Unidirectional Video Denoising by Mimicking Backward Recurrent Modules with Look-ahead Forward Ones

Supplementary Material

Junyi Li[1], Xiaohe Wu[1], Zhenxin Niu[2], and Wangmeng Zuo[1]

[1] Harbin Institute of Technology
[2] Xidian University
{nagejacob, csxhwu, zhenxingniu}@gmail.com
wmzuo@hit.edu.cn

A Quantitative results of BasicVSR [2] and BasicVSR++ [3]

In addition to our implemented BiRNN, BasicVSR [2] and BasicVSR++ [3] are two popular BiRNN methods for video restoration. BasicVSR [2] is introduced as a strong baseline with essential components for video super-resolution. BasicVSR++ [3] further improves BasicVSR [2] in propagation and alignment, and generalizes to compressed video enhancement [14]. However, their performance on video denoising is not well investigated. In this section, we conduct experiments to validate these two methods on video denoising task.

Table A: Quantitative comparison (PSNR/SSIM) of BasicVSR [2] and BasicVSR++ [3] for video denoising task on Set8 dataset [11].

Set8	BasicVSR [2]		BasicVSR++ [3]	
downsample/upsample	✔	✗	✔	✗
$\sigma = 10$	35.88/.9453	37.78/.9635	36.66/.9548	37.92/.9643
$\sigma = 20$	32.80/.9046	34.92/.9386	33.85/.9256	35.18/9408
$\sigma = 30$	31.02/.8694	33.24/.9159	32.24/.8996	33.56/.9195
$\sigma = 40$	29.78/.8379	32.03/.8945	31.10/.8759	32.40/.8996
$\sigma = 50$	28.83/.8091	31.07/.8738	30.20/.8539	31.48/.8808
avg	31.66/.8733	33.81/.9173	32.81/.9020	34.11/.9210
Time(s)	0.06	0.65	0.08	1.84

BasicVSR [2] and BasicVSR++ [3] are originally suggested for video super-resolution task, so they use pixel-shuffle [10] upsample layers at the end of the network to increase the spatial resolution. But for video denoising task, the upsample layers are no longer needed. There are two solutions to fit VSR networks to video denoising task: introducing additional downsample layers at the beginning of the network [4], or just removing the upsample layers. The former

Table B: Quantitative comparison of PSNR with BasicVSR++ [3] on Set8 dataset [11]. We apply flow-guided deformable alignment and second order propagation to FloRNN, named FloRNN++.

	BiRNN	(A)	(B)	(C)	BasicVSR++ [3]	FloRNN	FloRNN++
Flow-Guided Deform. Align		✔			✔		✔
Second-Order Propagation			✔		✔		✔
Grid Propagation				✔	✔		
PSNR	33.74	33.86	33.79	33.84	34.11	33.55	33.72

solution allows the network computing on low resolution video features, which seems as a more efficient choice. But we found the downsample and upsample layers are very harmful to video denoising performance.

We use the official code [1] to train the BasicVSR and BasicVSR++ on video denoising task with two variants: one add downsample layers at the beginning of the network [4], the other remove the upsample layers at the end of the network. We set the training video length to 10 instead of 30 [4] due to GPU memory limit, so the results of downsample/upsample version BasicVSR++ is slightly lower (∼0.2dB) than reported in [4]. From table A, we found that downsample/upsample layers affect the denoising performance. Although 4× downsample the input video and restoring in the low resolution feature space can reduce the computing complexity and speed up the inference, the performance drop can be up to 1∼2 dB. This indicates that getting rid of downsample/upsample layers is essential for best performing video denoising networks.

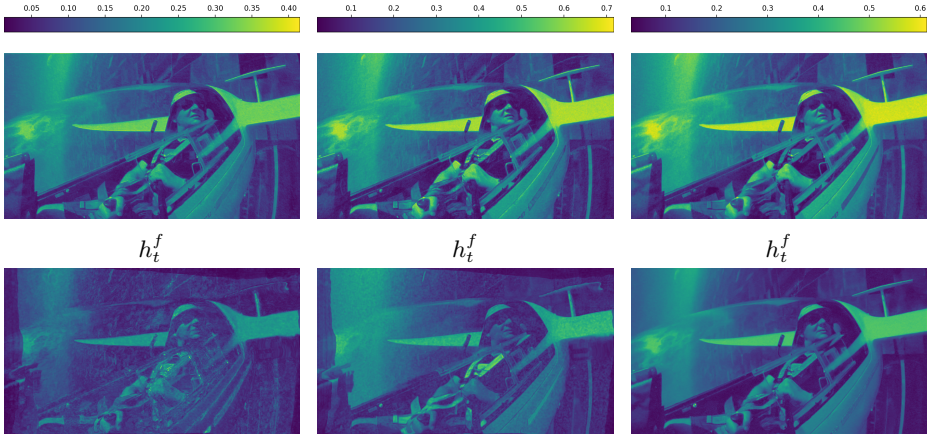
B Extend FloRNN to BasicVSR++ [3]

In this subsection, we show FloRNN can benefit from improvements of state-of-the-art recurrent methods. We extend FloRNN to a state-of-the-art BiRNN method, *i.e.*, BasicVSR++ [3]. BasicVSR++ [3] improves BasicVSR with three modules, *i.e.*, second-order propagation, grid propagation and flow-guided deformable alignment. Due to grid propagation performs bidirectional propagation twice, it can not be equipped to FloRNN. So we only apply second-order propagation and flow-guided deformable alignment to our FloRNN, named as FloRNN++. From table B, with the aid of two improvements, FloRNN++ outperforms 0.17dB over FloRNN, which is comparable with our implemented BiRNN. This demonstrates FloRNN can keep up with advances in BiRNN methods. Although BasicVSR++ [3] achieves better quantitative results, it suffers the common issue of BiRNNs, *i.e.*, large memory consumption, long latency and can only be performed in an offline manner. In contrast, with the proposed look-ahead module, our FloRNN can address the offline issue and be applied to various real-time applications.

[1] https://github.com/open-mmlab/mmediting

Table C: Ablation study of knowledge distillation on Set8 dataset [11], models with and without knowledge distillation show comparable results, which indicates our F_l is able to mimic the F_b of BiRNN and learn feature complementary to F_f.

Knowledge Distillation	✗	✔
PSNR/SSIM	33.55/0.9153	33.53/0.9061



(a) FloRNN (b) FloRNN+distillation (c) BiRNN

Fig. A: Visual comparision of hidden features. Look-ahead feature with knowledge distillation (b) is more similar to backward feature of BiRNN (c), in comparison to the no distillation counterpart (a). The features are visualized with their L_∞ norm.

C Knowledge Distillation

Analogous to other video denoising networks, our FloRNN can be simply trained from scratch using the reconstruction loss,

$$\mathcal{L}_{rec} = \sum_{t=1}^{T} (\hat{\mathbf{x}}_t - \mathbf{x}_t)^2, \tag{1}$$

where T denotes the number of video frames.

Nonetheless, our FloRNN shares many similarities with BiRNN. They both adopt a forward recurrent module and a decoder. The look-ahead recurrent module in FloRNN is suggested to play a similar role as the backward recurrent module in BiRNN for leveraging information from future frames. In order to show the feasibility of look-ahead recurrent module in mimicking backward recurrent module, we further suggest an alternative training scheme by incorporating

Table D: Quantitative comparison of PSNR/SSIM on the Derf dataset for grayscale Gaussian video denoising, hearinafter, Red and Blue indicate the best and the second best results, respectively.

Derf	VBM4D [8]	VNLB [1]	VNLNet [6]	FloRNN(Ours)
$\sigma = 10$	38.88/.9534	40.57/.9731	40.21/.9732	41.34/.9800
$\sigma = 20$	35.10/.9169	36.81/.9428	36.47/.9414	37.95/.9603
$\sigma = 40$	31.40/.8432	32.95/.8856	32.51/.8752	34.31/.9184
Avg	35.13/.9045	36.66/.9338	36.40/.9299	37.87/.9529

Table E: Quantitative comparison of PSNR on the DAVIS dataset [9] for clipped Gaussian video denoising.

DAVIS	ViDeNN [5]	FastDVDNet [12]	PaCNet [13]	FloRNN(Ours)
$\sigma = 10$	37.13	38.65	40.13	40.13
$\sigma = 30$	32.24	33.59	34.92	35.81
$\sigma = 50$	29.77	31.28	32.15	33.54
Avg	33.05	34.51	35.73	36.49

pre-trained BiRNN and distillation loss. Specifically, we first train a BiRNN with reconstruction loss. Then we substitute the backward recurrent module of BiRNN with our look-ahead recurrent module. And distillation loss is deployed to mimic the backward feature $\mathbf{h}_t^b$ with aligned look-ahead feature $\mathbf{h}_{t+k\to t}^l$,

$$\mathcal{L}_{distill} = \sum_{t=1}^{T} \left| \mathbf{h}_{t+k\to t}^l - \mathbf{h}_t^b \right|. \tag{2}$$

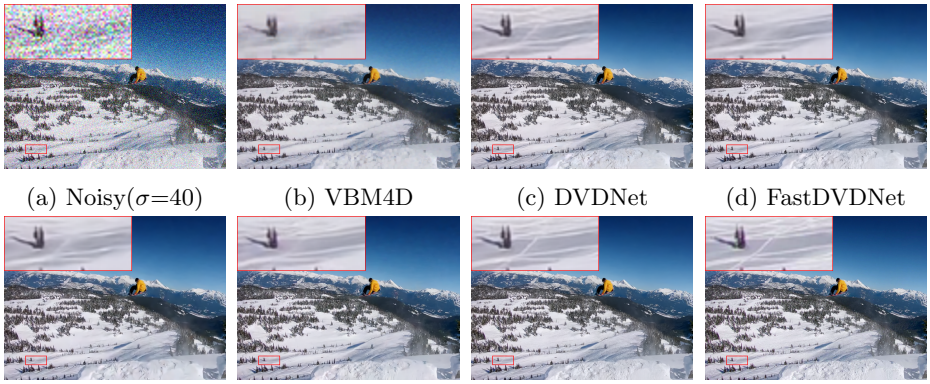
Knowledge distillation encourages the look-ahead recurrent module to learn feature similar to the backward recurrent module in BiRNN. And reconstruction loss is also used to finetune the look-ahead recurrent module and decoder. From Fig. A, the look-ahead feature $h_{t+k\to t}^l$ of the knowledge distillation counterpart is similar to the backward feature h_t^b of BiRNN. As shown in Table C, we empirically find such scheme achieves comparable performance in comparison to training from scratch using $\mathcal{L}_{rec}$. This indicates that look-ahead recurrent module is able to mimic backward recurrent module and learn hidden feature complementary to F_f for video denoising.

D More Experimental Results

We also evaluate FloRNN on grayscale videos and on clipped Gaussian noise. FloRNN shows compelling results in comparison to other methods. As shown in Table D, FloRNN outperforms VNLNet [6] by 1.47dB in average on Derf[2]

[2] https://media.xiph.org/video/derf

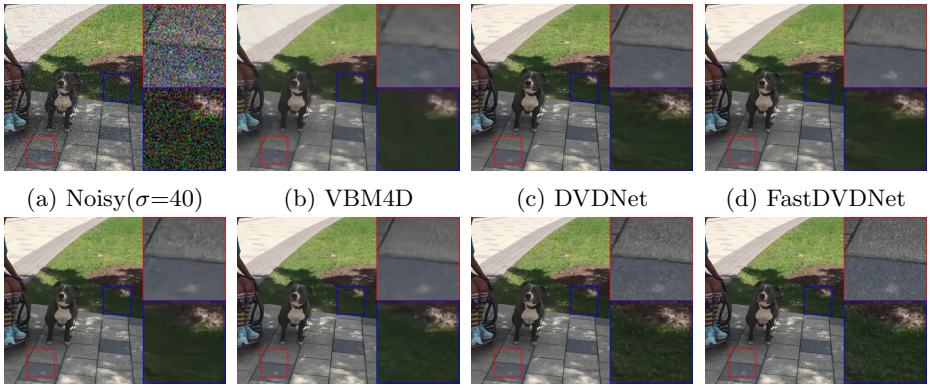
dataset. For clipped Gaussian noise, as shown in Table E, we achieve average PSNR of 0.76dB gain over PaCNet [13] on DAVIS dataset [9]. Figs. B, C, D, E, F show more qualitative results on Set8 [11], DAVIS [9], CRVD [15] and IOCV [7], respectively.



(a) Noisy(σ=40) (b) VBM4D (c) DVDNet (d) FastDVDNet

(e) VNLNet (f) PaCNet (g) FloRNN(Ours) (h) GT

Fig. B: More visual comparison for Gaussian denoising ($\sigma = 40$) on the Set8 dataset [11].



(a) Noisy(σ=40) (b) VBM4D (c) DVDNet (d) FastDVDNet

(e) VNLNet (f) PaCNet (g) FloRNN(Ours) (h) GT

Fig. C: More visual comparison for Gaussian denoising ($\sigma = 40$) on the DAVIS dataset [9].

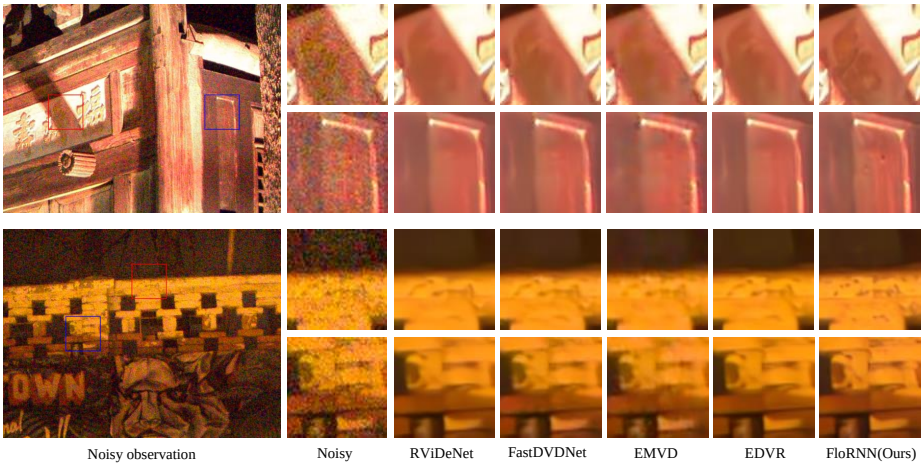
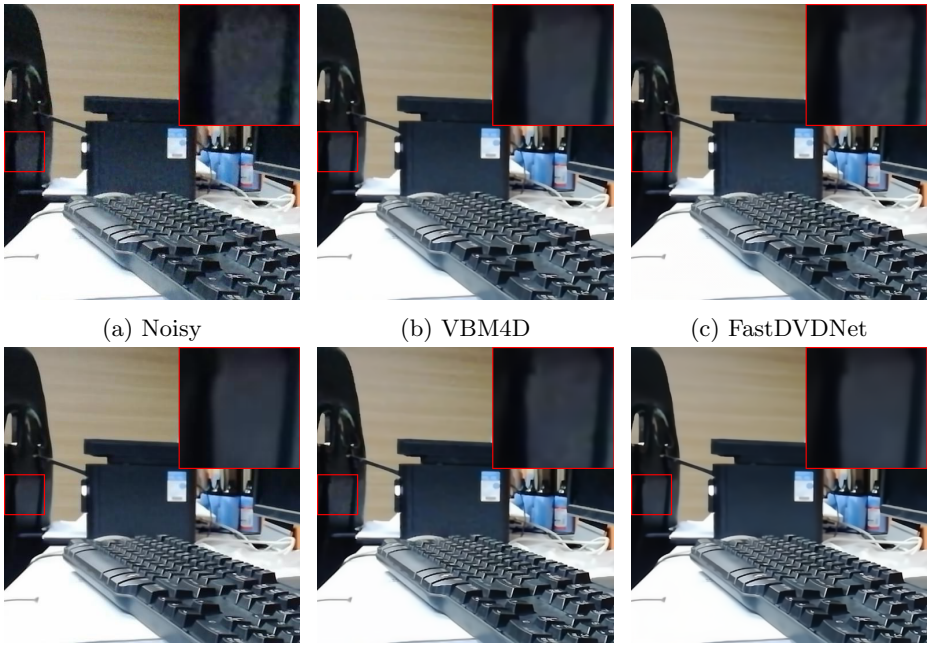


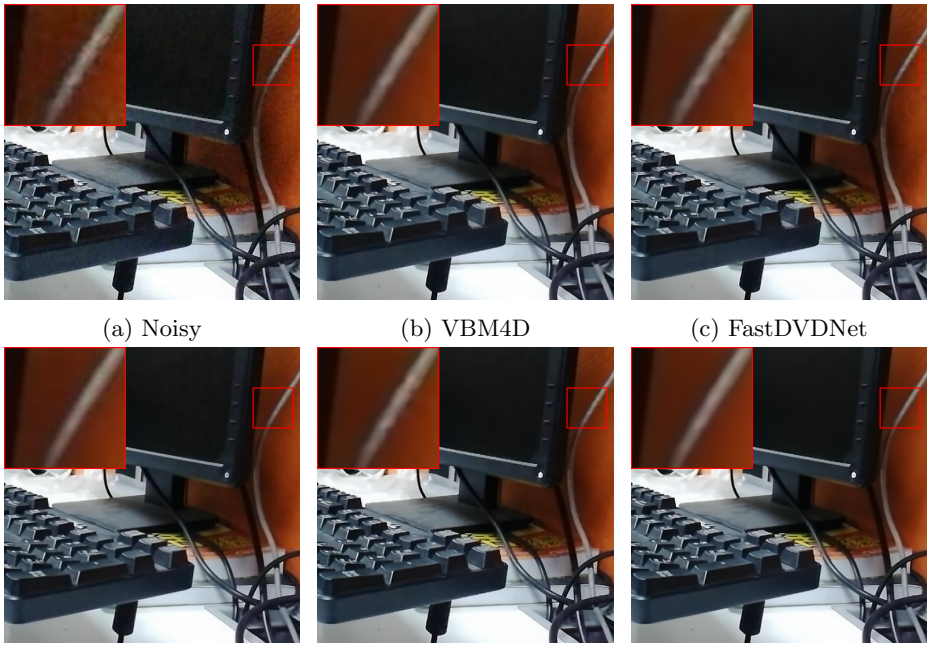
Fig. D: More visual comparison of an outdoor scene on the CRVD dataset [15].



(a) Noisy (b) VBM4D (c) FastDVDNet

(d) GT (e) VNLNet (f) FloRNN(Ours)

Fig. E: More visual comparison on the IOCV dataset [7].



(a) Noisy (b) VBM4D (c) FastDVDNet

(d) GT (e) VNLNet (f) FloRNN(Ours)

Fig. F: More visual comparison on the IOCV dataset [7].

References

1. Arias, P., Morel, J.M.: Video denoising via empirical bayesian estimation of space-time patches. Journal of Mathematical Imaging and Vision **60**(1), 70–93 (2018) 4
2. Chan, K.C., Wang, X., Yu, K., Dong, C., Loy, C.C.: Basicvsr: The search for essential components in video super-resolution and beyond. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4947–4956 (2021) 1
3. Chan, K.C., Zhou, S., Xu, X., Loy, C.C.: Basicvsr++: Improving video super-resolution with enhanced propagation and alignment. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5972–5981 (2022) 1, 2
4. Chan, K.C., Zhou, S., Xu, X., Loy, C.C.: On the generalization of basicvsr++ to video deblurring and denoising. arXiv preprint arXiv:2204.05308 (2022) 1, 2
5. Claus, M., van Gemert, J.: Videnn: Deep blind video denoising. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. pp. 0–0 (2019) 4
6. Davy, A., Ehret, T., Morel, J.M., Arias, P., Facciolo, G.: Video denoising by combining patch search and cnns. Journal of Mathematical Imaging and Vision **63**(1), 73–88 (2021) 4
7. Kong, Z., Yang, X., He, L.: A comprehensive comparison of multi-dimensional image denoising methods. arXiv preprint arXiv:2011.03462 (2020) 5, 7, 8
8. Maggioni, M., Boracchi, G., Foi, A., Egiazarian, K.: Video denoising, deblocking, and enhancement through separable 4-d nonlocal spatiotemporal transforms. IEEE Transactions on image processing **21**(9), 3952–3966 (2012) 4
9. Pont-Tuset, J., Perazzi, F., Caelles, S., Arbeláez, P., Sorkine-Hornung, A., Van Gool, L.: The 2017 davis challenge on video object segmentation. arXiv preprint arXiv:1704.00675 (2017) 4, 5, 6
10. Shi, W., Caballero, J., Huszár, F., Totz, J., Aitken, A.P., Bishop, R., Rueckert, D., Wang, Z.: Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1874–1883 (2016) 1
11. Tassano, M., Delon, J., Veit, T.: Dvdnet: A fast network for deep video denoising. In: 2019 IEEE International Conference on Image Processing (ICIP). pp. 1805–1809. IEEE (2019) 1, 2, 3, 5
12. Tassano, M., Delon, J., Veit, T.: Fastdvdnet: Towards real-time deep video denoising without flow estimation. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2020) 4
13. Vaksman, G., Elad, M., Milanfar, P.: Patch craft: Video denoising by deep modeling and patch matching. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 2157–2166 (October 2021) 4, 5
14. Yang, R.: Ntire 2021 challenge on quality enhancement of compressed video: Methods and results. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops. pp. 647–666 (June 2021) 1
15. Yue, H., Cao, C., Liao, L., Chu, R., Yang, J.: Supervised raw video denoising with a benchmark dataset on dynamic scenes. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2020) 5, 6