Revisiting Temporal Alignment for Video Restoration

Kun Zhou[1,2*] Wenbo Li[3*] Liying Lu[3] Xiaoguang Han[1] Jiangbo Lu[2†]
[1]The Chinese University of Hong Kong (Shenzhen), [2]SmartMore Corporation
[3]The Chinese University of Hong Kong
kunzhou@link.cuhk.edu.cn, {wenboli,lylu}@cse.cuhk.edu.hk
hanxiaoguang@cuhk.edu.cn,jiangbo.lu@gmail.com

1. Residual Block

As illustrated in Table 1, our residual block is comprised of two convolutions, where the first convolution is followed by ReLU activation. In previous works [1, 2, 11], the channel numbers typically keep identical within the residual block. In contrast, we set the channel dimension of hidden representations (M_1) no more than 64 to reduce the parameters of our network and speed up the training and inference phases:

$$M_1 = max(M//2, 64), \tag{1}$$

Input	x
Layer1	Conv(M,M_1,3,1) + ReLU
Layer2	Conv(M_1,M,3,1) $\Rightarrow y$
Output	x+y

Table 1. The structure of our residual block. M refers to the channel number of input.

2. Hyper-parameters in ARW

We examine the impacts of the patch size and α in our ARW module. By default, we use a patch size of 3 to calculate the accuracy-based re-weighting and set the α to -1.0 in consistency-based re-weighting. Also, we explore different settings of the two hyper-parameters. The results are shown in the Table 2. It is observed that a larger patch size leads to a performance drop and different values of α have limited influence on the final quality.

P. Size	3×3	5×5	7×7
PSNR	37.84	37.80	37.72

α	-0.5	-1.0	-2.0
PSNR	37.83	37.84	37.81

Table 2. Ablation study of the hyper-parameters in ARW.

*Equal contribution
†Corresponding author

3. More RBs in Reconstruction Module

To better understand the effect of the residual block (RB) numbers, apart from the default setting with 40 RBs, we train three additional models with 10, 20 and 60 RBs. As shown in the Table 3, the final performance is positively correlated with the number of RBs.

Num. of RBs	10	20	40	60
PSNR (dB)	37.51	37.69	37.84	37.89

Table 3. The influence of different RB numbers.

4. More Results

4.1. Temporal Consistency

We compare the temporal consistency of our method with several state-of-art video SR approaches [2, 4, 6, 11]. The visual results are illustrated in Figure 1. It is observed that other methods fail to restore the consistent textures clearly. While our method empowered with iterative alignment and two adaptively reweighting strategies is able to generate realistic image contents that are closest to the ground-truth.

4.2. Comparison with State-of-the-art

In Table 4 and 5, we give the detialed comparison with several state-of-the-art video SR approaches [1, 1, 2, 11] on REDS4 [8] and Vid4 [7]. The PSNR and SSIM of each video sequence are reported. For most video clips of both two validation sets, our model consistently achieves the best performance. Moreover, we provide extensive qualitative comparison on UDM10 [13], Vid4 [7], Vimeo-90K-T [12], and REDS4 [8] for video SR (in Figure 2, 3), VDB-T [9] for video deblurring (in Figure 4) and Set8 [10], DAVIS [5] for video denoising (in Figure 5). All the qualitative results demonstrate that our method has the capacity to handle various challenging cases in these three video restoration tasks.

Methods	nFrame	Clip_000	Clip_011	Clip_015	Clip_020	Average
Bicubic	1	24.55/0.6489	26.06/0.7261	28.52/0.8034	25.41/0.7386	26.14/0.7292
RCAN [14]	1	26.17/0.7371	29.34/0.8255	31.85/0.8881	27.74/0.8293	28.78/0.8200
TOF [12]	7	26.52/0.7540	27.80/0.7858	30.67/0.8609	26.92/0.7953	27.98/0.7990
DUF [4]	7	27.30/0.7937	28.38/0.8056	31.55/0.8846	27.30/0.8164	28.63/0.8251
EDVR [11]	5	28.01/0.8250	32.17/0.8864	34.06/**0.9206**	30.09/0.8881	31.09/0.8800
MuCAN [6]	5	27.99/0.8219	31.84/0.8801	33.90/0.9170	29.78/0.8811	30.88/0.8750
*BasicVSR [2]	5	27.67/0.8114	31.27/0.8740	33.58/0.9135	29.71/0.8803	30.56/0.8698
*IconVSR [2]	5	27.83/0.8182	31.69/0.8798	33.81/0.9164	29.90/0.8841	30.81/0.8746
VSR-T [1]	5	**28.06/0.8267**	**32.28/0.8883**	**34.15**/0.9199	**30.26/0.8912**	**31.19/0.8815**
Ours	5	**28.16/0.8316**	**32.24/0.8889**	**34.53/0.9275**	**30.26/0.8920**	**31.30/0.8850**

Table 4. Quantitative comparison on REDS4 [8] benchmark under $\times 4$ setting for video super-resolution. Numbers in **red** and **blue** refer to the best and second-best results. All the results are evaluated in the RGB channel. '$*$' indicates the results are from [1].

Clip Name	Bicubic	DUF [4]	EDVR [11]	MuCAN [6]	BasicVSR [2]	IconVSR [2]	VSR-T [1]	Ours
Calendar (Y)	20.39/0.5720	24.04/0.8110	24.05/0.8147	-	-	-	**24.08/0.8125**	**24.65/0.8270**
City (Y)	25.16/0.6028	**28.27/0.8313**	28.00/0.8122	-	-	-	27.94/0.8107	**29.92/0.8428**
Foliage (Y)	23.47/0.5666	**26.41/0.7709**	26.34/0.7635	-	-	-	26.33/0.7635	**26.41/0.7652**
Walk (Y)	26.10/0.7974	30.60/0.9141	31.02/0.9152	-	-	-	**31.10/0.9163**	**31.15/0.9167**
Average (Y)	23.78/0.6347	27.33/**0.8318**	27.35/0.8264	27.26/0.8215	27.24/0.8251	**27.39**/0.8279	27.36/0.8258	**27.90/0.8380**
Average (RGB)	22.37/0.6098	25.79/**0.8136**	**25.83**/0.8077	-	-	-	-	**26.57/0.8235**

Table 5. Quantitative comparison on Vid4 [7] under x4 setting for video super-resolution. We report the PSNR (dB)/SSIM results on both the RGB and the Y channel. Numbers in **red** and **blue** refer to the best and second-best results.

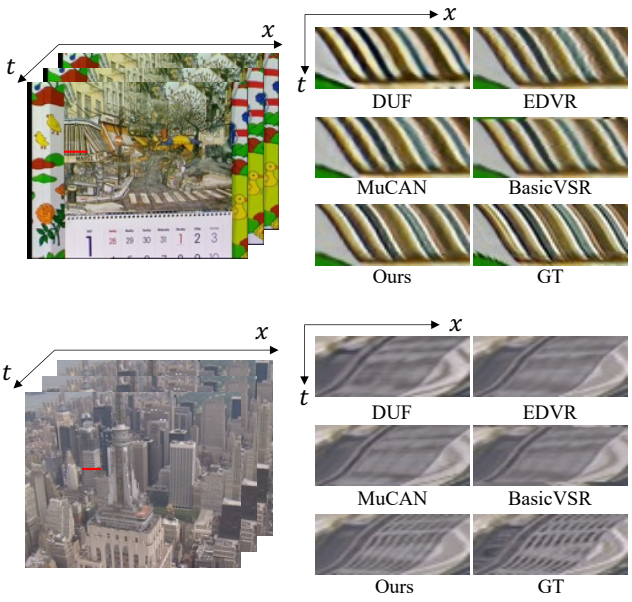

Figure 1. Visualization of temporal consistency on Vid4 [7].

4.3. Comparison with BasicVSR++

In Table 5 of our main text, the PSNR/SSIM values of BasicVSR++ on Vimeo-90K-T are obtained *by pre-training on REDS*. Though our model is only trained on Vimeo-90K without pre-training (as a typical setup), our method still performs better than it (37.79dB→37.84dB). As for REDS4 and Vid4, BasicVSR++ aggregates the information from the full sequence (i.e., 100 frames for REDS4 and 34-49 frames for Vid4) for super-resolving a video frame. In contrast, we adopt the commonly used 5/7-frame settings, like other methods evaluated in Table 4 of our manuscript. In summary, BasicVSR++ actually used extra information and different test setups.

4.4. Video Results

We also provide three videos for visual inspection.
"city.mp4". This video illustrates the visual comparison between bicubic and our method on a Vid4 clip for video super-resolution. It can be observed that our method restores much clear image details (e.g., the finer structure of buildings).
"IMG0030.mp4". This video demonstrates the visual results of our method on a testing sequence of VDB-T [9] for the video deblurring task. The burry input and the generated frames are shown in it.
"motorbike.mp4". This video shows the restoration results on a sequence of Set8 [10] for video denoising.

References

[1] Jiezhang Cao, Yawei Li, Kai Zhang, and Luc Van Gool. Video super-resolution transformer. *arXiv*, 2021. 1, 2

[2] Kelvin C.K. Chan, Xintao Wang, Ke Yu, Chao Dong, and Chen Change Loy. Basicvsr: The search for essential com-

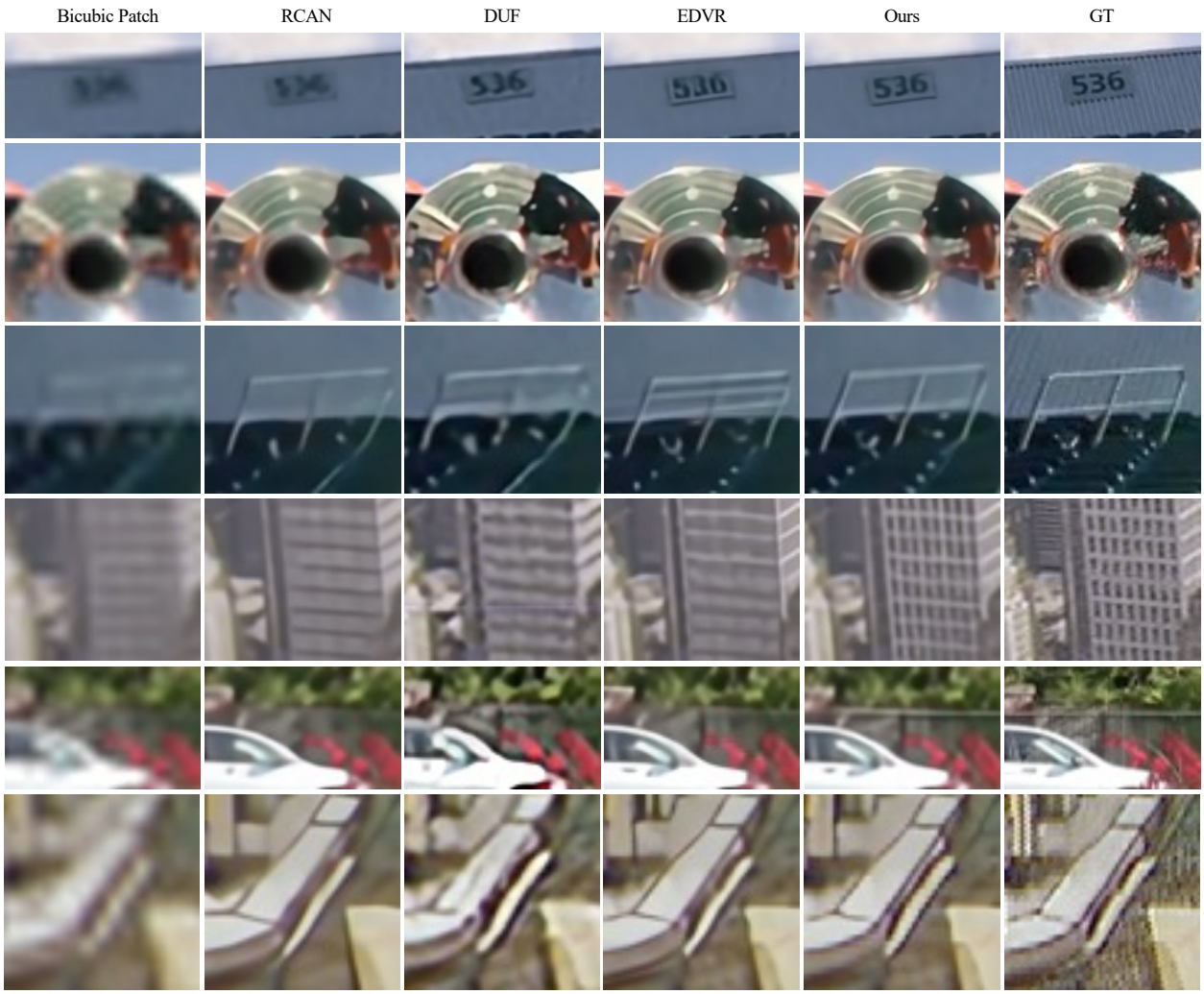

Figure 2. Qualitative comparison on UDM10 [13] and Vid4 [7] for video SR.

ponents in video super-resolution and beyond. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2021. 1, 2

[3] Kelvin CK Chan, Shangchen Zhou, Xiangyu Xu, and Chen Change Loy. Basicvsr++: Improving video super-resolution with enhanced propagation and alignment. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2022.

[4] Younghyun Jo, Seoung Wug Oh, Jaeyeon Kang, and Seon Joo Kim. Deep video super-resolution network using dynamic upsampling filters without explicit motion compensation. In *CVPR*, pages 3224–3232, 2018. 1, 2

[5] Anna Khoreva, Anna Rohrbach, and Bernt Schiele. Video object segmentation with language referring expressions. In *Asian Conference on Computer Vision*, pages 123–141. Springer, 2018. 1, 6

[6] Wenbo Li, Xin Tao, Taian Guo, Lu Qi, Jiangbo Lu, and Jiaya Jia. Mucan: Multi-correspondence aggregation network for video super-resolution. In *ECCV*, pages 335–351. Springer, 2020. 1, 2

[7] Ce Liu and Deqing Sun. On bayesian adaptive video super resolution. *IEEE transactions on pattern analysis and machine intelligence*, 36(2):346–360, 2013. 1, 2, 3

[8] Seungjun Nah, Sungyong Baik, Seokil Hong, Gyeongsik Moon, Sanghyun Son, Radu Timofte, and Kyoung Mu Lee. Ntire 2019 challenge on video deblurring and super-resolution: Dataset and study. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019. 1, 2, 4

[9] Shuochen Su, Mauricio Delbracio, Jue Wang, Guillermo Sapiro, Wolfgang Heidrich, and Oliver Wang. Deep video deblurring for hand-held cameras. In *CVPR*, pages 1279–1288, 2017. 1, 2, 5

[10] Matias Tassano, Julie Delon, and Thomas Veit. Fastdvdnet: Towards real-time deep video denoising without flow estimation. In *CVPR*, pages 1354–1363, 2020. 1, 2, 6

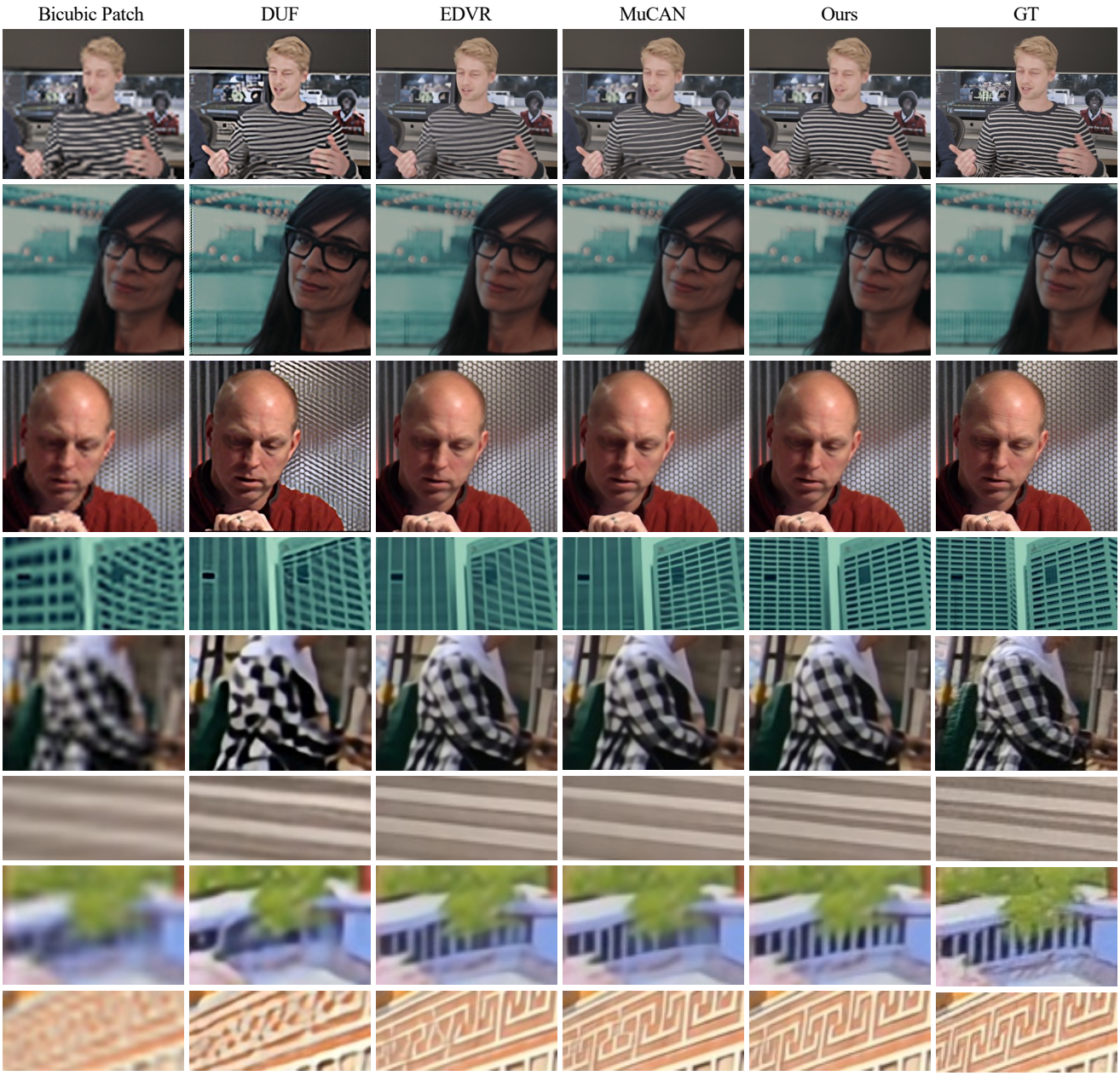

Figure 3. Qualitative comparison on Vimeo-90K-T [12] and REDS4 [8] for video SR.

[11] Xintao Wang, Kelvin CK Chan, Ke Yu, Chao Dong, and Chen Change Loy. Edvr: Video restoration with enhanced deformable convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019. 1, 2

[12] Tianfan Xue, Baian Chen, Jiajun Wu, Donglai Wei, and William T Freeman. Video enhancement with task-oriented flow. *IJCV*, 127(8):1106–1125, 2019. 1, 2, 4

[13] Peng Yi, Zhongyuan Wang, Kui Jiang, Junjun Jiang, and Jiayi Ma. Progressive fusion video super-resolution network via exploiting non-local spatio-temporal correlations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3106–3115, 2019. 1, 3

[14] Yulun Zhang, Kunpeng Li, Kai Li, Lichen Wang, Bineng Zhong, and Yun Fu. Image super-resolution using very deep residual channel attention networks. In *ECCV*, pages 286–301, 2018. 2

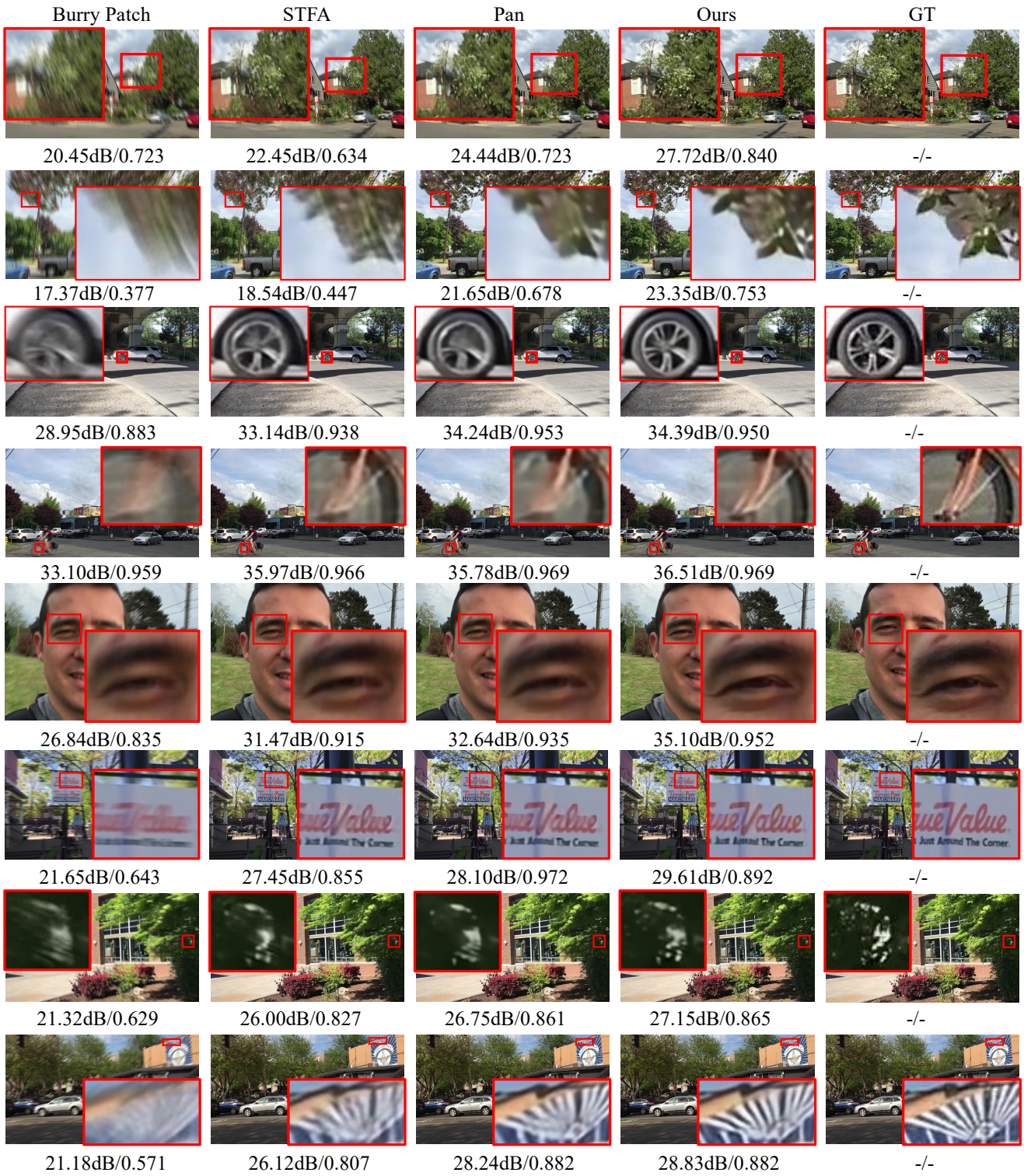

Figure 4. Qualitative comparison on VDB-T [9] for video deblurring.

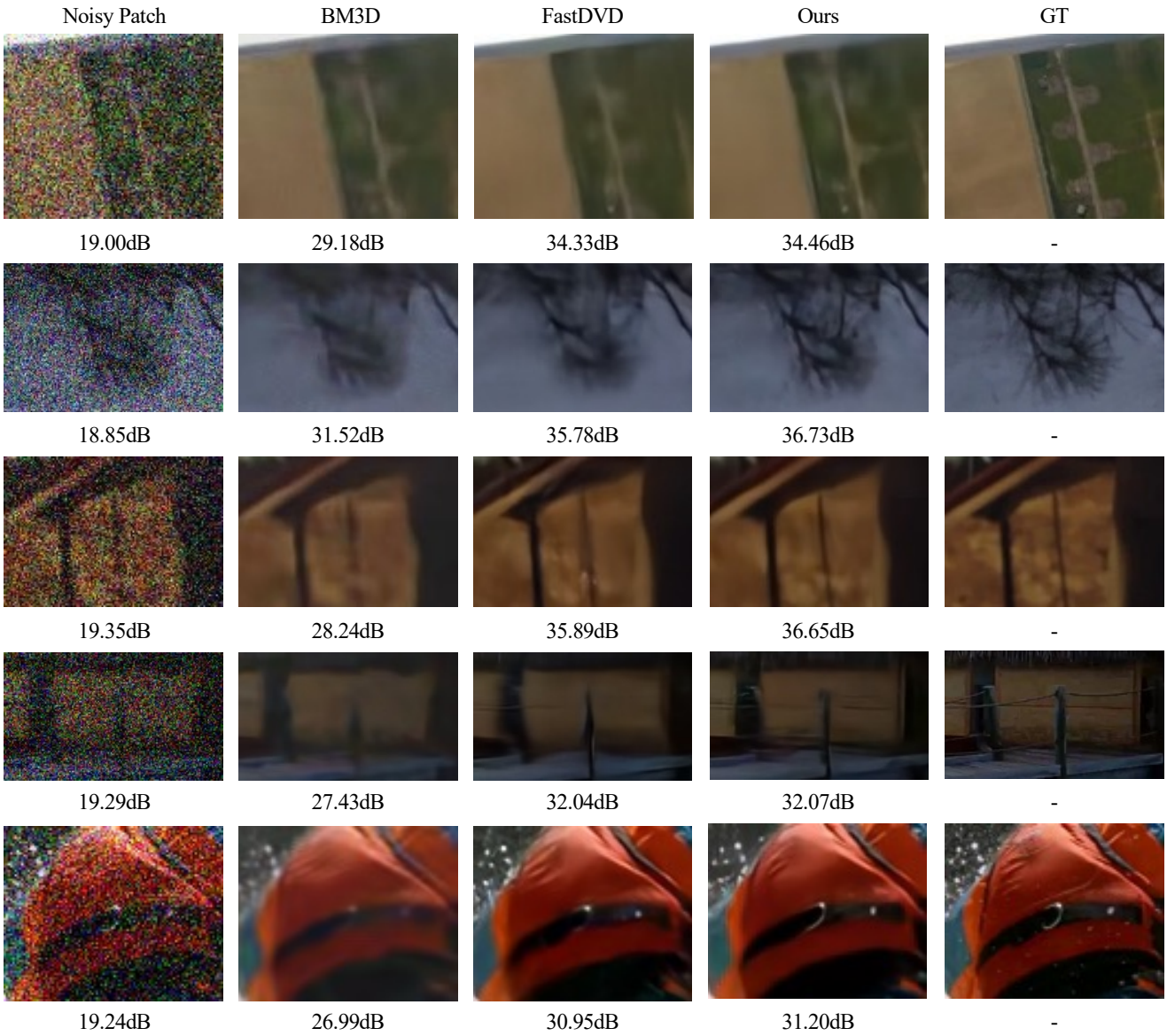

Figure 5. Qualitative comparison on Set8 [10], DAVIS [5] for video denoising. The values beneath images represent the PSNR (dB).