Efficient Visual State Space Model for Image Deblurring
- Supplemental Material -

Lingshun Kong[1], Jiangxin Dong[1], Jinhui Tang[1], Ming-Hsuan Yang[2,3], Jinshan Pan[1,†]
[1]Nanjing University of Science and Technology
[2]University of California, Merced [3]Google DeepMind

Overview

In this document, we first provide further analysis of the proposed method in Section A. Section B provides more visual comparisons of the proposed method and state-of-the-art ones.

A. Further Analysis on the Proposed Method

All baseline methods in this supplemental material are trained for 300K iterations with an initial learning rate of 1×10^{-3}, which is gradually reduced to 1×10^{-6} with the cosine annealing schedule. The batch size is set as 64 and the patch size is set as 128×128 pixels.

Effectiveness of the geometric transformation on Mamba2 [6]-based SSMs. Mamba2 [6] introduces a simplified architecture that eliminates sequential linear projections, opting instead for the generation of SSM parameters at the outset, independent of the input X. This modification, along with the ability to leverage larger state dimensions, has resulted in a model that not only trains faster but also scales more effectively. To demonstrate the effect of the geometric transformation on Mamba2 [6]-based SSMs, we further compare two baselines that respectively replace the EVS block in the proposed EVSSM (i.e., Figure 1(c) of the main paper) with the Mamba2 block in [6] (EVSSM2 w/o GeoT for short) and the proposed geometric transformation strategy followed by the Mamba2 block in [6] (EVSSM2 w/ GeoT for short). The comparison results in Table 9 demonstrate that using the geometric transformation achieves better results, improving the PSNR by 0.25dB.

Table 9. Effectiveness of the geometric transformation on Mamba2 [6]-based SSMs, evaluated on the GoPro dataset [14].

Method	Params (M)	FLOPs (G)	PSNR (dB)	SSIM
EVSSM2 w/o GeoT	24M	148	33.94	0.9683
EVSSM2 w/ GeoT	24M	148	**34.19**	**0.9691**

Effectiveness of the frequency loss. The frequency loss, i.e., the second term of Eq. (1), treats each frequency component equally and is widely used in existing image restoration methods [4,8]. It optimizes images in the frequency domain by directly targeting the frequency characteristics of blurriness and is able to effectively restore high-frequency details. In addition, the loss in Eq. (1) measures the similarity between the ground truth and the result in both the spatial and frequency domains, ensuring that the restored image aligns closely with the target in both spatial and frequency representations for better image deblurring as shown in Table 10.

Table 10. Effectiveness of the frequency loss, evaluated on the GoPro dataset [14].

Method	PSNR (dB)	SSIM
EVSSM (L1 loss)	33.86	0.9673
EVSSM (L1&Frequency loss)	**34.15**	**0.9690**

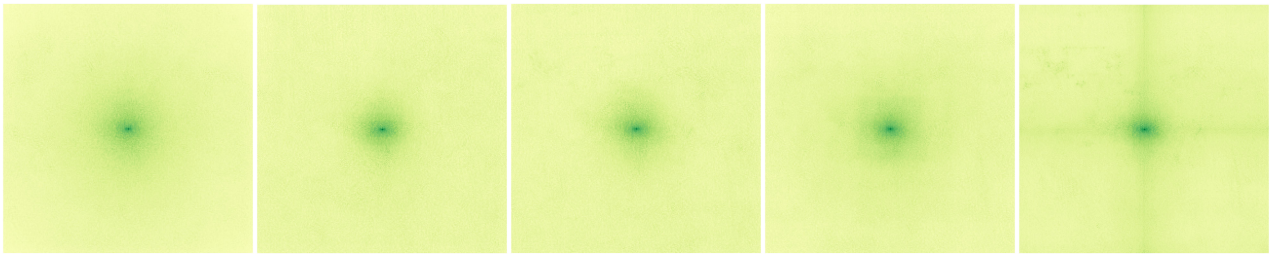
†Corresponding author

Selection for the kernel size of the depth-wise convolution and its placement. We choose a 7-size kernel in depth-wise convolution to better aggregate local information while maintaining efficient computation. It strikes a balance between capturing fine details and enabling global-local feature fusion, which is critical for high-quality restoration. We place it after the 1×1 convolution to generate weights for SSM computation, allowing the model to effectively integrate local and global features. We evaluate the effect of the kernel size by varying it from 3×3 to 9×9 in Table 11. We use a kernel size of 7×7 as a trade-off between image quality and efficiency.

Table 11. Effectiveness of the kernel size of the depth-wise convolution, evaluated on the GoPro dataset [14].

Kernel size	3×3	5×5	7×7	9×9
PSNR (dB)/SSIM	33.08/0.9687	33.12/0.9689	34.15/0.9690	34.15/0.9691
Runtime (ms)	83	85	89	97

Comparison of effective receptive fields. We further analyze the effectiveness of our proposed method in obtaining a global receptive field. As shown in [2, 13, 22], the spatial and structural properties can effectively characterize the image. Thus, a global receptive field is extremely important for image restoration tasks, including image deblurring. However, when handling visual tasks, the standard SSM, which flattens the image data into a 1D sequence, will disrupt the spatial structure of an image and make it difficult to capture local information from various adjacent pixels. In addition, the standard SSM, as a recursive process, can only utilize information from previous timesteps and not consider information from future timesteps.



(a) MIMO-UNet+ [4] (b) NAFNet [3] (c) Restormer [21] (d) FFTformer [8] (e) EVSSM (ours)

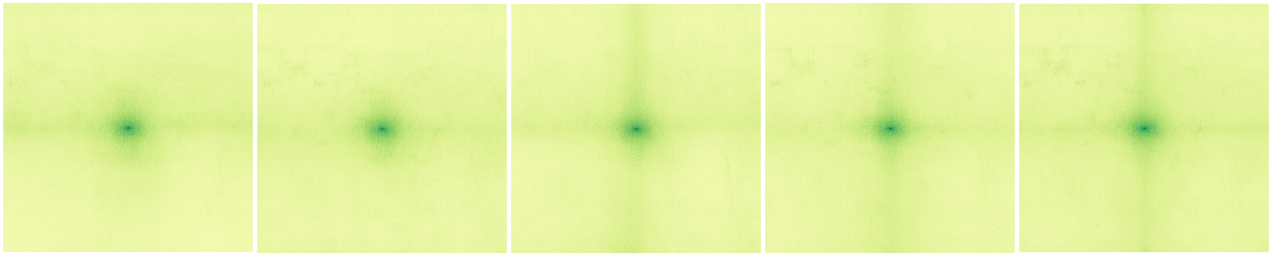
Figure 4. Visualization of effective receptive field (ERF) for NAFNet, Restormer, FFTformer, and our EVSSM. A larger ERF is indicated by a more extensively distributed dark area. Compared to existing methods in (a)-(d), our EVSSM achieves a significantly larger effective receptive field in (e).

In contrast, our approach is more effective in capturing both local and non-local information by applying two geometric transformations, since it allows SSM to make use of more local information from different directions and capture long-range modeling from both previous and future timesteps. To intuitively illustrate the effectiveness of our method, we visualize the effective receptive field of competing methods in Figure 4. Compared to existing CNN-based [3, 4] and Transformer-based methods [8, 21], our approach with both geometric transformations is able to obtain a larger receptive field in Figure 4, allowing the network to capture information from a wider region, both locally and non-locally.

Compared to the baseline methods that receptively remove both geometric transformations ("EVSSM w/o F&T" in Figure 5(a)), remove the transpose transformation ("EVSSM w/o T" in Figure 5(b)), and remove the flip transformation ("EVSSM w/o F" in Figure 5(c)), our method with both geometric transformations is able to obtain a larger receptive field in Figure 5(e), allowing the network to capture information from a wider region, both locally and non-locally. Compared to the method ("four-direction" in Figure 5(d)) that scans in four directions simultaneously [12], our approach can achieve the same receptive field.

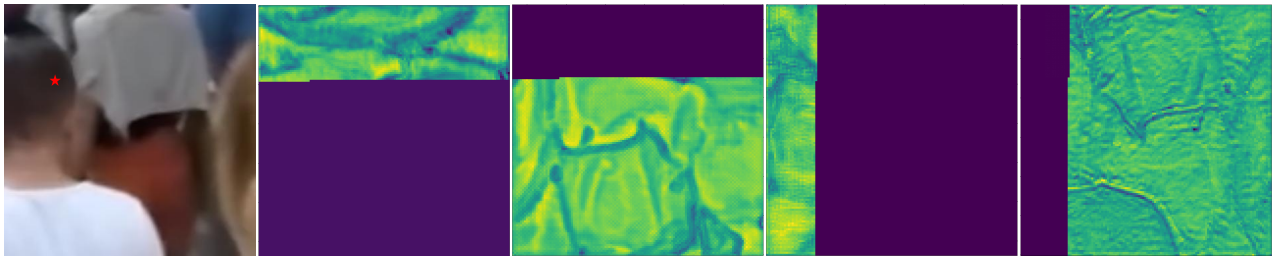
To intuitively illustrate why the geometric transformation works, we visualize the activation map in Figure 6. When geometric transformations are not used, the model is constrained to use information from preceding pixels (cf. Figure 6(b)) when estimating the value of the red-marked pixel in Figure 6(a). After applying geometric transformations, the proposed model obtains the activation maps in Figure 6(b) to (e), thereby enabling the utilization of all pixels in the image.

The difference between the proposed EVS and self-ensemble. Self-ensemble is a technique used during the testing phase of a model where the degraded input is flipped and rotated to create multiple distinct degraded images. These images are then restored using the same model, and finally, all the results are averaged with weights. This is a post-processing approach to enhance the model's performance during the testing phase without altering the model's architecture and affecting the training



(a) EVSSM w/o F&T (b) EVSSM w/o T (c) EVSSM w/o F (d) four-direction (e) EVSSM (ours)

Figure 5. Visualization of effective receptive field (ERF) for our EVSSM with different geometric transformations. A larger ERF is indicated by a more extensively distributed dark area. When applying both flip and transpose transformations, our method can achieve a larger effective receptive field in (e).



(a) Blurred input (b) (c) (d) (e)

Figure 6. Illustration of the activation map for query patches indicated by red stars. (b)-(e) are visualizations of activation maps associated with different EVSS modules. The purple regions indicate those pixels that are not activated.

Table 12. Effectiveness of the self-ensemble strategy for EVSSM, evaluated on the GoPro dataset [14].

Method	FLOPs (G)	Runtime (ms)	PSNR (dB)	SSIM
EVSSM w/o GeoT	126	88	33.89	0.9671
EVSSM w/o GeoT (self-ensemble)	504	352	34.14	0.9690
EVSSM	126	89	34.15	0.9690
EVSSM (self-ensemble)	504	356	34.37	0.9701

Table 13. Quantitative evaluations on the DPDD dataset [1] for single image defocus deblurring.

Method	EBDB [7]	DMENet [10]	JNB [17]	DPDNet [1]	KPAC [18]	IFAN [11]	Restormer [21]	EVSSM (ours)
PSNRs (dB)	23.45	23.41	23.84	24.34	25.22	25.37	25.98	**26.16**
SSIMs	0.683	0.714	0.715	0.747	0.774	0.789	0.811	**0.823**

process. However, employing this strategy can significantly increase the inference cost of the entire model. In contrast, our approach introduces geometric transformations before each layer to equip the model with a broader receptive field. This represents an architectural change to the model and is implemented with minimal increase in computational cost. We further evaluate the effect of the self-ensemble strategy on the proposed method in Table 12. Similar to previous methods, the use of the self-ensemble strategy improved our method's performance by 0.24 dB on the GoPro dataset. However, adopting this strategy increases the computational cost of the entire network to four times that of the original. Therefore, we have not employed this strategy in our paper.

Evaluations on the DPDD [1] dataset for single image defocus deblurring We show the quantitative evaluations of the proposed method against competing methods in the single image defocus deblurring task in Table 13. Our approach still outperforms other methods, which demonstrates the effectiveness of our method on different types of blur.

B. Qualitative Comparisons

In this section, we provide more visual comparisons of the proposed method with state-of-the-art ones on both synthetic and real-world benchmarks in Figures 7-12.

References

[1] Abdullah Abuolaim and Michael S Brown. Defocus deblurring using dual-pixel data. In *ECCV*, 2020. 3

[2] Yuval Bahat and Michal Irani. Blind dehazing using internal patch recurrence. In *ICCP*, 2016. 2

[3] Liangyu Chen, Xiaojie Chu, Xiangyu Zhang, and Jian Sun. Simple baselines for image restoration. In *ECCV*, 2022. 2, 5, 6, 7

[4] 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 *ICCV*, 2021. 1, 2, 5, 6, 7, 8, 9, 10

[5] Xiaojie Chu, Liangyu Chen, , Chengpeng Chen, and Xin Lu. Improving image restoration by revisiting global information aggregation. In *ECCV*, 2022. 5, 6, 7, 8

[6] Tri Dao and Albert Gu. Transformers are SSMs: Generalized models and efficient algorithms through structured state space duality. In *ICML*, 2024. 1

[7] Ali Karaali and Claudio Rosito Jung. Edge-based defocus blur estimation with adaptive scale selection. *IEEE TIP*, 27(3):1126–1137, 2017. 3

[8] Lingshun Kong, Jiangxin Dong, Jianjun Ge, Mingqiang Li, and Jinshan Pan. Efficient frequency domain-based transformers for high-quality image deblurring. In *CVPR*, 2023. 1, 2, 5, 6, 7, 8, 9, 10

[9] Orest Kupyn, Volodymyr Budzan, Mykola Mykhailych, Dmytro Mishkin, and Jiri Matas. Deblurgan: Blind motion deblurring using conditional adversarial networks. In *CVPR*, 2018. 9, 10

[10] Junyong Lee, Sungkil Lee, Sunghyun Cho, and Seungyong Lee. Deep defocus map estimation using domain adaptation. In *CVPR*, 2019. 3

[11] Junyong Lee, Hyeongseok Son, Jaesung Rim, Sunghyun Cho, and Seungyong Lee. Iterative filter adaptive network for single image defocus deblurring. In *CVPR*, 2021. 3

[12] Yue Liu, Yunjie Tian, Yuzhong Zhao, Hongtian Yu, Lingxi Xie, Yaowei Wang, Qixiang Ye, and Yunfan Liu. Vmamba: Visual state space model. *arXiv preprint arXiv:2401.10166*, 2024. 2

[13] Tomer Michaeli and Michal Irani. Blind deblurring using internal patch recurrence. In *ECCV*, 2014. 2

[14] Seungjun Nah, Tae Hyun Kim, and Kyoung Mu Lee. Deep multi-scale convolutional neural network for dynamic scene deblurring. In *CVPR*, 2016. 1, 2, 3, 5, 6

[15] Jaesung Rim, Haeyun Lee, Jucheol Won, and Sunghyun Cho. Real-world blur dataset for learning and benchmarking deblurring algorithms. In *ECCV*, 2020. 9, 10

[16] Ziyi Shen, Wenguan Wang, Xiankai Lu, Jianbing Shen, Haibin Ling, Tingfa Xu, and Ling Shao. Human-aware motion deblurring. In *ICCV*, 2019. 7, 8

[17] Jianping Shi, Li Xu, and Jiaya Jia. Just noticeable defocus blur detection and estimation. In *CVPR*, 2015. 3

[18] Hyeongseok Son, Junyong Lee, Sunghyun Cho, and Seungyong Lee. Single image defocus deblurring using kernel-sharing parallel atrous convolutions. In *ICCV*, 2021. 3

[19] Xin Tao, Hongyun Gao, Xiaoyong Shen, Jue Wang, and Jiaya Jia. Scale-recurrent network for deep image deblurring. In *CVPR*, 2018. 9, 10

[20] Fu-Jen Tsai, Yan-Tsung Peng, Yen-Yu Lin, Chung-Chi Tsai, and Chia-Wen Lin. Stripformer: Strip transformer for fast image deblurring. In *ECCV*, 2022. 5, 6, 7, 8, 9, 10

[21] 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 *CVPR*, 2022. 2, 3, 7, 8

[22] Maria Zontak, Inbar Mosseri, and Michal Irani. Separating signal from noise using patch recurrence across scales. In *CVPR*, 2013. 2

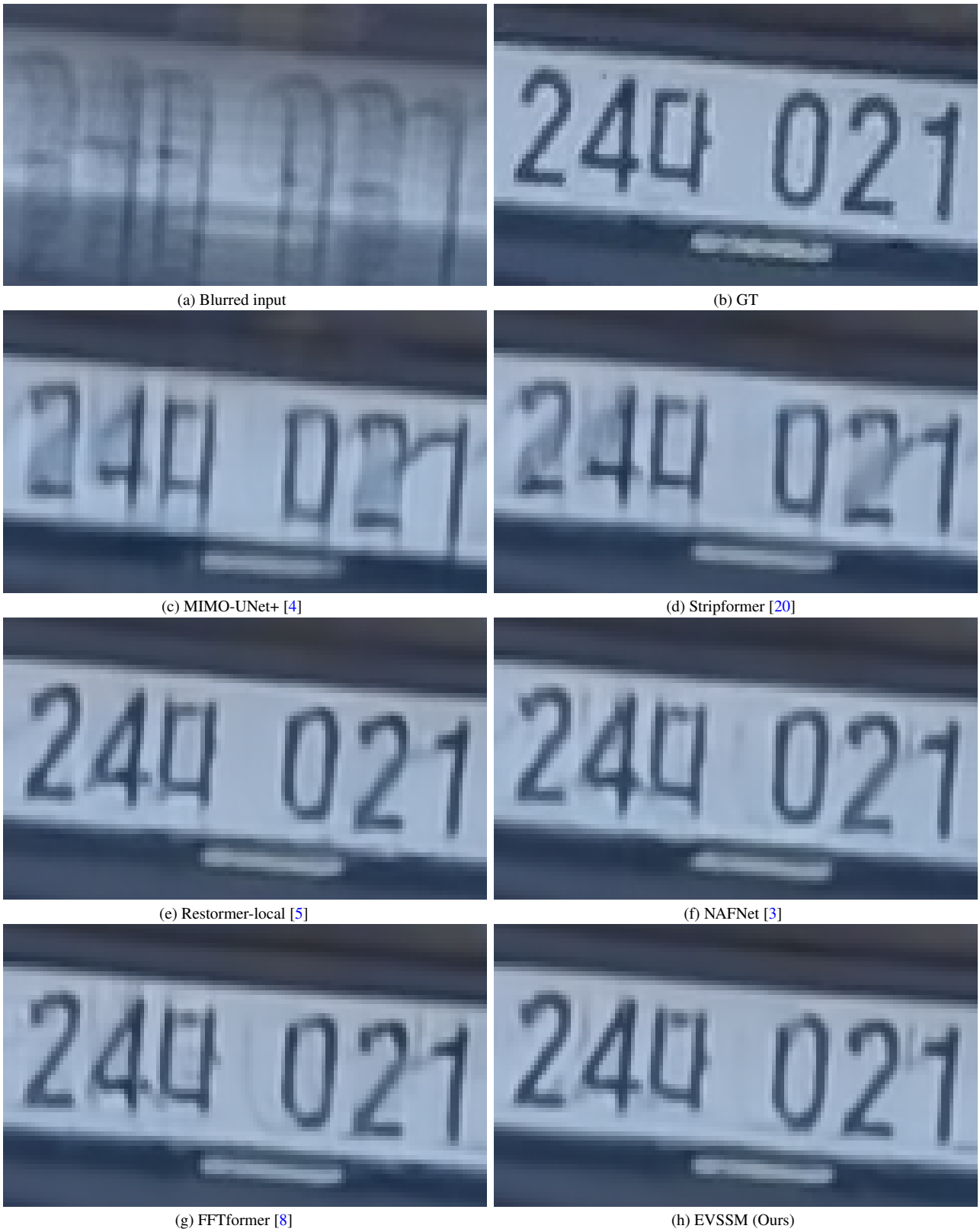
(a) Blurred input (b) GT

(c) MIMO-UNet+ [4] (d) Stripformer [20]

(e) Restormer-local [5] (f) NAFNet [3]

(g) FFTformer [8] (h) EVSSM (Ours)

Figure 7. Deblurred results on the GoPro dataset [14]. The deblurred results in (c)-(g) still contain significant blur effects. The proposed method generates a clearer image, where the structures of the windows are much clearer.



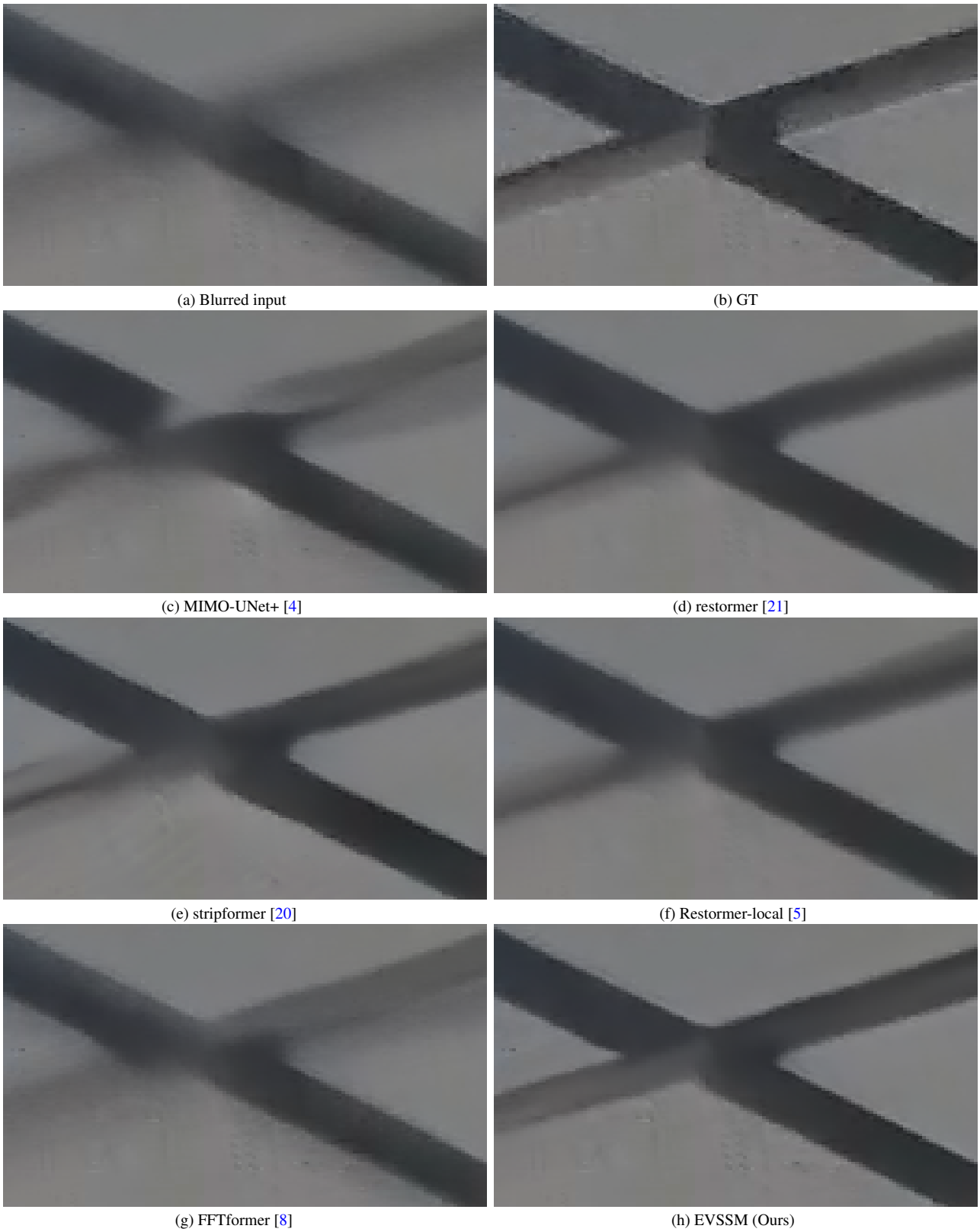
(a) Blurred input (b) GT

(c) MIMO-UNet+ [4] (d) Stripformer [20]

(e) Restormer-local [5] (f) NAFNet [3]

(g) FFTformer [8] (h) EVSSM (Ours)

Figure 8. Deblurred results on the GoPro dataset [14]. State-of-the-art methods [3–5, 8, 20] do not restore the numbers well. In contrast, our method generates a clearer image, where the numbers are much clearer.



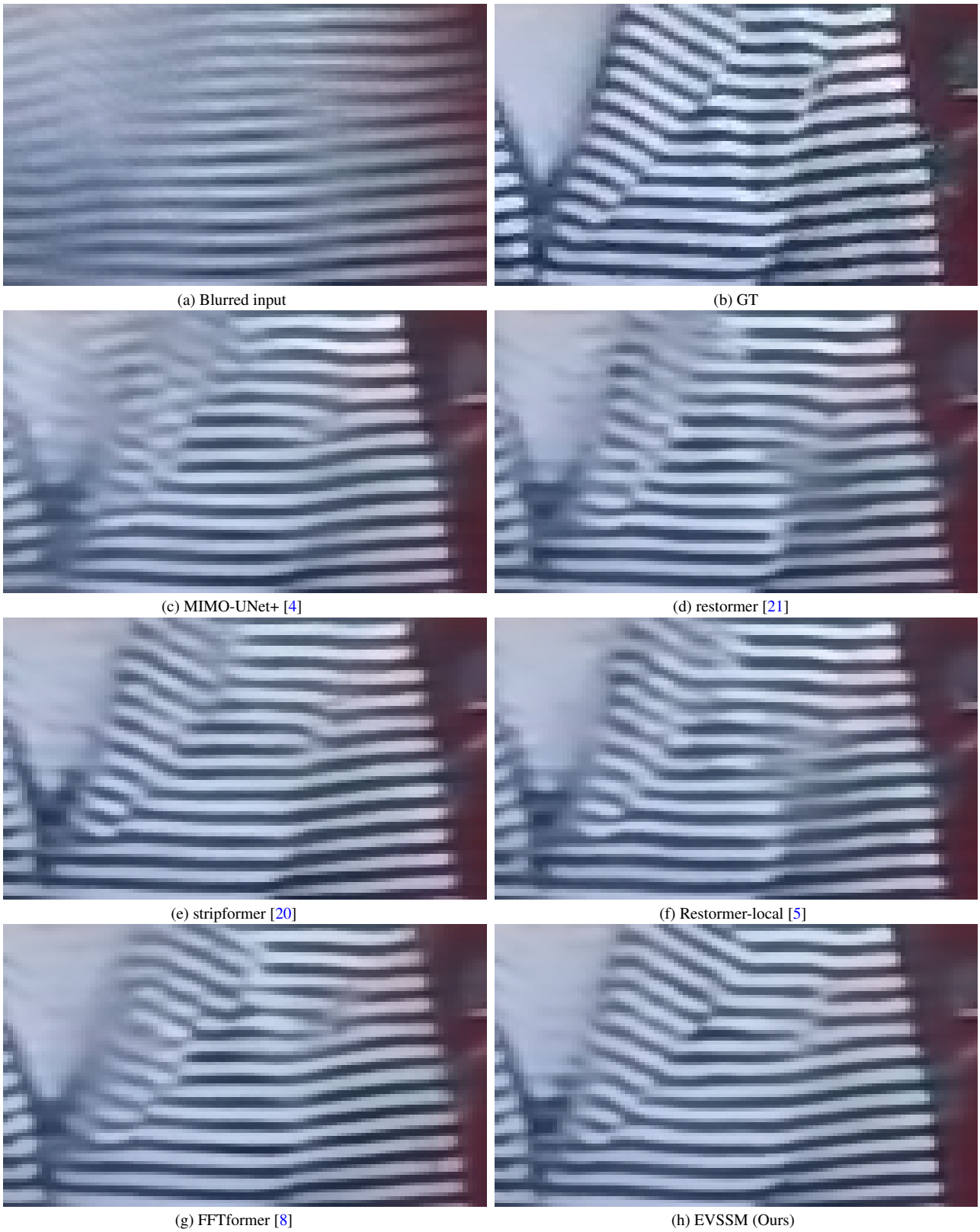
(a) Blurred input (b) GT

(c) MIMO-UNet+ [4] (d) restormer [21]

(e) stripformer [20] (f) Restormer-local [5]

(g) FFTformer [8] (h) EVSSM (Ours)

Figure 9. Deblurred results on the HIDE dataset [16]. State-of-the-art methods [3–5, 8, 20] do not restore the structures well. In contrast, the proposed method generates a better image with clearer structures.



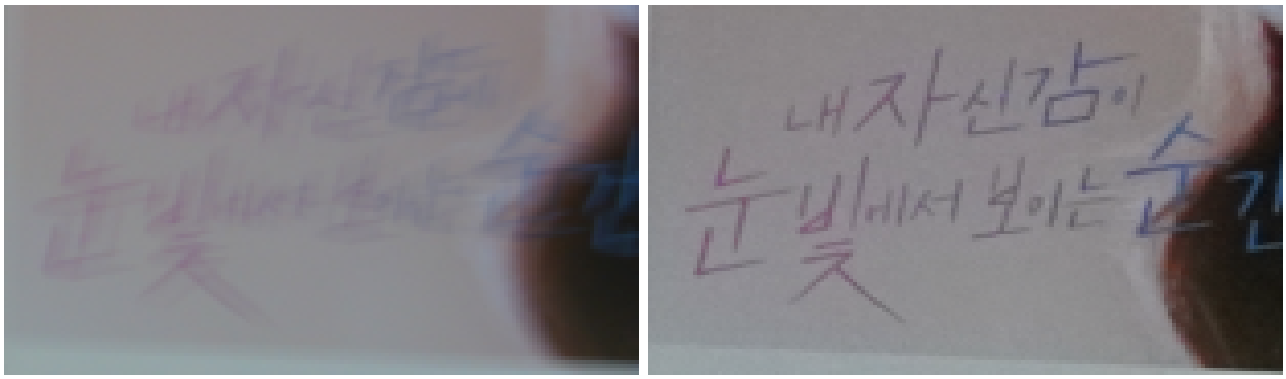
(a) Blurred input (b) GT

(c) MIMO-UNet+ [4] (d) restormer [21]

(e) stripformer [20] (f) Restormer-local [5]

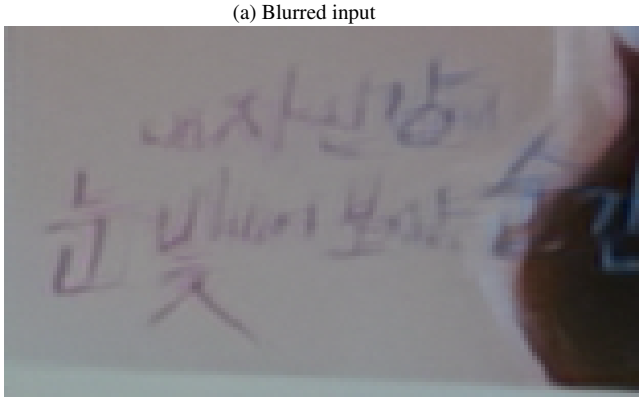
(g) FFTformer [8] (h) EVSSM (Ours)

Figure 10. Deblurred results on the HIDE dataset [16]. The deblurred results in (c)-(g) still contain significant blur effects. The proposed method generates a clearer image. For example, the the textures of clothes are much clearer.

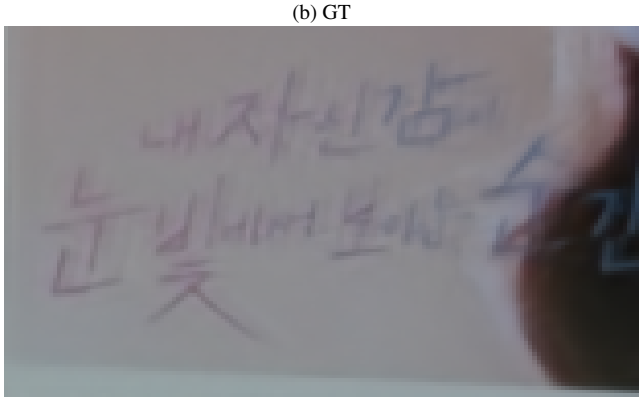


(a) Blurred input

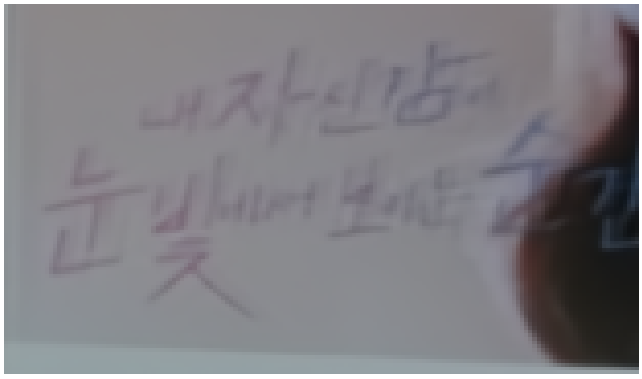
(b) GT



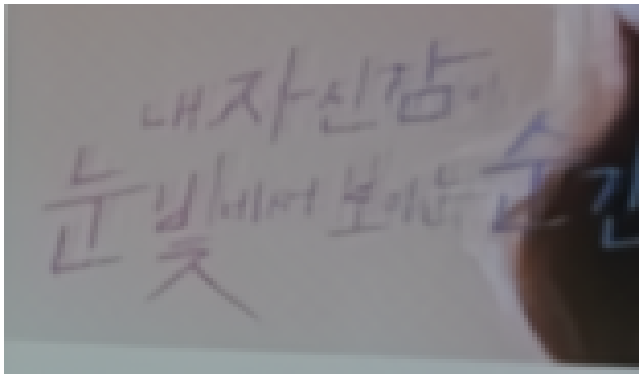
(c) DeblurGAN [9]



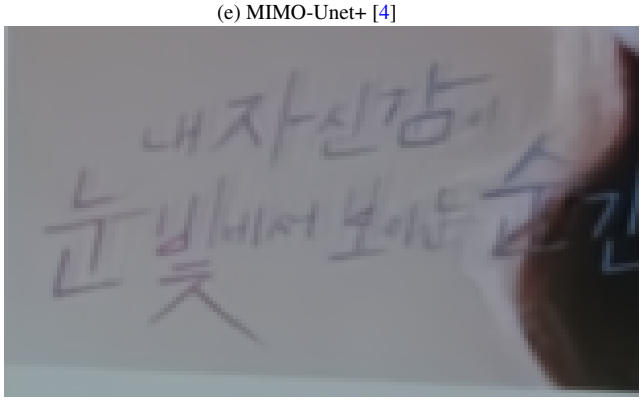
(d) SRN [19]



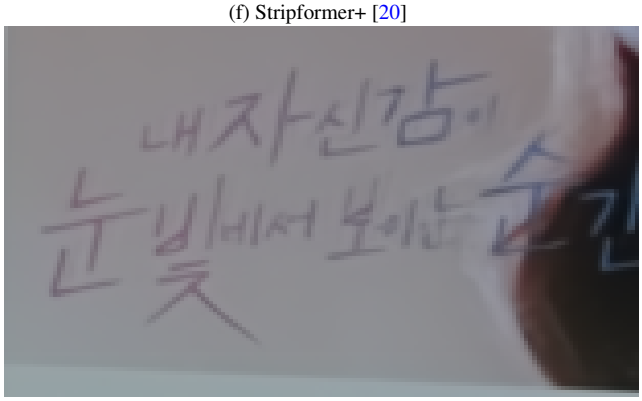
(e) MIMO-Unet+ [4]



(f) Stripformer+ [20]

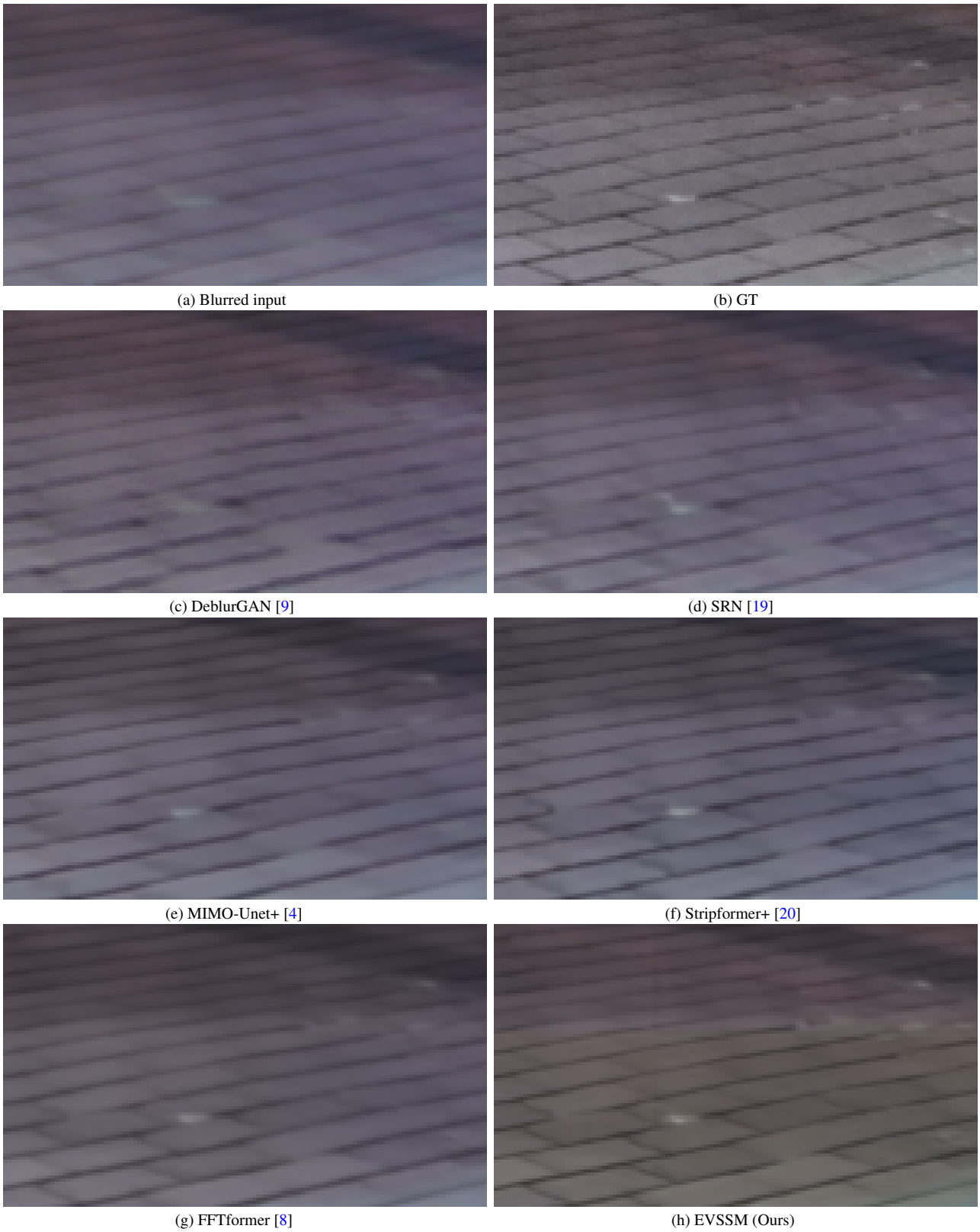


(g) FFTformer [8]



(h) EVSSM (Ours)

Figure 11. Deblurred results on the RealBlur dataset [15]. State-of-the-art methods [4, 8, 9, 19, 20] do not restore the characters well. In contrast, the proposed method generates a better image with clearer characters.



(a) Blurred input (b) GT

(c) DeblurGAN [9] (d) SRN [19]

(e) MIMO-Unet+ [4] (f) Stripformer+ [20]

(g) FFTformer [8] (h) EVSSM (Ours)

Figure 12. Deblurred results on the RealBlur dataset [15]. State-of-the-art methods [4, 8, 9, 19, 20] do not restore the texture on the ground well. In contrast, the proposed method generates a better image.