SCIENCE CHINA
Information Sciences

• Supplementary File •
Title for citation

Robust Video Denoising with Sparse and Dense Noise Modelings

Guiping SHEN[1,2], Zhi HAN[1*], Xi'ai CHEN[1,2] & Yandong TANG[1]

[1]*State Key Laboratory of Robotics, Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang* 110016*, China;*
[2]*University of Chinese Academy of Sciences, Beijing* 100049*, China*

Appendix A Algorithm details

The proposed algorithm iterates within three steps until reaching convergence. The parameters of sparse part are updated in the first step and the parameters of the mixed Gaussians are updated simultaneously by the following two steps of EM algorithm. The norm constraint items in formula (A1) is solved by Augmented lagrange Multipliers [1]. The intermediate result in this step is denoted by $tempA$, which is a latent low rank matrix. Then the sparse noise $TempS = UV^T - TempA$, in which UV^T is from the iteration result of MoG approximating continuous noise and contains the latent low rank matrix and sparse noise. The following two iterations aim to estimate gaussian model parameters. In the next mixture gaussian model parameter estimation, We put the $P = P - TempS$ as input matrix. In order to get the final latent low rank matrix, U and V are needed to update during iteration.

$$L(Q,S,C,\mu) = ||Q||_* + ||S||_1 + \sum_{n=1}^{N} \pi_n p\left(C|n\right) + \mu \left\|P - Q - S - C\right\|_F^2 \tag{A1}$$

First step, we take the input matrix P for SVD decomposition, then get sparse partial noise S with soft threshold method and obtain matrix $TempQ$ containing continuous noise that equal to $Q + C$, simultaneously. Second step, it assumes a latent variable $z_{r,d,n}$ with $z_{r,d,n} \in (0,1)$ and $\sum\limits_{n=1}^{N} z_{r,d,n} = 1$, implying which specific components of the mixture Gaussian model is most likely better fitting the continuous distribution noise $c_{r,d,n}$. Next, we calculate the maximum expectation of $\log p\left(\left(p_{r,d}, z_{r,d,n}\right)|U,V,\Pi,\Delta\right)$ in regard to the variable $z_{r,d,n}$. The third step, it makes the upper bound of the expectation being the largest, it is formulated as follow:

$$E_z\left(p\left(p_{r,d}, z_{r,d,n}\left|U,V,\Pi,\Delta\right.\right)\right) = \sum_{r,d\in\Omega}\sum_{n=1}^{N} q_{r,d,n}\left(\log\pi_n - \log\sqrt{2\pi}\sigma_n \;\; - \frac{\left(p_{r,d} - u^r\left(v^d\right)^T - s_{r,d}\right)^2}{2\pi\sigma_n^2}\right) \tag{A2}$$

Eq. (A3) calculates the posterior responsibility of mixture gaussian $q_{r,d,n}$ which referred in (A2).

$$q_{r,d,n} = p(z_{r,d,n}\left|p_{r,d}; U,V,\Pi,\Delta\right.) = \frac{\pi_n \mathrm{N}\left(p_{r,d}\middle|u^r\left(v^d\right)^T + s_{r,d}, \sigma_n^2\right)}{\sum\limits_{n=1}^{N}\pi_n \mathrm{N}\left(p_{r,d}\middle|u^r\left(v^d\right)^T + s_{r,d}, \sigma_n^2\right)} \tag{A3}$$

This step is designed to iterative update parameter values to maximize likelihood function. The parameters U, V, Π, Δ will be alternatively updated in a closed form as shown in the follows.

$$\pi_n = \frac{S_n^q}{D_n^q} \tag{A4}$$

$$\sigma_n^2 = \frac{1}{S_n^q}\sum_{r,d\in\Omega}\left[S_n^q \frac{\left(p_{r,d} - u^r\left(v^d\right)^T - s_{r,d}\right)^2}{2\pi\sigma_n^2}\right] \tag{A5}$$

The dimension of $\{q_{r,d,n}\}$ is D_n^q and $S_n^q = \sum\limits_{r,d\in\Omega} q_{r,d,n}$. During the process of iteration, the total number of Gaussian components(denoted by$N$) needs to be adjusted according to the variance of the gaussian distribution. The i^{th} and j^{th} Gaussian components need to be combined into a unique Gaussian distribution when it faces with the condition $\frac{\sigma_i^2 - \sigma_j^2}{\sigma_i^2 + \sigma_j^2} \leqslant \varepsilon$,

* Corresponding author (email: hanzhi@sia..com)

let n_i signify the element number of the i^{th} Gaussian component, then, we change the parameters of mixture Gaussian by setting $\pi_i = \pi_i + \pi_j$, $\sigma_i^2 = \frac{n_i\sigma_i^2 + n_j\sigma_j^2}{n_i + n_j}$, at the same time, it removes the the j^{th} Gaussian component and all of its parameters. Finally, make the number N minus one automatically. The low rank matrix components U and V is formulated as

$$\begin{aligned}&\sum_{r,d\in\Omega}\sum_{n=1}^{N} q_{r,d,n}\left(-\frac{\left(p_{r,d}-u^r\left(v^d\right)^T-s_{r,d}\right)^2}{2\pi\sigma_n^2}\right)\\ &=-\sum_{r,d\in\Omega}\left(\sum_{n=1}^{N}\frac{q_{r,d,n}}{2\pi\sigma_n^2}\right)\left(p_{r,d}\ -u^r\left(v^d\right)^T-s_{r,d}\right)^2\\ &=-\left\|W\odot\left(P_{j,k}-UV^T-S\right)\right\|_{L_2}^2\end{aligned} \tag{A6}$$

where W is a weighting indicator matrix.

Several existing algorithms can be adopted to optimize (A6), like Augmented Lagrange algorithm (ALS). The optimization process of the algorithm is described in Algorithm A1.

Algorithm A1

```
Input: $P_{j,k} = \left(p_{1,j,k}, p_{2,j,k}, \cdots, p_{m,j,k}\right) \in R^{r\times d}$;
Output: $U, V$, $Q_{j,k} = UV^T$, $S$, $C = P - Q - S$;
 1: Initialization: $U, V, \Pi, \Delta, S$, the mixture gaussian number $N$, the small iteration threshold $\varepsilon$;
 2: repeat:
 3: use ALM algorithm to obtain $tempA$; $TempS = UV^T - TempA$;
 4: (E step):Evaluate $\{q_{i,j,n}\}$ for i=1,2,...,r; j=1,2,...,d; n=1,2,...N by (A3);
 5: (M step):Evaluate $\Pi, \Delta$ by (A4) and (A5), respectively;
 6: Evaluate $U, V$ by minimize (A6);
 7: until converged
 8: Output $Q_{j,k}$ $Q_{j,k} = UV^T$
```

Appendix B More experiments

Appendix B.1 Video denoising on various noise sets

We conduct three groups of experiments by adding different sets of noises.

1) The video data is corrupted by only continuous noise: two Gaussian noises ($\delta_1 = 20$, $\delta_2 = 30$) and Poisson noise($p = 25$). The qualitative and quantitative results are shown in Figure B1 and Table B1, respectively. In this experiment, MoG performs best because it is mainly designed for dealing with continuous noise. Our method also performs well but after MoG. Besides the MoG part, our model also has the L_1 norm part for sparse noise, which may lower the performance in such situation.

2) The video data is corrupted by only sparse noise: salt & pepper (10%). From the results shown in Figure B2, neither cvBM3D nor MoG is robust to sparse noise, and LRMC can not well preserve the detail of the images. Table B2 shows the quantitative results of PSNR, from which, we can see our method performs best in this set of experiments.

3) The video data is corrupted by mixed noises with Gaussian noise ($\delta_1 = 20$), Poisson noise ($p = 10$) and salt & pepper (10%). Figure B4 gives the qualitative results, in which, the comparisons of some details are also provided in a zoomed-in show (in red box). As shown, our method gives the best visual effect on both the global view and the details. Also in the comparison on PSNR in Table B3, it is illustrated that our method performs best in all the six clips of videos.

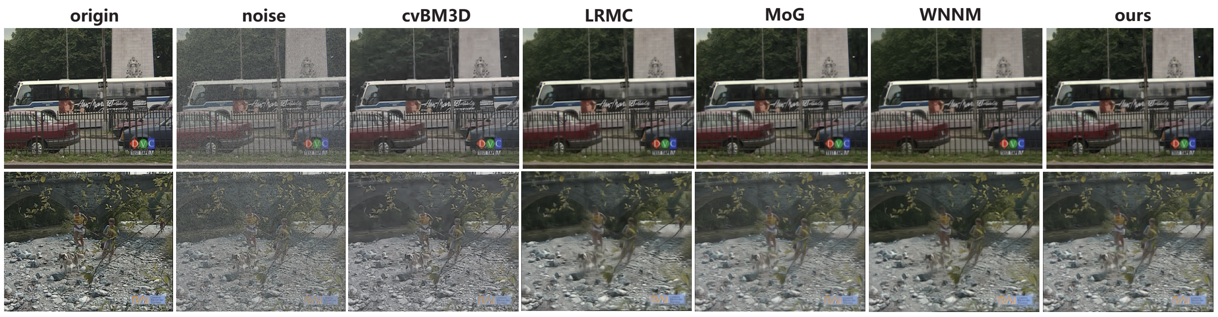

Figure B1 Experiments on videos with continuous noise: two Gaussian noise variance($\delta_1 = 20$, $\delta_2 = 30$); Poisson noise parameter($p = 25$).

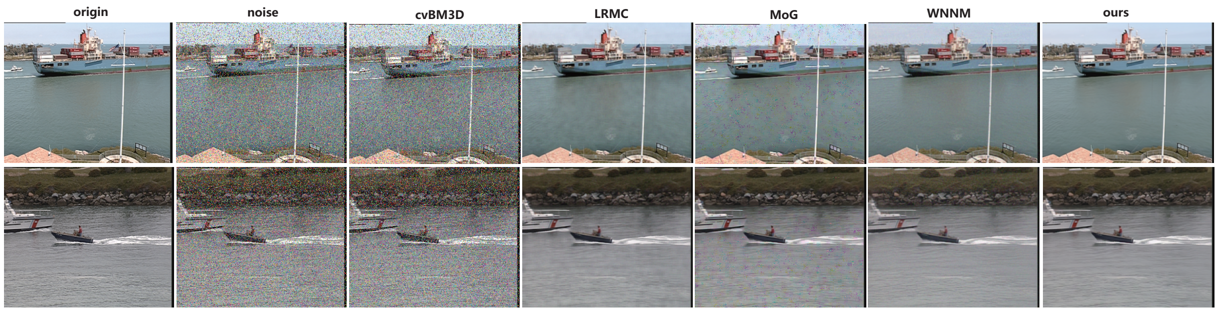

Figure B2 Experiments on videos with sparse noise: salt and pepper(20%).

Table B1 The PSNR of experiments on videos with continuous noise

experiments	cvBM3D	LRMC	MoG	WNNM	ours
1	20.2534	23.1455	**24.3606**	21.9008	23.9501
2	20.0587	20.2334	**23.9782**	20.1454	23.1128

Table B2 The PSNR of experiments on videos with sparse noise

experiments	cvBM3D	LRMC	MoG	WNNM	ours
1	13.0371	28.2307	17.1410	25.6431	**30.2928**
2	13.0900	27.7906	13.6710	26.4360	**29.0930**

Table B3 The PSNR of experiments on videos with mixture noise

experiments	cvBM3D	LRMC	MoG	WNNM	ours
1	19.8413	25.4806	21.3826	24.1212	**25.5223**
2	12.5072	21.4574	14.5898	20.4538	**22.3574**
3	17.1914	26.0974	20.7647	24.0839	**30.4888**
4	19.0221	19.9309	19.4526	19.6430	**20.2429**
5	16.1576	20.4733	18.5426	20.3272	**23.2940**
6	20.3652	23.1670	20.9946	21.0780	**24.3736**

Appendix B.2 The choice of W and L and the influence on the performance

W and L mentioned in the paper are the size of patch match block. For investigating the influence of the block size change on the final results, we carry out the series of experiments on various values of W and L. Figure B3 shows the comparison result.

As the block size changes from small to large, more and more details get lost. Specifically, when the block size is (20,20), the block-artifact phenomenon is severe. The main reason is that when the block is too large, the low-rank property is not well guaranteed. For the smaller size cases, e.g. (4,4), although more details are preserved, it also generates some artificial noises on the overlapping area of blocks. And also, it costs more computing time as sacrifice. It can be seen form Figure B3 that size (8,8) can achieve most robust result. It is effective for most of cases. For all the rest experiments in this paper, W and L are chosen 8×8.

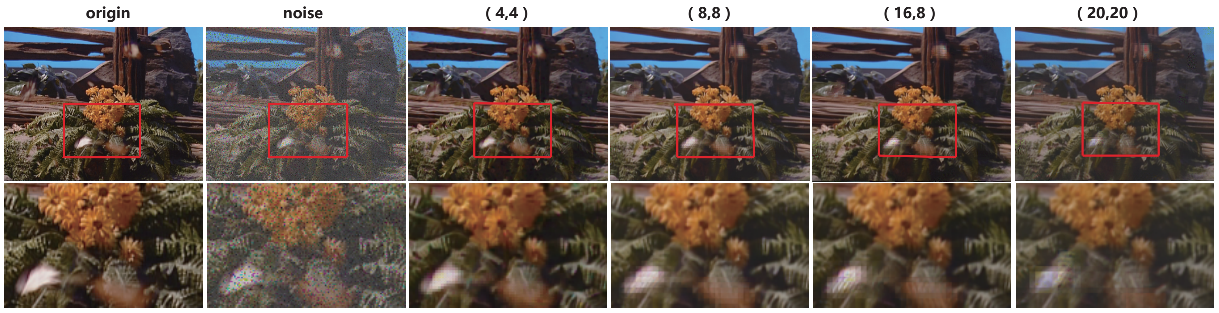

Figure B3 Experiments on various values of (W, L).

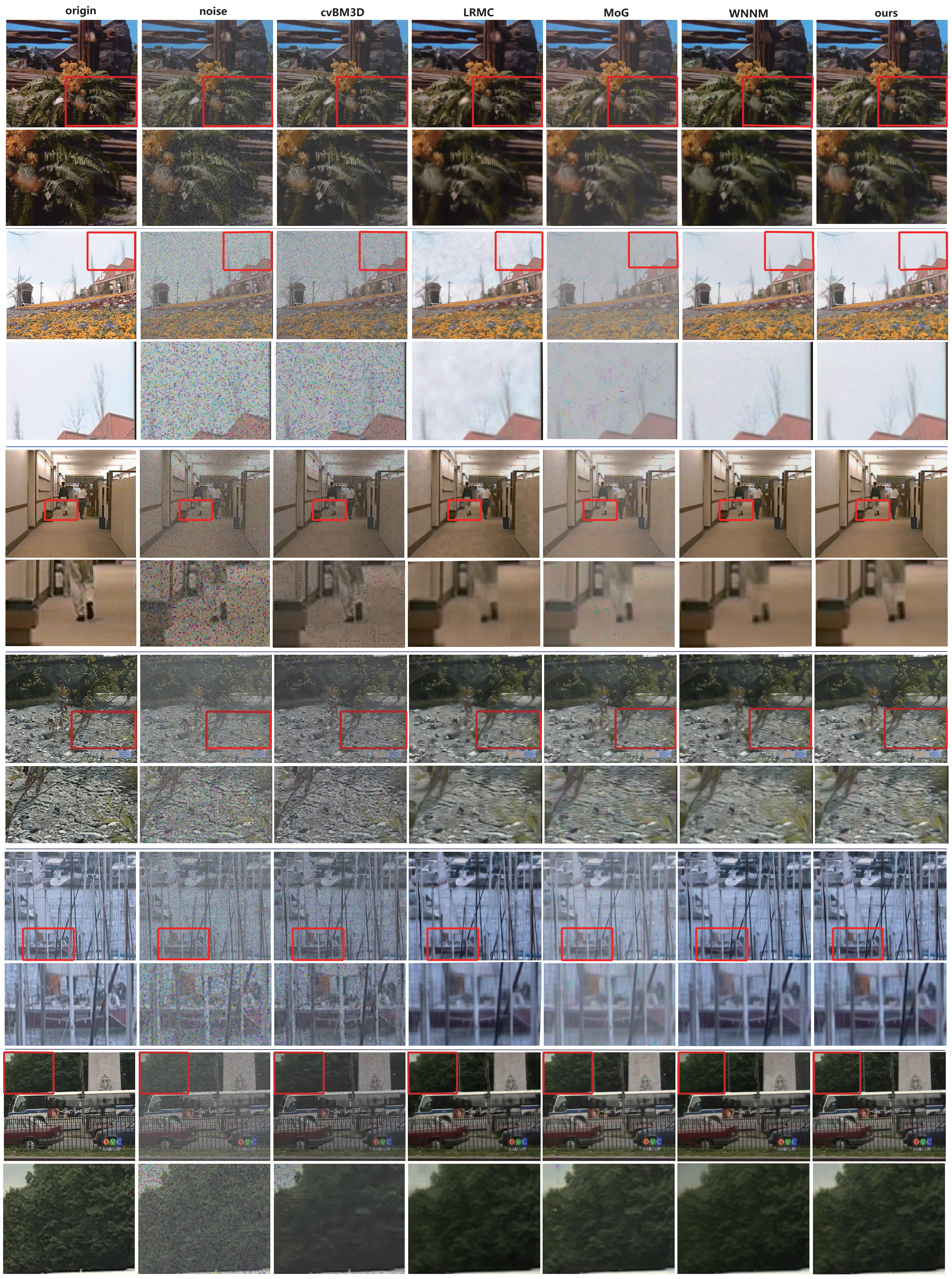

Figure B4 Experiments on videos with mixed noise: Gaussian noise variance ($\delta_1 = 20$), Poisson noise parameter ($p = 10$), pepper and salt(10%). In each group, the upper line and the lower line correspond to global and detailed (zoomed-in) results, respectively.

References

1 Lin Z, Chen M, Ma Y. The Augmented Lagrange Multiplier Method for Exact Recovery of Corrupted Low-Rank Matrices. Eprint Arxiv, 2010, vol: 1-9