Efficient Multi-scale Network with Learnable Discrete Wavelet Transform for Blind Motion Deblurring

Xin Gao*[1,2] Tianheng Qiu*[2,3,4]
Xinyu Zhang†[2] Hanlin Bai[1] Kang Liu[1] Xuan Huang†[4] Hu Wei[4] Guoying Zhang†[1] Huaping Liu[2]
*Equal Contribution † Coresponding Author

[1]China University of Mining & Technology-Beijing [2]Tsinghua University
[3]University of Science and Technology of China
[4]Hefei Institutes of Physical Science,Chinese Academy of Sciences

A. Appendix

In this document, we first provide further derivation of the proposed method in Sec. A.1. Then we demonstrate the way to initialize the wavelet and the low memory usage of our method in practical applications in Sec. A.2 and Sec. A.3, respectively. Finally, we show more visual comparisons between the proposed method and state-of-the-art methods in Sec. A.4.

A.1. Further Derivation on the Proposed Method

For the z-transform $X(z)$ of the input x in Sec. **??**, which undergoes a forward wavelet transform to produce high and low frequency components with a downsampling factor of 2 can be expressed as,

$$X_{down,low}\left(z\right) = \frac{1}{2}\left[X(z^{\frac{1}{2}})A_0(z^{\frac{1}{2}}) + X(-z^{\frac{1}{2}})A_0(-z^{\frac{1}{2}})\right] \tag{12}$$

$$X_{down,high}\left(z\right) = \frac{1}{2}\left[X(z^{\frac{1}{2}})A_1(z^{\frac{1}{2}}) + X(-z^{\frac{1}{2}})A_1(-z^{\frac{1}{2}})\right] \tag{13}$$

After upsampling by a factor of 2, the z-transform of the reconstructed x is obtained as,

$$\begin{aligned} X_{up}\left(z\right) =& \frac{1}{2}\left[X\left(z\right)A_0\left(z\right) + X\left(-z\right)A_0\left(-z\right)\right]S_0\left(z\right) \\ &+ \frac{1}{2}[X(z)A_1(z) + X(-z)A_1(-z)]S_1(z) \\ =& \frac{1}{2}\left[A_0\left(z\right)S_0\left(z\right) + A_1\left(z\right)S_1\left(z\right)\right]X\left(z\right) \\ &+ \frac{1}{2}\left[A_0(-z)S_0(z) + A_1(-z)S_1(z)\right]X(-z) \end{aligned} \tag{14}$$

where $X(-z)$ denotes an aliased signal in the reconstruction, which is set to cancel its effect,

$$A_0\left(-z\right)S_0\left(z\right) + A_1\left(-z\right)S_1\left(z\right) = 0 \tag{15}$$

Obviously, that gives another item,

$$A_0\left(z\right)S_0\left(z\right) + A_1\left(z\right)S_1\left(z\right) = 2 \tag{16}$$

Splicing Eq. 15 and Eq. 16 gives the perfect reconstruction condition. Using the convolutional property of the z-transform, the loss function shown in $L_{wavelet}$ (see Sec. **??**) optimized toward the zeros can be obtained.

A.2. Initialization of Wavelet Convolution

For $K_{wavelet}$ in Eq. **??**, we performed the following experiments using db2 wavelet initialization [10], haar wavelet initialization [3] and random initialization.

Method	PSNR	SSIM	wavelet kernel size
db2 [10]	32.69	0.931	4×4
haar [3]	32.62	0.931	2×2
random	32.53	0.930	2×2

Table 7. Comparison of results of different initialization wavelet ways.

As shown in Tab. 7, better accuracy can be obtained by using subtly constructed classical wavelets for initialization, which corresponds to the introduction of artificial prior knowledge. Considering the balance between efficiency and accuracy, in the specific implementation of MLWNet, we use haar wavelets for initialization. From the above tables, it can be seen that the initialization using haar wavelet with a priori information results in a higher PSNR metric of 0.09 dB compared to random initialization.

A.3. GPU Memory Usage under High-Resolution Images

In this section, we use a Tesla A40 with 48GB GPU memory to test the average memory usage on the RSBlur dataset (the image resolution is approximately 1920×1200). The comparison results with various advanced algorithms are shown in Tab. 8. It is noted that our method uses only 1.4GB more memory than MIMO-UNet+ [1], and is only one-fifth of the advanced algorithm FFTformer [5]. Combining Tab. **??**, Tab. **??** and Tab. **??**, it can be seen that our proposed MLWNet is able to well balance the accuracy and model complexity at large resolution, and the smaller memory allocation also ensures that the algorithm can be deployed on some GPUs with poorer arithmetic power.

A.4. More Experimental Results

In this section, we provide more visual comparisons of the proposed method and state-of-the-art ones on RealBlur-J [8] (see Fig. 9, Fig. 10) and RSBlur [9] (see Fig. 11, Fig. 12, Fig. 13) datasets.

References

[1] Sung-Jin Cho, Seo-Won Ji, Jun-Pyo Hong, Seung-Won Jung, and Sung-Jea Ko. Rethinking coarse-to-fine approach in single image deblurring. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4641–4650, 2021. 2, 3, 4, 5

[2] Yuning Cui, Yi Tao, Zhenshan Bing, Wenqi Ren, Xinwei Gao, Xiaochun Cao, Kai Huang, and Alois Knoll. Selective frequency network for image restoration. In *International Conference on Learning Representations, ICLR*, 2023. 4, 5

[3] Alfred Haar. *Zur theorie der orthogonalen funktionensysteme*. Georg-August-Universitat, Gottingen., 1909. 1

[4] Kiyeon Kim, Seungyong Lee, and Sunghyun Cho. Mssnet: Multi-scale-stage network for single image deblurring. In *European Conference on Computer Vision*, pages 524–539. Springer, 2022. 3

[5] Lingshun Kong, Jiangxin Dong, Jianjun Ge, Mingqiang Li, and Jinshan Pan. Efficient frequency domain-based transformers for high-quality image deblurring. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5886–5895, 2023. 2, 3

[6] Orest Kupyn, Tetiana Martyniuk, Junru Wu, and Zhangyang Wang. Deblurgan-v2: Deblurring (orders-of-magnitude) faster and better. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8878–8887, 2019. 3

[7] Yawei Li, Yuchen Fan, Xiaoyu Xiang, Denis Demandolx, Rakesh Ranjan, Radu Timofte, and Luc Van Gool. Efficient and explicit modelling of image hierarchies for image restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18278–18289, 2023. 3

[8] Jaesung Rim, Haeyun Lee, Jucheol Won, and Sunghyun Cho. Real-world blur dataset for learning and benchmarking deblurring algorithms. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXV 16*, pages 184–201. Springer, 2020. 2, 3

[9] Jaesung Rim, Geonung Kim, Jungeon Kim, Junyong Lee, Seungyong Lee, and Sunghyun Cho. Realistic blur synthesis for learning image deblurring. In *European conference on computer vision*, pages 487–503. Springer, 2022. 2, 4, 5

[10] Alistair CH Rowe and Paul C Abbott. Daubechies wavelets and mathematica. *Computers in Physics*, 9(6):635–648, 1995. 1

[11] Xin Tao, Hongyun Gao, Xiaoyong Shen, Jue Wang, and Jiaya Jia. Scale-recurrent network for deep image deblurring. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8174–8182, 2018. 3, 4, 5

[12] Fu-Jen Tsai, Yan-Tsung Peng, Yen-Yu Lin, Chung-Chi Tsai, and Chia-Wen Lin. Stripformer: Strip transformer for fast image deblurring. In *European Conference on Computer Vision*, pages 146–162. Springer, 2022. 3

[13] Fu-Jen Tsai, Yan-Tsung Peng, Chung-Chi Tsai, Yen-Yu Lin, and Chia-Wen Lin. Banet: a blur-aware attention network for dynamic scene deblurring. *IEEE Transactions on Image Processing*, 31:6789–6799, 2022. 3

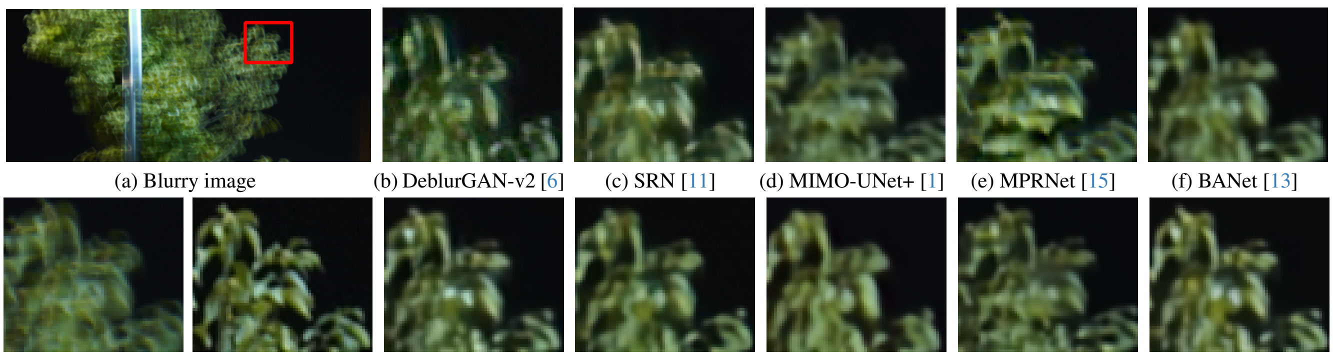
[14] Zhendong Wang, Xiaodong Cun, Jianmin Bao, Wengang Zhou, Jianzhuang Liu, and Houqiang Li. Uformer: A general u-shaped transformer for image restoration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 17683–17693, 2022. 4, 5

[15] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, Ming-Hsuan Yang, and Ling Shao. Multi-stage progressive image restoration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14821–14831, 2021. 3, 4, 5

[16] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, and Ming-Hsuan Yang. Restormer: Efficient transformer for high-resolution image restoration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5728–5739, 2022. 4, 5

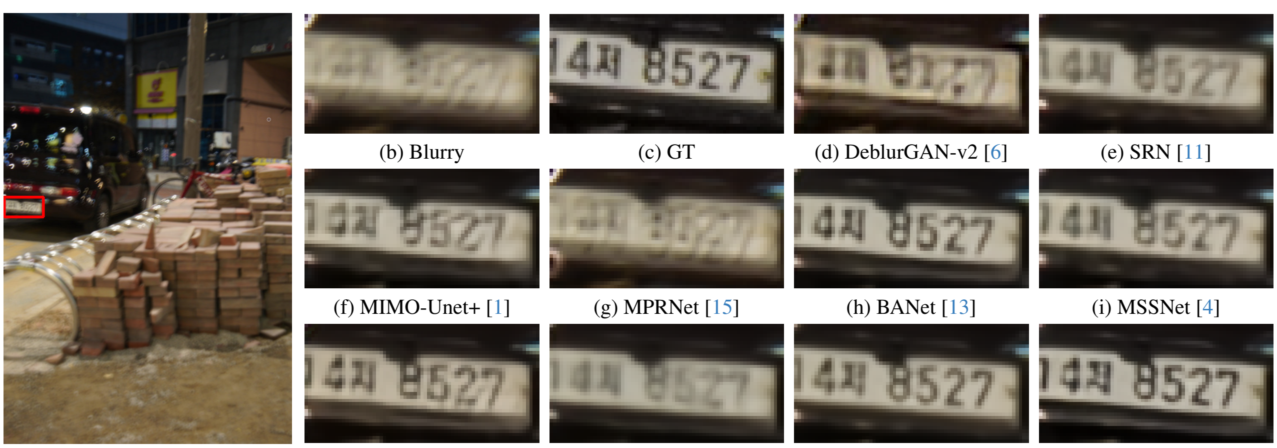
Method	MIMO-UNet+	MSSNet	MPRNet	Stripformer	GRL	FFTformer	MLWNet-B(Ours)
GPU memory(GB)	8.5	13.7	21.8	27.5	Out of memory	47.3	9.9

Table 8. Memory comparisons of our method and the advanced algorithms. “GPU memory” denotes the maximum GPU memory consumption that is computed by the“torch.cuda.max_memory_allocated()” function.



(a) Blurry image (b) DeblurGAN-v2 [6] (c) SRN [11] (d) MIMO-UNet+ [1] (e) MPRNet [15] (f) BANet [13]
(g) Blurry (h) GT (i) MSSNet [4] (j) Stripformer [12] (k) GRL [7] (l) FFTformer [5] (m) Ours

Figure 9. Visualization results for the RealBlur-J [8] dataset. Our proposed MLWNet performs excellently under the conditions of rich light, shadow, and details.



(b) Blurry (c) GT (d) DeblurGAN-v2 [6] (e) SRN [11]
(f) MIMO-Unet+ [1] (g) MPRNet [15] (h) BANet [13] (i) MSSNet [4]
(a) Blurry image (j) Stripformer [12] (k) GRL [7] (l) FFTformer [5] (m) Ours

Figure 10. Visualization results for the RealBlur-J [8] dataset. Our proposed MLWNet restores the image closest to GT, and it can be seen that MLWNet excels in terms of color, sharpening, and details.

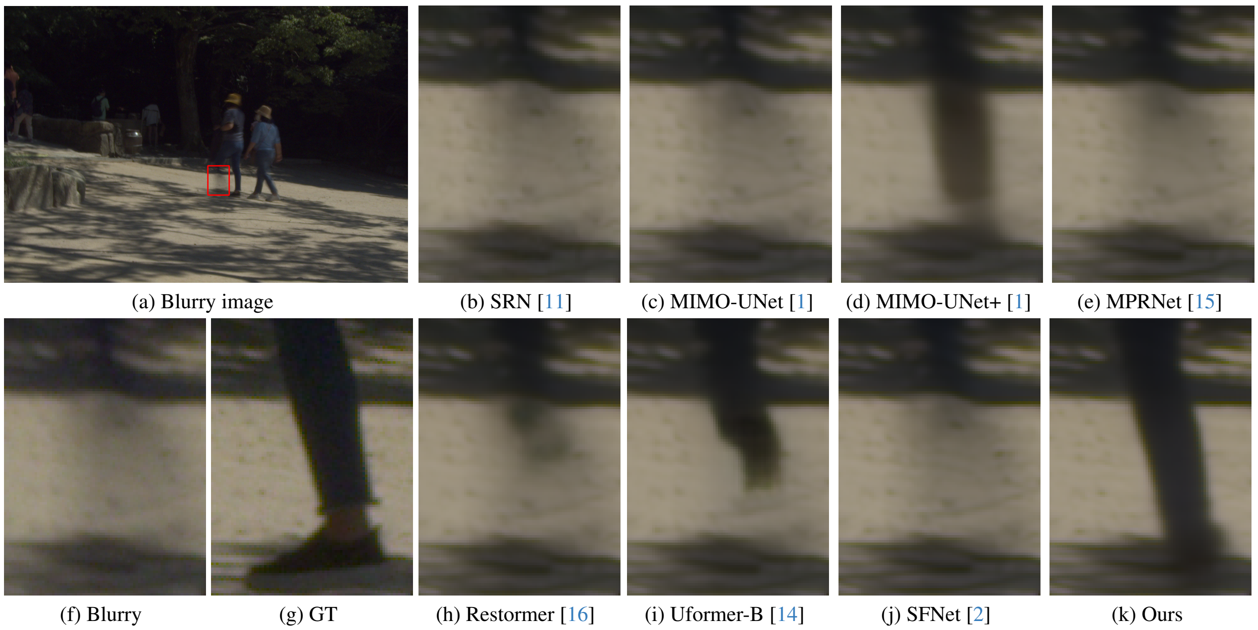


Figure 11. Visualization results for the RSBlur [9] dataset. Our proposed MLWNet restores the sharpest leg details of fast-moving pedestrians, which shows that MLWNet still performs well in the difficult motion blur restoration.

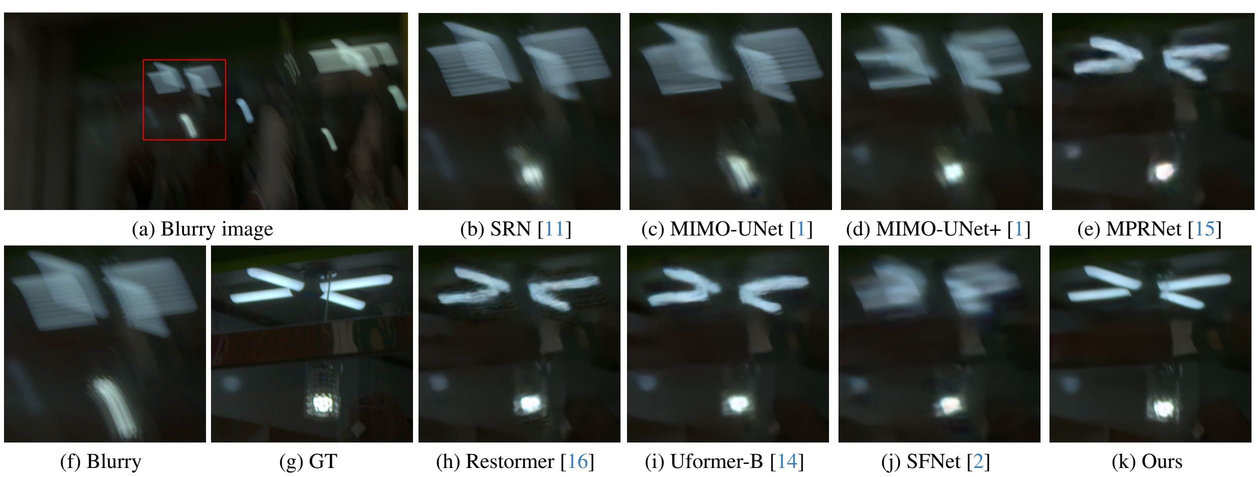
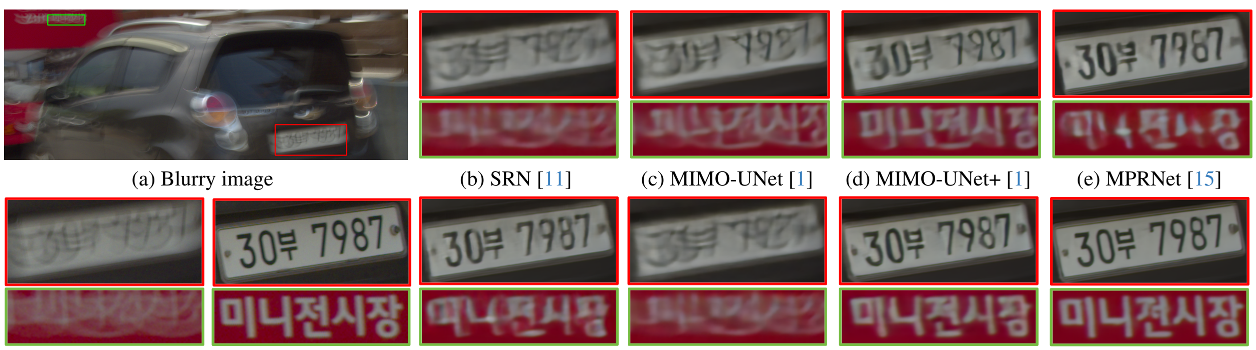


Figure 12. Visualization results for the RSBlur [9] dataset. Our proposed MLWNet restores the most realistic image in terms of light and shadow. It can be easily seen that MLWNet has the least retention on residual shadows.



(a) Blurry image (b) SRN [11] (c) MIMO-UNet [1] (d) MIMO-UNet+ [1] (e) MPRNet [15]

(f) Blurry (g) GT (h) Restormer [16] (i) Uformer-B [14] (j) SFNet [2] (k) Ours

Figure 13. Visualization results for the RSBlur [9] dataset. Our proposed MLWNet restores the clearest text details and achieves eye-pleasing results in noise suppression.