an $M_p \times M_p$ patch centered about pixel i in frame k from the sequence $\{y_k(\cdot)\}$. The variable $d(i,j)^2$ is the squared spatial distance between pixel i and j. The parameter σ_y^2 is a tuning parameter to control the decay of the exponential weight function with regard to patch similarity, and σ_d^2 is a tuning parameter controlling the decay with regard to spatial proximity between pixels i and j. It can seen from Equation (3) that the weight given to pixel $y_k(j)$ goes down as $\left\|\mathbf{y}_{k,i} - \mathbf{y}_{k,j}\right\|^2$ goes up. The weight also goes down with the spatial distance between pixel i and j. Note that the tuning parameter, σ_y^2, is often set close to the noise variance. Studies of filter parameter selection can be found in [2], [16]–[18].

III. Denoising by Anscombe transformation and the proposed RNLM algorithm

The block diagram of video denoising using the Anscombe transformation and exact unbiased inverse of Anscombe transformation along with our proposed method RNLM for video denoising is shown in Figure I. The block diagram shows the procedure of Poisson-Gaussian denoising. First, the forward Anscombe transformation is applied to stabilize the noise variance and remove the signal-dependency which can be treated as additive Gaussian noise with unitary variance. Second, the proposed method is applied to reduce the noise, treating it as independent and constant variance noise. Third, to reduce the bias error which occurs when the nonlinear forward Anscombe transformation is applied. The exact unbiased inverse of Anscombe transformation is applied to obtain the estimated frame of interest.

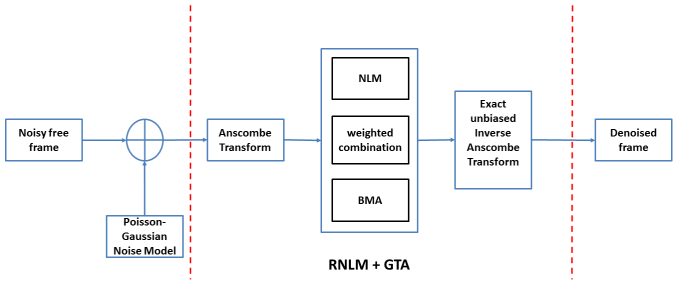
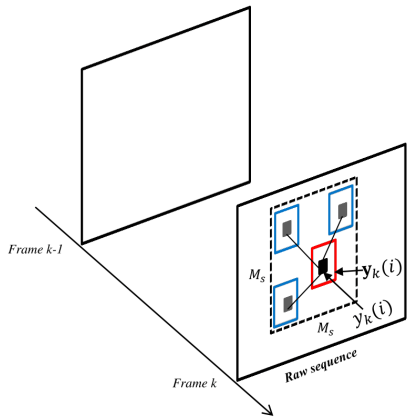


Fig. I. Block Diagram of the Process of denoising using generalized Anscombe transformation combined with Recursive Non-Local Means Filter for Poisson-Gaussian noise.

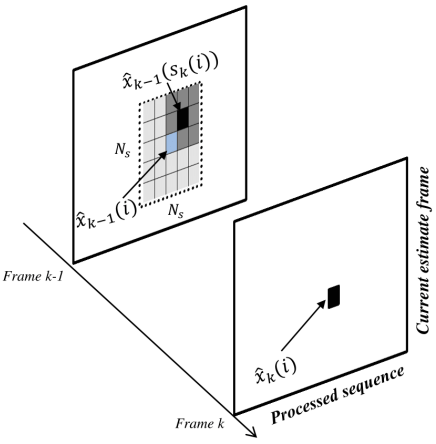
A. RNLM Definition

The goal of the proposed RNLM method is to effectively exploit spatio-tempotal information, but with a computational complexity more in line with the SNLM in Equation (2). To do so, RNLM estimate is formed as a weighted sum of pixels from the current frame, like Equation (2), but it also includes a previous frame pixel estimate. That is, the current input frame and the prior output frame are used to form the current output. This type of recursive processing helps to exploit the temporal correlation, without significantly increasing the search window size or overall computational complexity.

Specifically, the estimate for the proposed RNLM is given



(a)



(b)

Fig. II. Block Diagram of Proposed Algorithm.

by

$$\hat{x}_k(i) = \frac{1}{W_{k,i}} \left[w_{\hat{x},k}(i)\hat{x}_{k-1}(s_k(i)) + \sum_{j \in \varepsilon(i)} w_{y,k}(i,j) y_k(j) \right], \quad (5)$$

where $\hat{x}_{k-1}(s_k(i))$ is the previous frame estimate (i.e., frame $k-1$) at pixel $s_k(i) \in \{1, 2, ..., N\}$. Pixel $s_k(i)$ is selected from $\{\hat{x}_{k-1}(\cdot)\}$ based on block matching with respect to the block in frame k centered about pixel i. For the selection of $s_k(i)$, we allow for a potentially different block and search size from that used for the within-frame processing. In particular, the block matching block size is $N_b \times N_b$, with an $N_s \times N_s$ search window. As in Equation (2) for the SNLM, the set $\varepsilon(i)$ in Equation (5) contains the indices of the pixels within an $M_s \times M_s$ search window centered about pixel i. The recursive weight in Equation (5) is $w_{x,k}(i)$, and the non-recursive weights, $w_{y,k}(i,j)$, are similar to that for the SNLM. We shall define and discuss all of the weights shortly. The relationship among the various pixels used in the RNLM estimation process is depicted in Figure II. Shown are the raw unprocessed frames in parallel with the processed frames. Output $\hat{x}_k(i)$ is formed using a weighted sum of the input frames pixels, shown on the left, and the previous processed output, shown on the back right.

Let us now define the weights. The non-recursive weights are defined in a manner similar to SNLM. Specifically, these are given by

$$w_{y,k}(i,j) = \exp\left\{ -\frac{\|\mathbf{y}_k(i) - \mathbf{y}_k(j)\|^2}{h_{y_b}} - \frac{\sigma_n^2}{h_{y_n}} \right\}, \quad (6)$$

where h_{y_b} and h_{y_n} are tuning parameters. Here, we do not use the spatial distance weighting term of the SNLM. This could easily be added, but it did not provide improved performance in our experimental results. The recursive weights are given by

$$w_{\hat{x},k}(i) = \exp\left\{ -\frac{\|\mathbf{y}_k(i) - \hat{\mathbf{x}}_{k-1}(s_k(i))\|^2}{h_{\hat{x}_b}} - \frac{\sigma^2_{\hat{x}_{k-1}(s_k(i))}}{h_{\hat{x}_n}} \right\}, \quad (7)$$

where $h_{\hat{x}_b}$ and $h_{\hat{x}_n}$ are tuning parameters, and $\sigma^2_{\hat{x}_{k-1}(s_k(i))}$ is the residual noise variance associated with $\hat{x}_{k-1}(s_k(i))$. The vector $\hat{\mathbf{x}}_{k-1}(s_k(i))$ is the $M_p \times M_p$ patch of pixels about pixel $\hat{x}_{k-1}(s_k(i))$ (shown in Figure IIb) in lexicographical vector form. The weight normalization factor here is given by

$$W_{k,i} = w_{\hat{x},k}(i) + \sum_{j \in \varepsilon_y(i)} w_{y,k}(i,j). \quad (8)$$

Finally, assuming the noise is independent and identically distributed, the residual noise variance can be computed recursively as follows:

$$\sigma^2_{\hat{x}_k}(i) = \frac{w^2_{\hat{x},k}(i)\sigma^2_{\hat{x}_{k-1}}(s_k(i)) + \sum_{j \in \varepsilon_y(i)} w^2_{y,k}(i,j)\sigma_n^2}{W^2_{k,i}}. \quad (9)$$

Note that Equation (9) is not the error variance. Rather it only accounts for the variance of the noise component of the error associated with the estimate $\hat{x}_{k-1}(s_k(i))$.

The RNLM weights in Equations (6) and (7) have a total of four tuning parameters, two to govern the non-recursive weights, and two to govern the recursive weights. In the non-recursive weights in Equation (6), the parameter h_{y_b} serves the same role as σ_y^2 in the SNLM in Equation (3). We refer to this weight as the non-recursive bias error weight. We view $||\mathbf{y}_k(i) - \mathbf{y}_k(j)||^2$ as a measure of the bias error in $y_k(j)$ with respect to the true sample $x_k(i)$ that we are estimating. That is, underlying signal differences between the pixel i and j are being quantified by this term. The tuning parameter h_{y_b} controls the exponential decay of the weights as a function of this bias error. The noise error associated with $y_k(j)$ is given by the constant noise variance σ_n^2. The noise variance in Equation (6) is scaled by the h_{y_n} to control the weight decay as a function of the noise variance. For the recursive weight, we have similar tuning parameters. The term $||\mathbf{y}_k(i) - \hat{\mathbf{x}}_{k-1}(s_k(i))||^2$ quantifies the bias error associated with $\hat{x}_{k-1}(s_k(i))$, and the residual noise variance associated with this estimate is $\sigma_{\hat{x}^2_{k-1}(s_k(i))}$, and may be computed using Equation (9). We give each of these error quantities a tuning parameter, to balance their impact on the resulting filter weights. The bias error for the recursive sample is scaled by $h_{\hat{x}_b}$, and the residual noise term is scaled by $h_{\hat{x}_n}$.

IV. Results and Discussion

A set of experiments was performed in order to evaluate the proposed RNLM algorithm. We compare against state-of-art denoising algorithms. The spatio-temporal resolution of each sequence is indicated in Table A. These sequences present variations of condition, for instance, dark and low light conditions, and global and local motion. The original sequences were simulated under different acquisition parameters.

A. Simulated Data

In this section, we provide quantitative quality evaluations of the results that we obtained with the proposed RNLM algorithm. As a measure of quality, the peak signal-to-noise ratio (PSNR) is computed for each experiment and the resulting values are displayed in Table A. PSNR is defined as $PSNR = 10log_{10}(L^2/MSE)$, where L is the the maximum intensity of the original image and MSE is the mean square error between the original image and restored image.

In order to test the proposed method, we have conducted comparisons with two competing algorithms on simulated data. The video sequences we have used in the simulations are publicly available video sequences [19] allowing for reproducible and comparative results. The original video sequences are composed of 50 frames. We added mixed Poisson-Gaussian noise into the original data with a scaled Poisson noise and a zero-mean additive Gaussian noise.

We show in Table A the PSNR results obtained by the proposed method, along with BM3D and SNLM. Table A presents the results of three experiments with different values of the scaling parameter α of the Poisson noise, and the variance σ^2 of the Gaussian noise. Experiment 1 is defined by $\alpha = 0.5$ and $\sigma^2 = [1, 5, 10, 20]$. Experiment 2 by $\alpha = 1$ and $\sigma^2 = [1, 5, 10, 20]$, and Experiment 3 uses $\alpha = 1.5$ and $\sigma^2 = [1, 5, 10, 20]$. The results confirm that our proposed algorithm combined with the generalized Anscombe transformation (RNLM+GAT) shows the highest performances and best noise reduction of the methods tested on our sequences.

In Figure III, we show a visual result of the various algorithms applied on Tennis, Salesman, Miss America, and News sequences. We would suggest that the RNLM+GAT denoised frames exhibit the best subjective visual appearance. The method is not over-smoothing, as we see with some of the other methods. In our results, we have found that it is generally advantageous to choose $N_b > M_p$. This helps to provide a better match for the the important sample $\hat{x}_{k-1}(s_k(i))$. The size of the search window N_s maybe selected based on the temporal motion expected in the video sequence.

V. Conclusion

We have proposed a computationally efficient recursive NLM method (RNLM), and combined it with the Anscombe transformation to allow it to restore video with Poisson-Gaussian noise. Our results show that RNLM is competitive with more computationally complex algorithms, such as BM3D. We believe the proposed method offers a simple and practical video denoising solution, capable of balancing noise reduction and detail preservation. Because of its low computational cost, we believe it is well suited for real-time implementation.

References

[1] Luisier, F., Blu, T., Unser, M.: Image denoising in mixed poisson–gaussian noise. IEEE Transactions on Image Processing **20**(3), 696–708 (2011)

[2] Buades, A., Coll, B., Morel, J.-M.: A review of image denoising algorithms, with a new one. Multiscale Modeling & Simulation **4**(2), 490–530 (2005)

[3] Mahmoudi, M., Sapiro, G.: Fast image and video denoising via nonlocal means of similar neighborhoods. Signal Processing Letters, IEEE **12**(12), 839–842 (2005)

[4] Brox, T., Kleinschmidt, O., Cremers, D.: Efficient nonlocal means for denoising of textural patterns. Image Processing, IEEE Transactions on **17**(7), 1083–1092 (2008)

[5] Coupé, P., Yger, P., Prima, S., Hellier, P., Kervrann, C., Barillot, C.: An optimized blockwise nonlocal means denoising filter for 3-d magnetic resonance images. Medical Imaging, IEEE Transactions on **27**(4), 425–441 (2008)

[6] Dabov, K., Foi, A., Katkovnik, V., Egiazarian, K.: Image denoising with block-matching and 3d filtering. In: Electronic Imaging 2006, pp. 606414–606414 (2006). International Society for Optics and Photonics

[7] Foi, A., Trimeche, M., Katkovnik, V., Egiazarian, K.: Practical poissonian-gaussian noise modeling and fitting for single-image raw-data. IEEE Transactions on Image Processing **17**(10), 1737–1754 (2008)

[8] Makitalo, M., Foi, A.: Optimal inversion of the anscombe transformation in low-count poisson image denoising. IEEE Transactions on Image Processing **20**(1), 99–109 (2011)

[9] Makitalo, M., Foi, A.: Optimal inversion of the generalized anscombe transformation for poisson-gaussian noise. IEEE transactions on image processing **22**(1), 91–103 (2013)

[10] Anscombe, F.J.: The transformation of poisson, binomial and negative-binomial data. Biometrika **35**(3/4), 246–254 (1948)

[11] Mäkitalo, M., Foi, A.: Poisson-gaussian denoising using the exact unbiased inverse of the generalized anscombe transformation. In: 2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 1081–1084 (2012). IEEE

[12] Bindilatti, A.A., Mascarenhas, N.D.: A nonlocal poisson denoising algorithm based on stochastic distances. IEEE Signal Processing Letters **20**(11), 1010–1013 (2013)

[13] Deledalle, C.-A., Tupin, F., Denis, L.: Poisson nl means: Unsupervised non local means for poisson noise. In: 2010 IEEE International Conference on Image Processing, pp. 801–804 (2010). IEEE

[14] He, L., Greenshields, I.R.: A nonlocal maximum likelihood estimation method for rician noise reduction in mr images. IEEE transactions on medical imaging **28**(2), 165–172 (2009)

[15] Salmon, J., Harmany, Z., Deledalle, C.-A., Willett, R.: Poisson noise reduction with non-local pca. Journal of mathematical imaging and vision **48**(2), 279–294 (2014)

[16] Wang, Z., Zhang, L.: An adaptive fast non-local image denoising algorithm. Journal of Image and Graphics **14**(4), 669–675 (2009)

[17] Manjón, J.V., Carbonell-Caballero, J., Lull, J.J., García-Martí, G., Martí-Bonmatí, L., Robles, M.: Mri denoising using non-local means. Medical image analysis **12**(4), 514–523 (2008)

[18] Li, H., Suen, C.Y.: A novel non-local means image denoising method based on grey theory. Pattern Recognition **49**, 237–248 (2016)

[19] Video Trace Library. http://trace.eas.asu.edu/index.html

TABLE A. PSNR COMPARISON WITH COMPETITIVE DENOISING ALGORITHM.

a	s	Video: Alg.	Salesm.				Tennis			
			RNLM(bma)	RNLM(no bma)	BM3D	SNLM	RNLM(bma)	RNLM(no bma)	BM3D	SNLM
0.5	1		38.33	38.22	37.14	35.87	34.50	34.10	33.92	33.09
0.5	5		37.23	37.14	35.76	34.45	33.65	33.20	33.08	32.26
0.5	10		35.36	35.28	33.66	32.24	31.79	31.61	31.60	30.85
0.5	20		32.32	32.23	30.79	29.13	29.76	29.38	29.14	29.11
1	1	PSNR	36.51	36.51	35.27	33.92	32.67	32.23	32.14	31.36
1	5		35.82	35.69	34.51	33.13	32.20	31.77	31.75	30.98
1	10		34.37	34.23	32.98	31.58	31.21	30.79	30.91	30.20
1	20		31.56	31.45	30.59	28.91	29.52	29.15	29.44	28.93
1.5	1		35.50	35.37	34.21	32.81	31.75	31.23	31.20	30.50
1.5	5		35.00	34.87	33.68	32.24	31.46	30.95	30.90	30.27
1.5	10		33.86	33.70	32.55	31.06	30.73	30.16	30.43	29.77
1.5	20		31.42	31.29	30.40	28.70	29.30	28.94	29.28	28.78
a	s	Video: Alg.	Miss America				News			
			RNLM(bma)	RNLM(no bma)	BM3D	SNLM	RNLM(bma)	RNLM(no bma)	BM3D	SNLM
0.5	1		42.01	41.84	41.90	41.05	41.74	41.83	38.32	36.97
0.5	5		41.05	40.89	40.92	40.11	40.33	40.33	35.56	35.72
0.5	10		39.66	39.16	39.71	38.38	37.82	37.90	34.11	33.49
0.5	20		36.55	36.24	37.14	35.42	34.12	34.11	31.72	29.97
1	1	PSNR	40.72	40.52	40.78	39.62	39.62	39.69	36.24	34.85
1	5		40.21	39.98	40.26	39.03	38.34	38.26	35.56	34.16
1	10		39.10	38.84	39.15	37.73	36.61	36.62	34.11	32.60
1	20		36.88	36.61	36.91	35.20	33.47	33.17	31.40	29.65
1.5	1		39.94	39.70	39.95	38.70	37.77	37.84	35.01	33.61
1.5	5		39.55	39.30	39.57	38.25	37.18	37.19	34.55	33.12
1.5	10		38.67	38.38	38.69	37.22	35.82	35.83	33.45	31.93
1.5	20		36.75	36.41	36.71	34.97	33.19	33.05	31.12	29.33

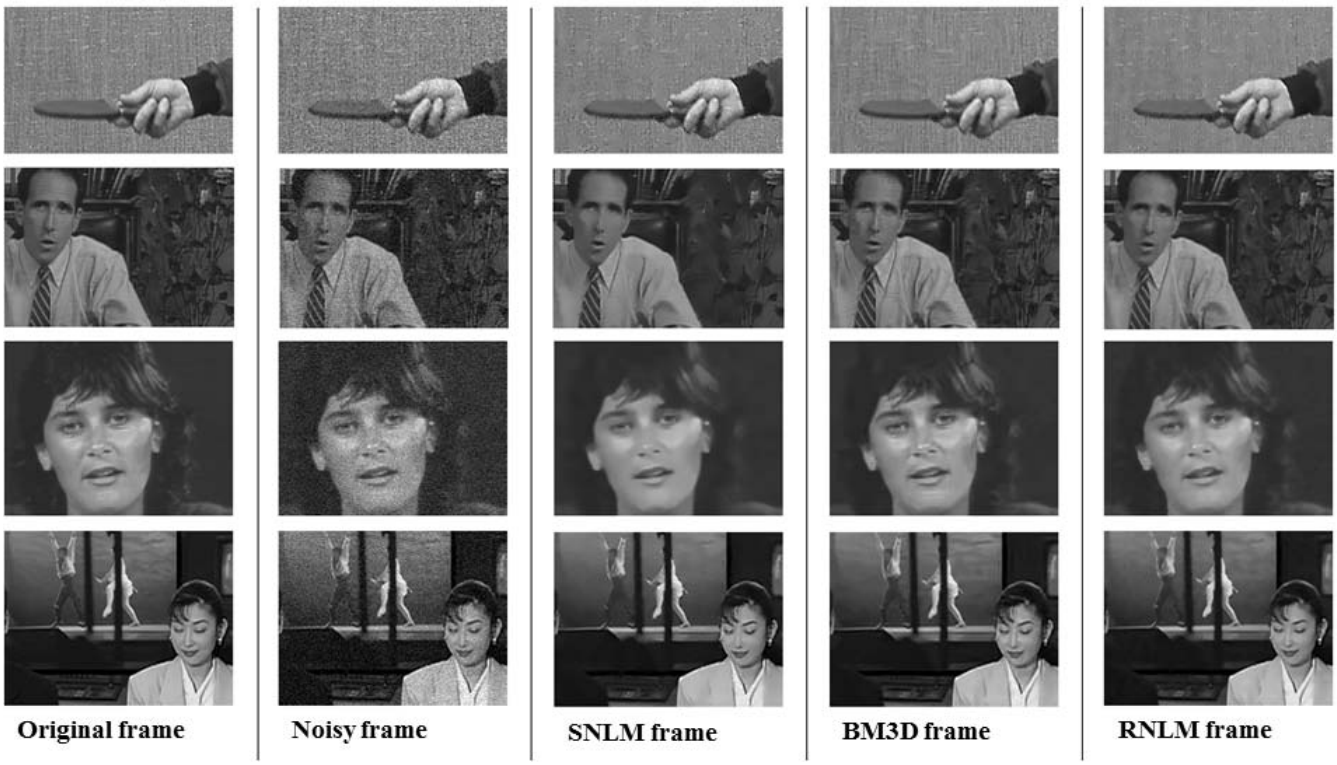


Fig. III. Visual Comparison of denoised Frames from top to bottom, Tennis, Salesman, Miss America, and News corrupted by mixed Poisson-Gaussian noise with $\sigma = 10$ and $\alpha = 0.5$.