Supplementary Material: High-Pass Kernel Prediction for Efficient Video Deblurring

Bo Ji Angela Yao
National University of Singapore
{jibo,ayao}@comp.nus.edu.sg

1. High-pass filter kernel

The proof for Proposition 1 is presented below.

Proof. Consider $h(x) = \sum_{i=1}^{M} \alpha_i h_i(x)$. By utilizing the linearity of the Fourier transform, it can be expressed in the Fourier domain as:

$$H(f) = \sum_{i=1}^{M} \alpha_i H_i(f) \tag{1}$$

In this domain, $H_i(f)$, representing a high-pass filtering function, is generally observed as a non-decreasing function $g_i(f)$ ranging from 0 to 1. A value of $g_i(f) = 0$ signifies complete attenuation of the frequency component f, while $g_i(f) = 1$ denotes no attenuation.

Given that the sum and scalar multiplication of non-decreasing functions remain non-decreasing, $H(f)$ is also non-decreasing. Therefore, $h(x)$ is identified as a high-pass filter. □

2. The kernels for high-pass filtering

We describe the kernels we used in Section 4.3. The Sobel filter is defined as follows:

$$\text{Sobel}_x = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1, \end{bmatrix}, \text{Sobel}_y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}. \tag{2}$$

The Kirsch filter is:

$$\text{Kirsch}_x = \begin{bmatrix} -3 & -3 & 5 \\ -3 & 0 & 5 \\ -3 & -3 & 5, \end{bmatrix}, \text{Kirsch}_y = \begin{bmatrix} 5 & 5 & 5 \\ -3 & 0 & -3 \\ -3 & -3 & -3 \end{bmatrix}. \tag{3}$$

The Laplacian filter is:

$$\text{Laplacian} = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}. \tag{4}$$

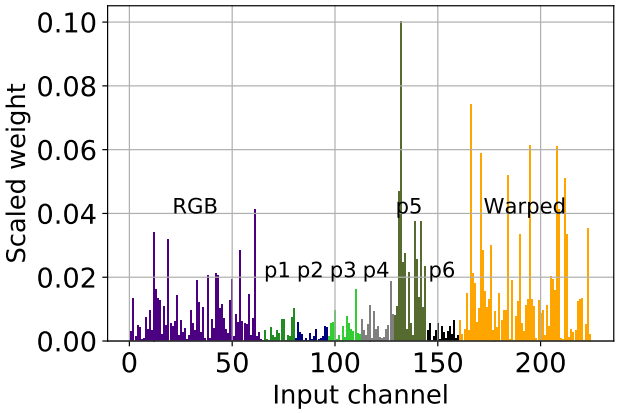

Figure 1. Histogram of the weights in the fusion.

3. Architecture

The detailed architecture of our model is presented in Tables 1 to 4. For the pre-process module $\mathcal{P}$, we utilize residual dense blocks (RDB) [6] as the foundational building blocks. The architectures of $\mathcal{P}^0$ and $\mathcal{P}^n$, where $n = 1, \ldots, N$, differ. Greater computational resources are allocated to P^0 due to its role in the final restoration in RGB format, prioritizing the direct RGB input over other features.

4. Experiments

4.1. Weights in the fusion module

To understand the importance of each input mentioned in Eq.10 of the main paper, we visualize the weights of the first convolution in $\mathcal{R}$, which processes the concatenated inputs from Eq.10. The results are presented in Figure 1. A larger weight indicates that the deblurring effect is more significantly attributed to that particular input component. We use p^i to represent the weight of the i-th representation feature. The model fuses 224 channels, with the first 64 channels representing the RGB input, followed by 6×16 channels for 6 different features, and finally 64 channels for the warped

Layer	Output	Coefficient Generator
conv1	$T \times H/4 \times W/4 \times 3$	$3 \times 3 \times 3$, stride 1
conv2	$T \times H/4 \times W/4 \times 3$	$3 \times 3 \times 3$, stride 1
AvgPool	$1 \times 1 \times 1 \times 3$	-
Linear	$1 \times 1 \times 1 \times 3$	$3 \times NM$

Table 1. Coefficient generator $\mathcal{G}$ architecture.

Layer	Output	Pre-process Module
Conv1	$H \times W \times 3$	5×5, stride 1
RDB1	$H \times W \times 3$	$[3 \times 3, 16$, dense conv$] \times 4$
Conv2	$H/2 \times W/2 \times 32$	5×5, stride 2
RDB2	$H/2 \times W/2 \times 32$	$[3 \times 3, 24$, dense conv$] \times 4$
Conv3	$H/4 \times W/4 \times 64$	5×5, stride 2

Table 2. Pre-process module $\mathcal{P}^0$ architecture.

Layer	Output	Pre-process Module
Conv1	$H \times W \times 3$	5×5, stride 1
RDB1	$H \times W \times 3$	$[3 \times 3, 16$, dense conv$] \times 2$
Conv2	$H/2 \times W/2 \times 16$	5×5, stride 2
RDB2	$H/2 \times W/2 \times 16$	$[3 \times 3, 24$, dense conv$] \times 2$
Conv3	$H/4 \times W/4 \times 16$	5×5, stride 2

Table 3. Pre-process module $\mathcal{P}^n, n = 1, \ldots, N$ architecture.

Layer	Output	Deblurring Module
Conv1	$H/4 \times W/4 \times 64$	3×3, stride 1
ResBlock1	$H/4 \times W/4 \times 64$	$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 30$
Transposed conv1	$H/2 \times W/2 \times 32$	3×3, stride 2
Transposed conv2	$H \times W \times 16$	3×3, stride 2
Conv1	$H \times W \times 3$	5×5, stride 1

Table 4. Deblurring module $\mathcal{D}$ architecture.

output from the previous frame. The histogram values are scaled with respect to the input scale to ensure that the comparison is meaningful. The channel for p^5 represents the spatial gradient (see k^4 in Figure 14). The fusion convolution assigns more weight to the input, the spatial gradient map, and the previous warped results, which further illustrates the importance of the spatial gradient in deblurring.

4.2. Visual comparison

Additional visual comparisons are provided in Figures 2 to 9. Figure 14 provides more visual examples of the generated kernels k_t and the corresponding high-frequency result.

References

[1] Seungjun Nah, Tae Hyun Kim, and Kyoung Mu Lee. Deep multi-scale convolutional neural network for dynamic scene deblurring. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3883–3891, 2017. 3, 4, 5

[2] Jinshan Pan, Haoran Bai, and Jinhui Tang. Cascaded deep video deblurring using temporal sharpness prior. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3043–3051, 2020. 3, 4, 5

[3] Shuochen Su, Mauricio Delbracio, Jue Wang, Guillermo Sapiro, Wolfgang Heidrich, and Oliver Wang. Deep video deblurring for hand-held cameras. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1279–1288, 2017. 4, 5

[4] Yusheng Wang, Yunfan Lu, Ye Gao, Lin Wang, Zhihang Zhong, Yinqiang Zheng, and Atsushi Yamashita. Efficient video deblurring guided by motion magnitude. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XIX*, pages 413–429. Springer, 2022. 3, 4

[5] Huicong Zhang, Haozhe Xie, and Hongxun Yao. Spatio-temporal deformable attention network for video deblurring. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XVI*, pages 581–596. Springer, 2022. 4, 5

[6] Zhihang Zhong, Ye Gao, Yinqiang Zheng, and Bo Zheng. Efficient spatio-temporal recurrent neural network for video deblurring. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VI 16*, pages 191–207. Springer, 2020. 1, 3, 4

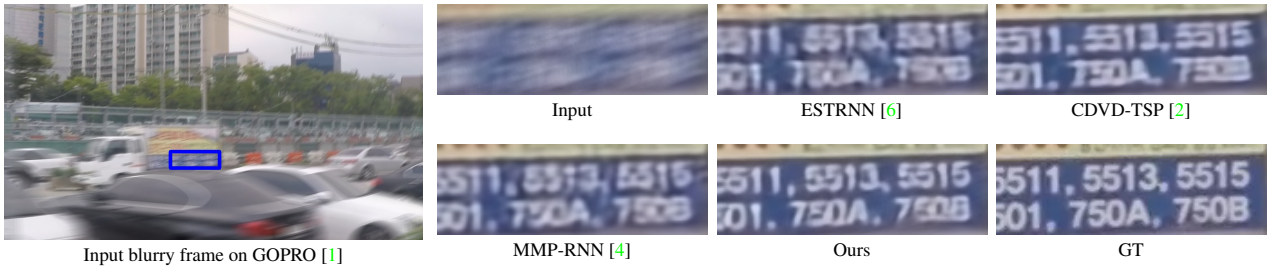

Figure 2. Qualitative comparisons to models with a similar training memory footprint.

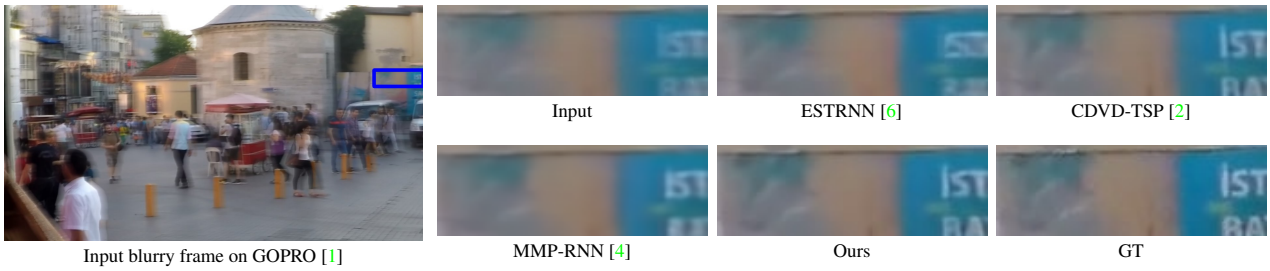

Figure 3. Qualitative comparisons to models with a similar training memory footprint.

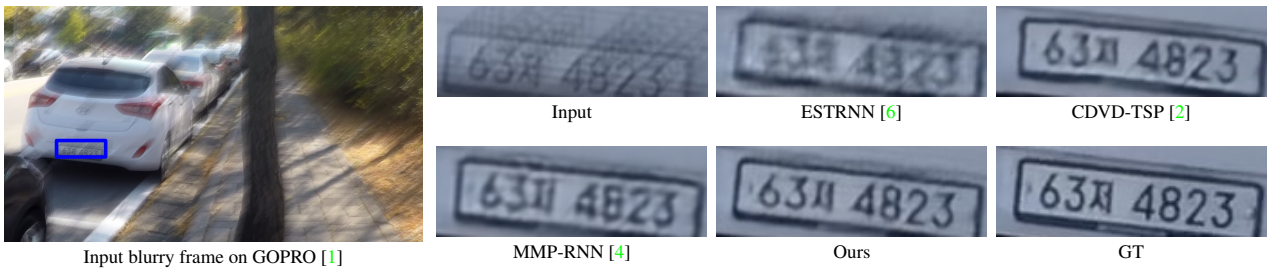

Figure 4. Qualitative comparisons to models with a similar training memory footprint.

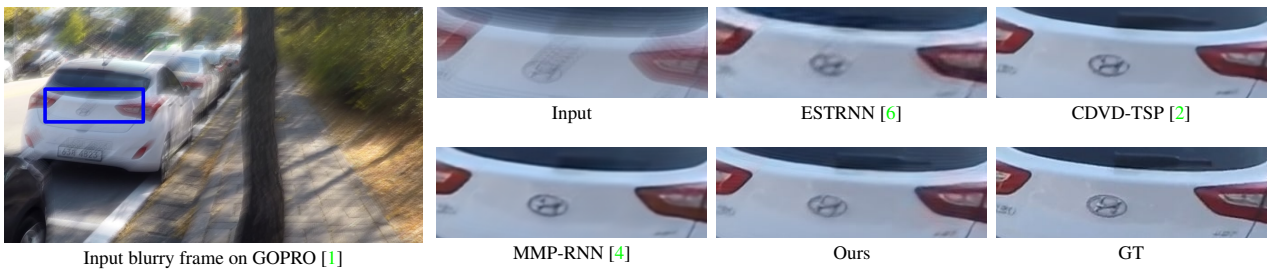

Figure 5. Qualitative comparisons to models with a similar training memory footprint.

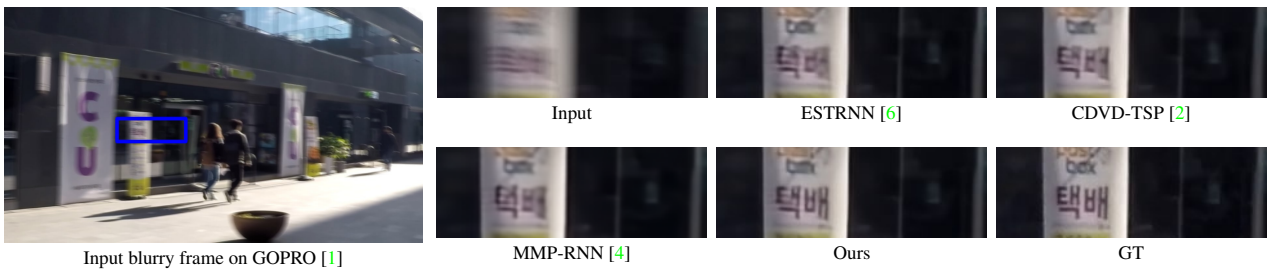

Figure 6. Qualitative comparisons to models with a similar training memory footprint.

Figure 7. Qualitative comparisons to models with a similar training memory footprint.

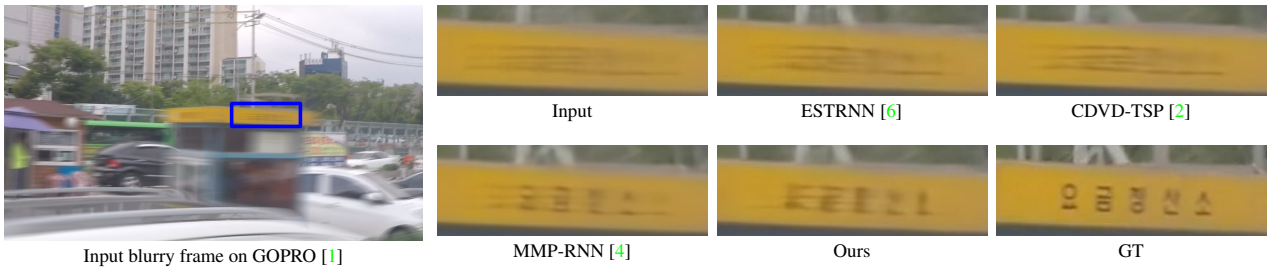

Figure 8. Qualitative comparisons to models with a similar training memory footprint.

Figure 9. Qualitative comparisons to models with a similar training memory footprint.

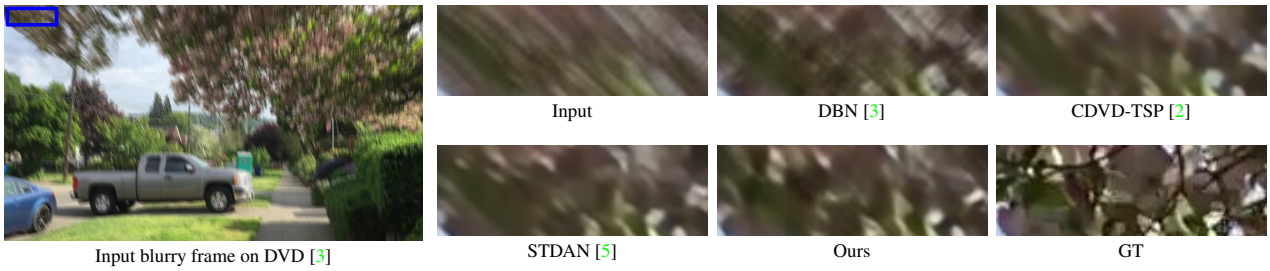

Figure 10. Qualitative comparisons to models with a similar training memory footprint.

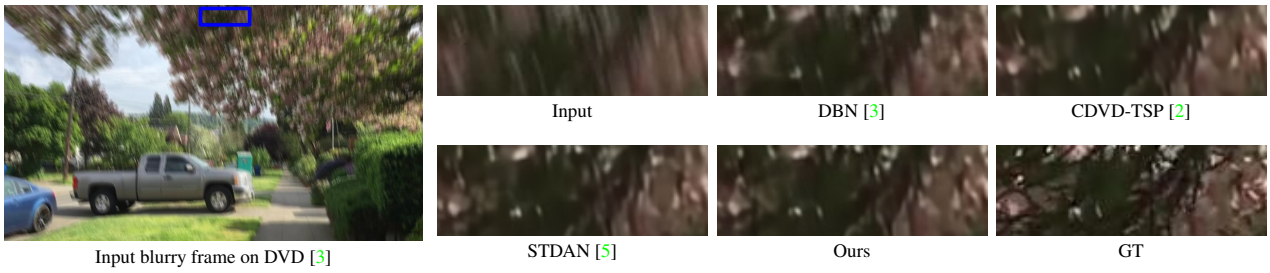

Figure 11. Qualitative comparisons to models with a similar training memory footprint.

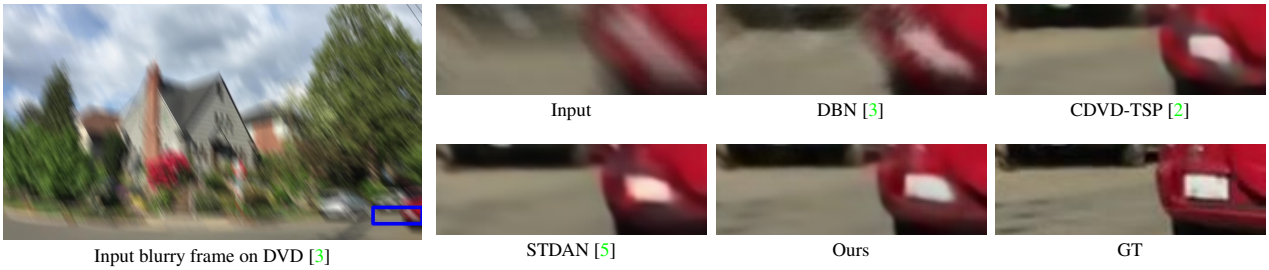

Figure 12. Qualitative comparisons to models with a similar training memory footprint.

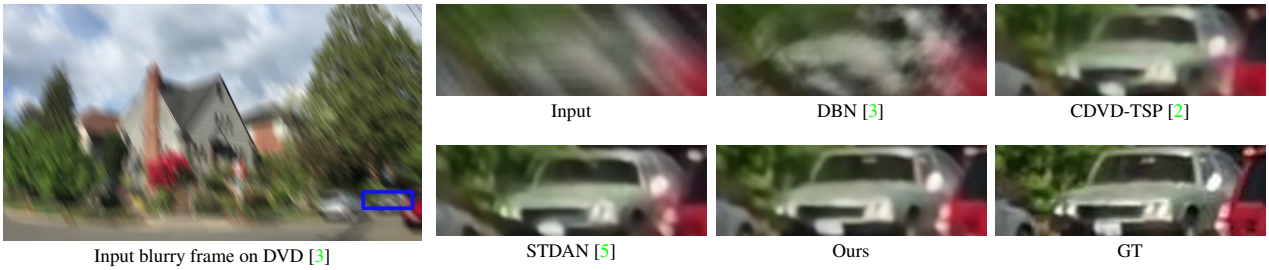

Figure 13. Qualitative comparisons to models with a similar training memory footprint.

Figure 14. Examples of learned kernels and features (please zoom in for a better view).