Spatio-Temporal Filter Adaptive Network for Video Deblurring Supplementary Materials

Shangchen Zhou[1*] Jiawei Zhang[1*] Jinshan Pan[2†] Haozhe Xie[1,3] Wangmeng Zuo[3] Jimmy Ren[1]
[1]SenseTime Research [2]Nanjing University of Science and Technology, Nanjing, China
[3]Harbin Institute of Technology, Harbin, China
https://shangchenzhou.com/projects/stfan

Overview

In this supplementary material, we provide additional information to complement the paper, which consists of the configurations of our networks, more explanations about FAC layer, and the detail structures of variant networks. At last, we present more qualitative experimental results compared with other algorithms.

1. Configurations of the Proposed Networks

As shown in Figure 2 in manuscript, our network is composed of three sub-networks: a spatio-temporal filter adaptive network (STFAN), a feature extraction network (FEN), and a reconstruction network (RN). The Table 1, 2 and 3 list the detailed configurations of STFAN, FEN and RN, respectively. And the table 4 presents the configurations of residual block, which are the basic building blocks of our proposed networks.

Table 1: Configurations of spatio-temporal filter adaptive network (STFAN). "Conv" denotes the convolution layer, "Res" denotes the residual block, and "$(\cdot)$" denotes the concatenate operation. B_{t-1}, R_{t-1} denote the blurry and restored image of the previous frame, respectively, and B_t denotes the current input blurry image.

STFAN	Input	Output	In channels	Out channels	Kernel size	Stride
Conv1	(R_{t-1}, B_{t-1}, B_t)	Conv1	9	32	3×3	1
Res2-3	Conv1	Res3	32	32	3×3	1
Conv4	Res3	Conv4	32	64	3×3	2
Res5-6	Conv4	Res6	64	64	3×3	1
Conv7	Res6	Conv7	64	128	3×3	2
Res8-9	Conv7	Res9	128	128	3×3	1
Conv10	Res9	Conv10	128	128	3×3	1
Res11-12	Conv10	Res12	128	128	3×3	1
Conv13	Res12	$\mathcal{F}_{align}$	128	$128 \times k^2$	1×1	1
Conv14	$\mathcal{F}_{align}$	Conv14	$128 \times k^2$	128	1×1	1
Conv15	(Res9, Conv14)	Conv15	256	128	3×3	1
Res16-17	Conv15	Res17	128	128	3×3	1
Conv18	Res17	$\mathcal{F}_{deblur}$	128	$128 \times k^2$	1×1	1
FAC19	$H_{t-1}, \mathcal{F}_{align}$	$\hat{H}_{t-1}$	$128 \times k^2$, 128	128	5×5	1
FAC20	$E_t, \mathcal{F}_{deblur}$	$\hat{E}_t$	$128 \times k^2$, 128	128	5×5	1
Concat21	$(\hat{H}_{t-1}, \hat{E}_t)$	C_t	128, 128	256	-	-
Conv22	C_t	H_t	256	128	3×3	1

*Equal contribution †Corresponding author: sdluran@gmail.com.

Table 2: Configurations of feature extraction network (FEN). "Conv" denotes the convolution layer and "Res" denotes the residual block.

FEN	Input	Output	In channels	Out channels	Kernel size	Stride
Conv1	B_t	Conv1	3	32	3×3	1
Res2-3	Conv1	Res3	32	32	3×3	1
Conv4	Res3	Conv4	32	64	3×3	2
Res5-6	Conv4	Res6	64	64	3×3	1
Conv7	Res6	Conv7	64	128	3×3	2
Res8-9	T_t	Res9	128	128	3×3	1

Table 3: Configurations of reconstruction network (RN). "Conv" denotes the convolution layer, "Res" denotes the residual block, "Upconv" denotes the up-sample layer by transposed convolution operator.

RN	Input	Output	In channels	Out channels	Kernel size	Stride
Upconv1	C_t	Upconv1	256	64	4×4	1/2
Res2-3	Upconv1	Res3	64	64	3×3	1
Upconv4	Res3	Upconv4	64	32	3×3	1/2
Res5-6	Upconv4	Res6	32	32	3×3	1
Conv7	Res6	Conv7	32	3	3×3	1
Sum	B_t, Conv7	R_t	3, 3	3	-	-

Table 4: Configurations of residual block. "In" denotes input of the block.

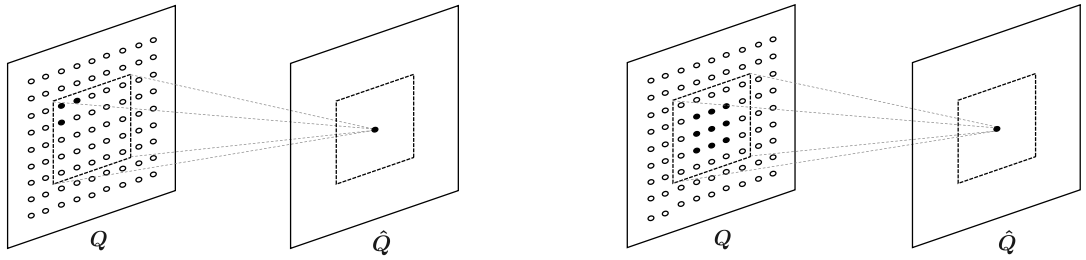
Residual Block	In channels	Out channels	Kernel size	Stride	Sum
Conv1	C_{in}	C_{in}	3×3	1	-
ReLU1	-	-	-	-	-
Conv2	C_{in}	C_{in}	3×3	1	In

2. More explanations on FAC layer

In this section, we present further explanations and discussions on proposed FAC layer.

2.1. Illustration of Alignment and Deblurring Processes by FAC layer

The frame alignment and deblurring are both spatially variant tasks. Using the proposed FAC layer, we consider these two processes as two filter adaptive convolution in the feature domain. As figure 1 shown, the convolution operation can transform the pixels in feature maps, which can be used for frames alignment (a) and deblurring (b) in the feature domain, using estimated corresponding element-wise filters.



(a) Alignment (b) Deblurring

Figure 1: Illustration of Alignment and Deblurring processes by FAC layer. Q and $\hat{Q}$ denote input feature maps and transformed feature maps, respectively.

2.2. Backwards of FAC layer

According to the manuscript, the forward pass of the proposed Filter Adaptive Convolutional (FAC) layer is as follows:

$$\begin{aligned}\hat{Q}(x,y,c_i) &= \mathcal{F}_{x,y,c_i} * Q_{x,y,c_i} \\ &= \sum_{n=-r}^{r}\sum_{m=-r}^{r} \mathcal{F}(x,y,k^2c_i+kn+m) \times Q(x-n,y-m,c_i),\end{aligned} \tag{1}$$

in which $r = \frac{k-1}{2}$, $\mathcal{F}$ is the generated filter, Q and $\hat{Q}$ denote the input features and transformed features, respectively. Based on (1), its backward pass can be presented as:

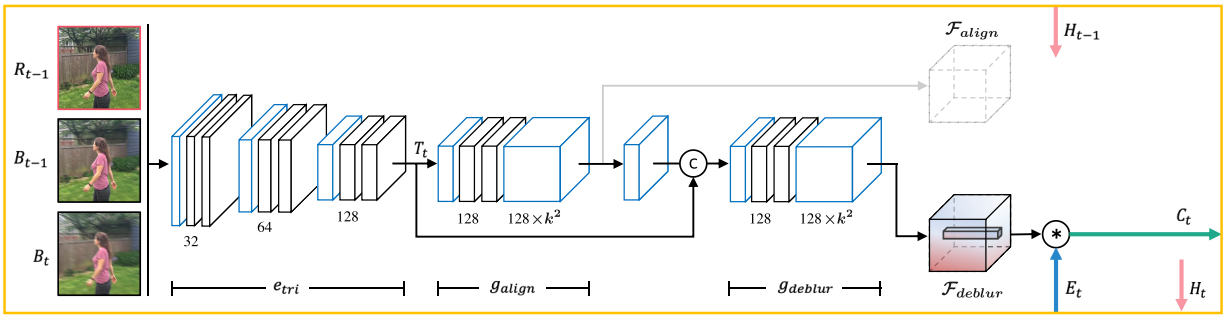
$$\Delta Q(x,y,c_i) = \sum_{n=-r}^{r}\sum_{m=-r}^{r} \mathcal{F}(x+n,y+m,k^2c_i+kn+m) \times \Delta\hat{Q}(x+n,y+m,c_i) \tag{2}$$

and

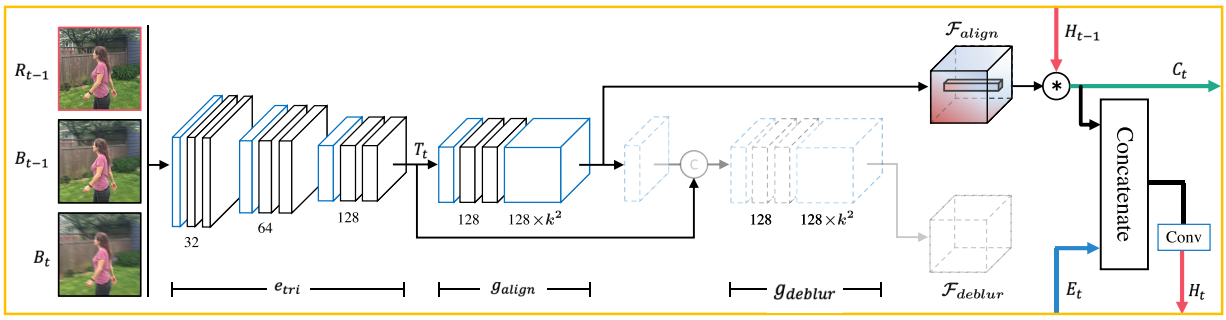
$$\Delta\mathcal{F}(x,y,k^2c_i+kn+m) = Q(x-n,y-m,c_i) \times \Delta\hat{Q}(x,y,c_i). \tag{3}$$

3. Detail structures of variant networks in manuscript Sec 5.2

We present detail structures of two variant networks (-, w D) and (w A, -) of STFAN in manuscript Sec 5.2. As shown in Figure 2, The (-, w D) removes the features of the alignment branch, which reconstructs current sharp image only from features from the previous frame. And (w A, -) removes features of the deblurring branch.



(a) The structure of STFAN in (-, w D)



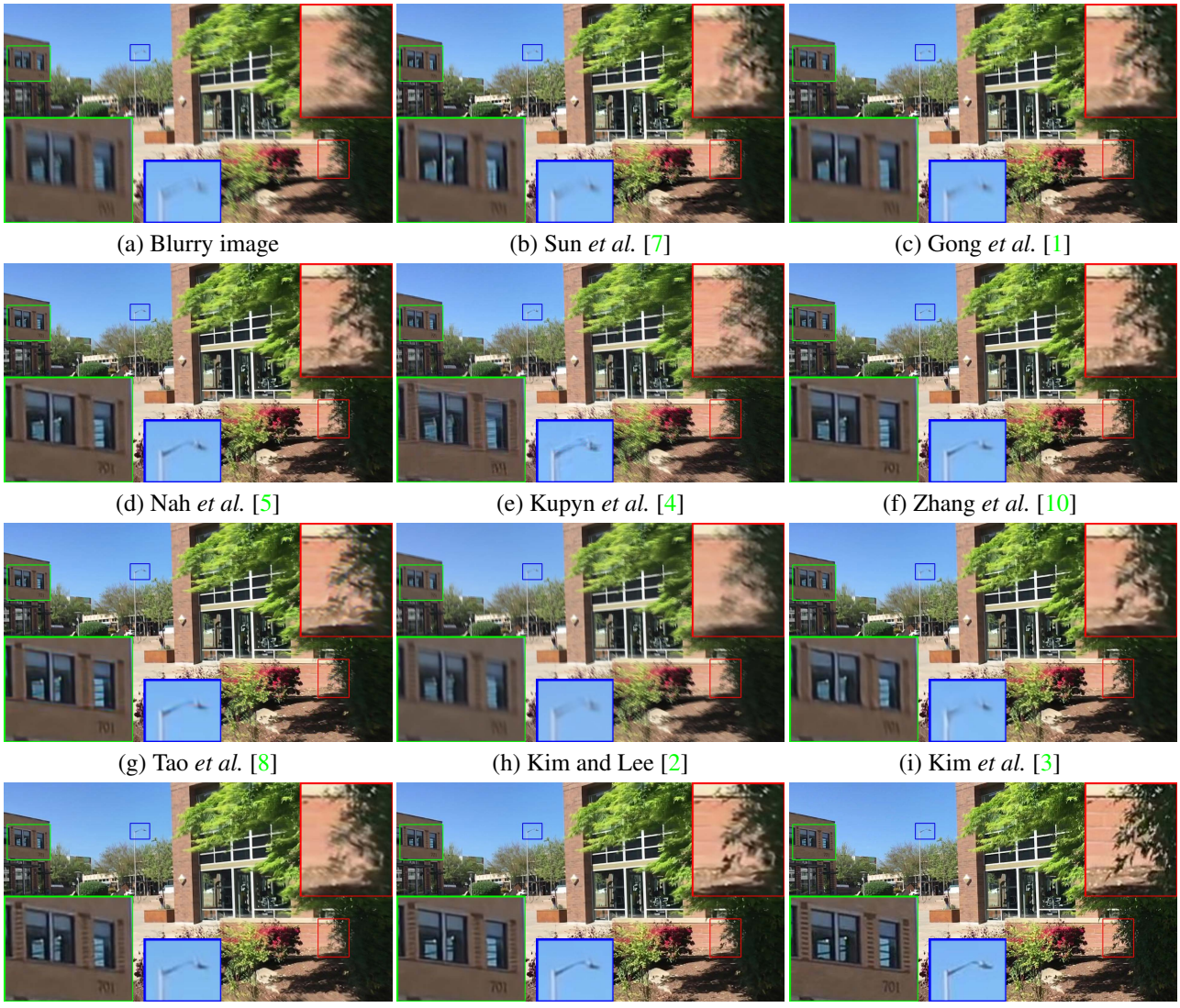
(b) The structure of STFAN in (w A, -)

Figure 2: Detail structures of STFAN in two variant networks (-, w D) and (w A, -) in manuscript Sec 5.2.

4. Qualitative Comparisons

In this section, we provide more visual comparisons with the state-of-the-art image and video deblurring methods [9, 7, 1, 5, 4, 10, 8, 2, 3, 6] on both synthetic dataset [6] and real blurred videos.

4.1. Video Delburring Dataset



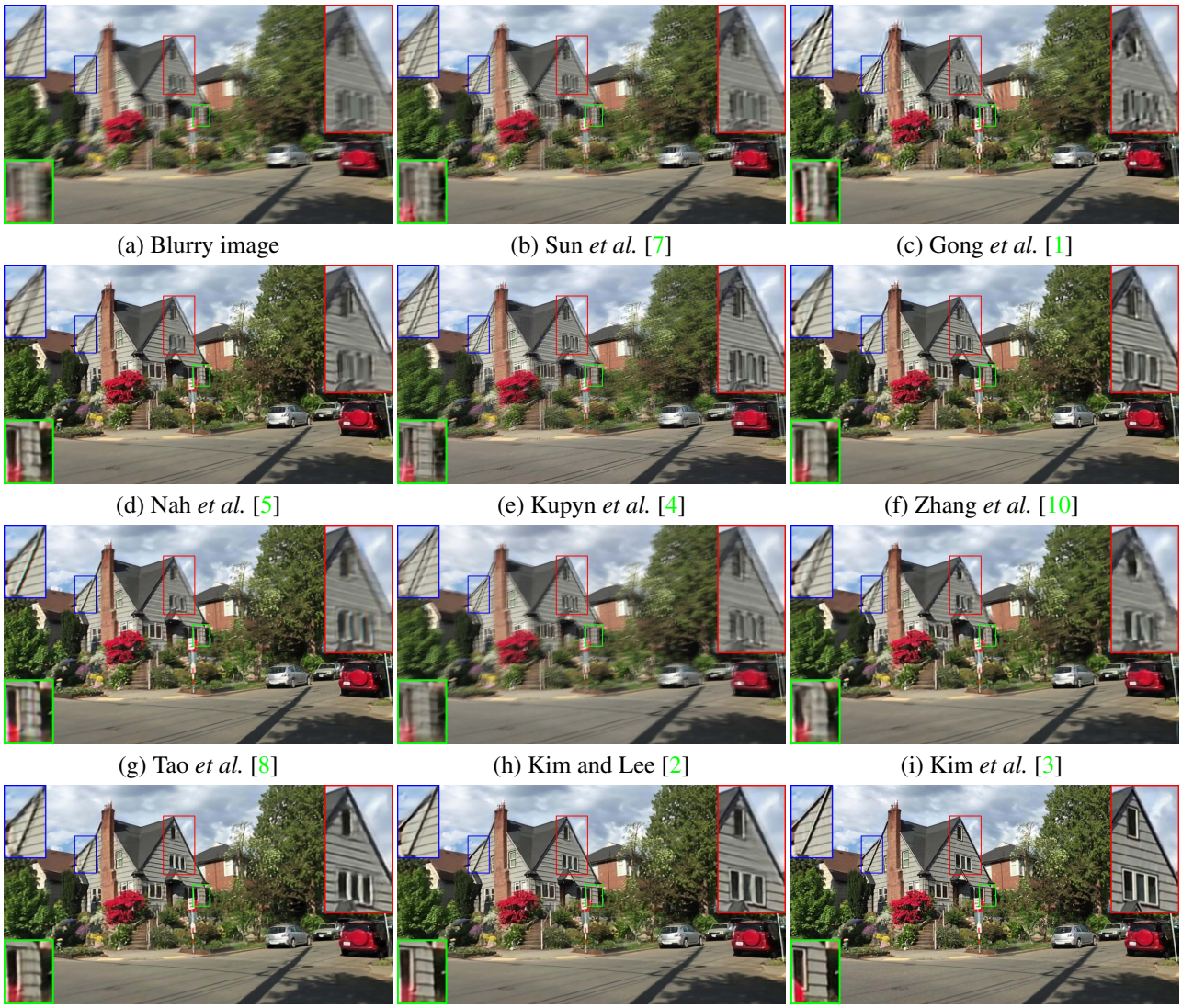
(a) Blurry image (b) Sun *et al.* [7] (c) Gong *et al.* [1]

(d) Nah *et al.* [5] (e) Kupyn *et al.* [4] (f) Zhang *et al.* [10]

(g) Tao *et al.* [8] (h) Kim and Lee [2] (i) Kim *et al.* [3]

(j) Su *et al.* [6] (k) Ours (l) Ground truth

Figure 3: Visual comparisons on video deblurring dataset [6]. Our method generates clearer image.



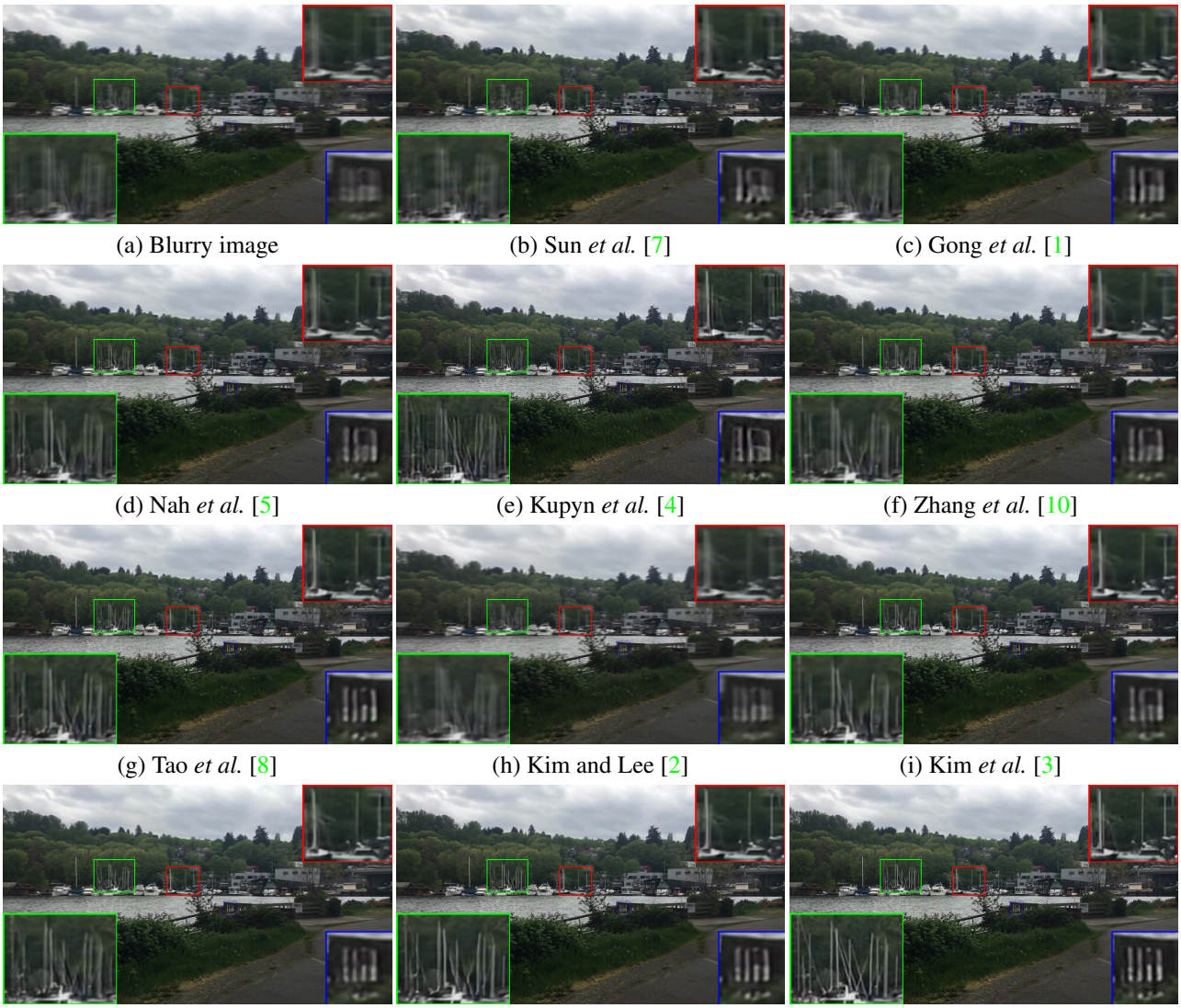
(a) Blurry image (b) Sun *et al.* [7] (c) Gong *et al.* [1]

(d) Nah *et al.* [5] (e) Kupyn *et al.* [4] (f) Zhang *et al.* [10]

(g) Tao *et al.* [8] (h) Kim and Lee [2] (i) Kim *et al.* [3]

(j) Su *et al.* [6] (k) Ours (l) Ground truth

Figure 4: Visual comparisons on video deblurring dataset [6]. Our method generates clearer image.



(a) Blurry image (b) Sun *et al.* [7] (c) Gong *et al.* [1]

(d) Nah *et al.* [5] (e) Kupyn *et al.* [4] (f) Zhang *et al.* [10]

(g) Tao *et al.* [8] (h) Kim and Lee [2] (i) Kim *et al.* [3]

(j) Su *et al.* [6] (k) Ours (l) Ground truth

Figure 5: Visual comparisons on video deblurring dataset [6]. Our method generates clearer image.

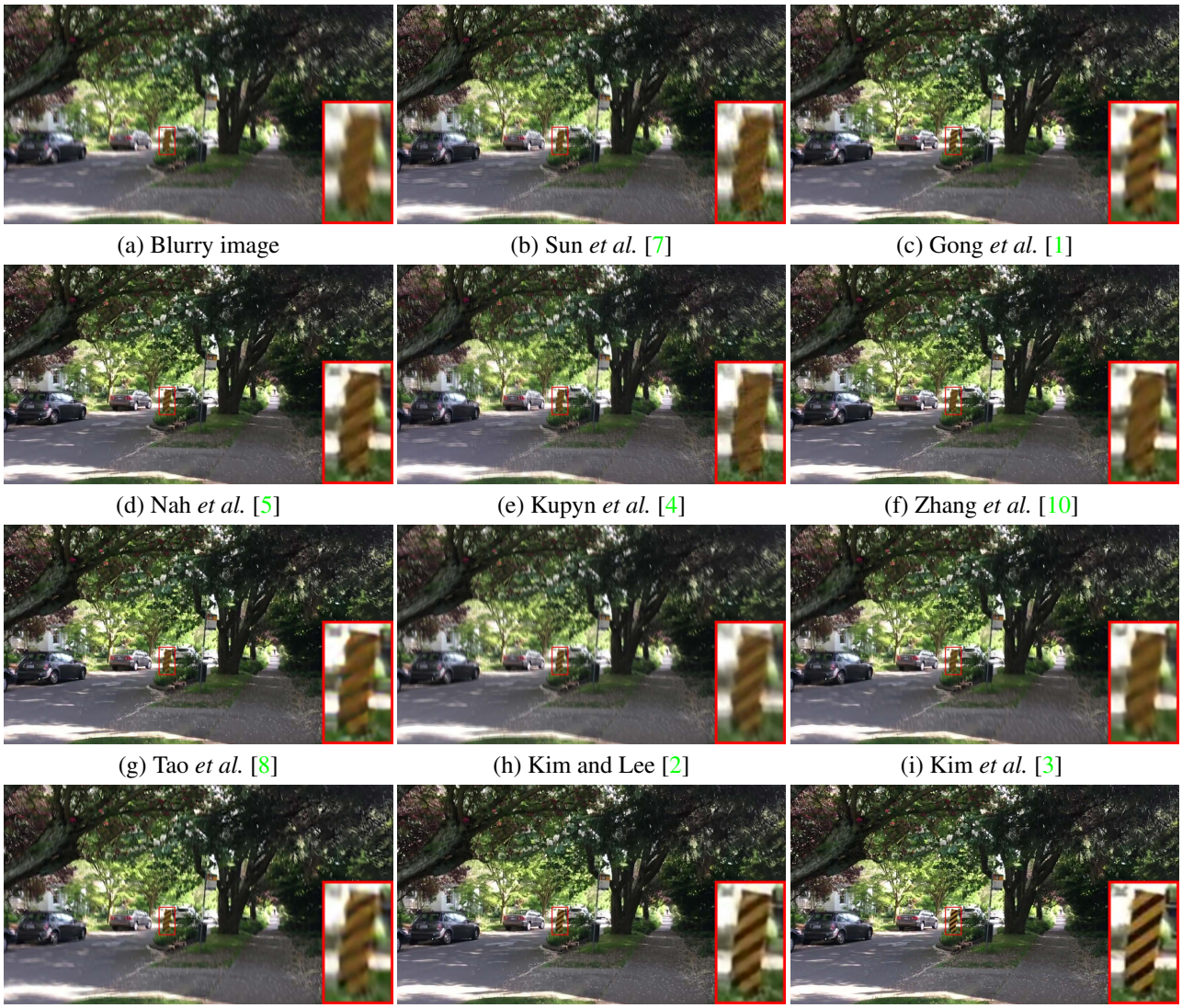
(a) Blurry image (b) Sun *et al.* [7] (c) Gong *et al.* [1]

(d) Nah *et al.* [5] (e) Kupyn *et al.* [4] (f) Zhang *et al.* [10]

(g) Tao *et al.* [8] (h) Kim and Lee [2] (i) Kim *et al.* [3]

(j) Su *et al.* [6] (k) Ours (l) Ground truth

Figure 6: Visual comparisons on video deblurring dataset [6]. Our method generates clearer image.



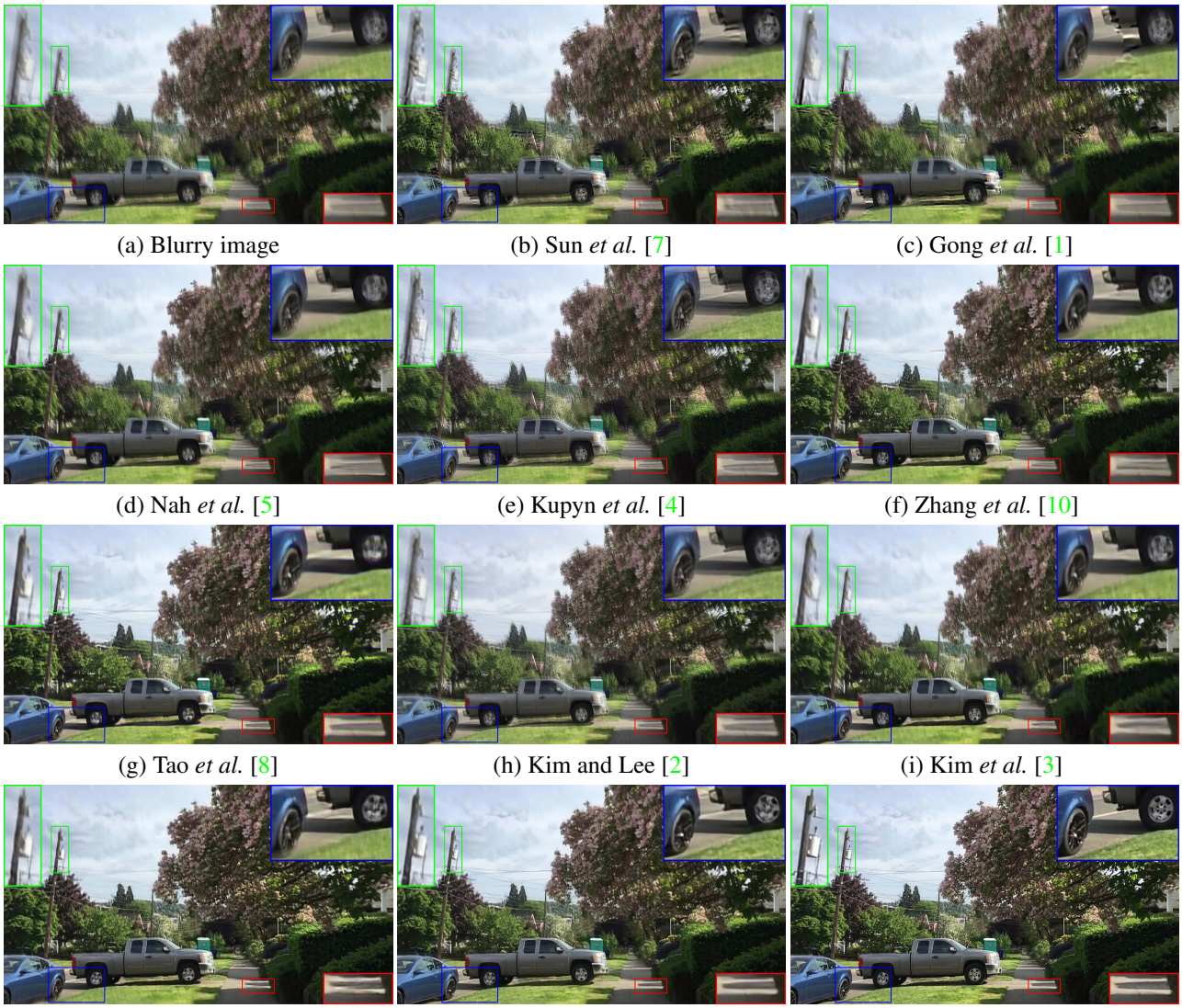
(a) Blurry image (b) Sun *et al.* [7] (c) Gong *et al.* [1]

(d) Nah *et al.* [5] (e) Kupyn *et al.* [4] (f) Zhang *et al.* [10]

(g) Tao *et al.* [8] (h) Kim and Lee [2] (i) Kim *et al.* [3]

(j) Su *et al.* [6] (k) Ours (l) Ground truth

Figure 7: Visual comparisons on video deblurring dataset [6]. Our method generates clearer image.



(a) Blurry image (b) Sun *et al.* [7] (c) Gong *et al.* [1]

(d) Nah *et al.* [5] (e) Kupyn *et al.* [4] (f) Zhang *et al.* [10]

(g) Tao *et al.* [8] (h) Kim and Lee [2] (i) Kim *et al.* [3]

(j) Su *et al.* [6] (k) Ours (l) Ground truth

Figure 8: Visual comparisons on video deblurring dataset [6]. Our method generates clearer image.

(a) Blurry image (b) Sun *et al.* [7] (c) Gong *et al.* [1]

(d) Nah *et al.* [5] (e) Kupyn *et al.* [4] (f) Zhang *et al.* [10]

(g) Tao *et al.* [8] (h) Kim and Lee [2] (i) Kim *et al.* [3]

(j) Su *et al.* [6] (k) Ours (l) Ground truth

Figure 9: Visual comparisons on video deblurring dataset [6]. Our method generates clearer image.

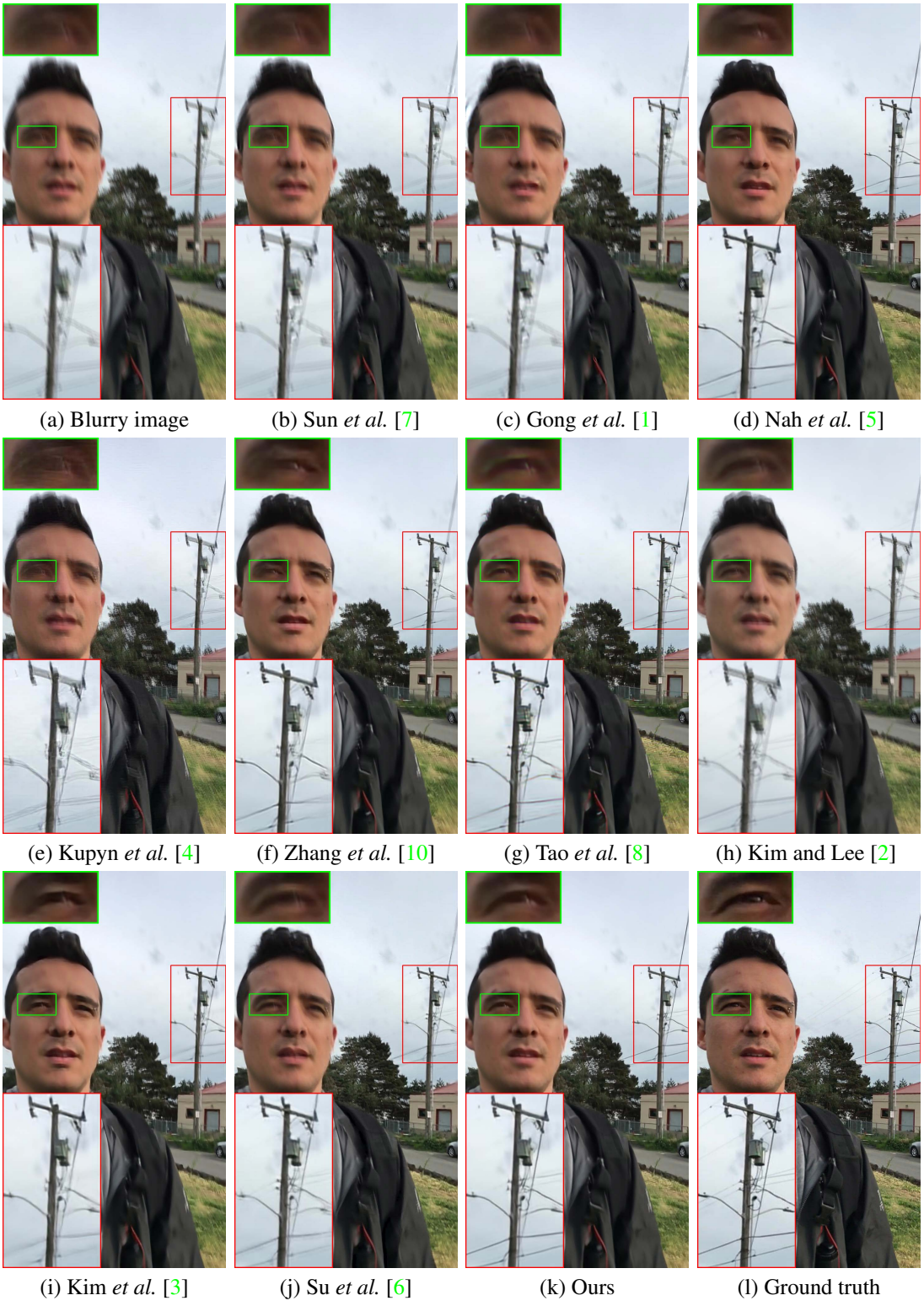
(a) Blurry image (b) Sun *et al.* [7] (c) Gong *et al.* [1]

(d) Nah *et al.* [5] (e) Kupyn *et al.* [4] (f) Zhang *et al.* [10]

(g) Tao *et al.* [8] (h) Kim and Lee [2] (i) Kim *et al.* [3]

(j) Su *et al.* [6] (k) Ours (l) Ground truth

Figure 10: Visual comparisons on video deblurring dataset [6]. Our method generates clearer image.



(a) Blurry image (b) Sun *et al.* [7] (c) Gong *et al.* [1] (d) Nah *et al.* [5]

(e) Kupyn *et al.* [4] (f) Zhang *et al.* [10] (g) Tao *et al.* [8] (h) Kim and Lee [2]

(i) Kim *et al.* [3] (j) Su *et al.* [6] (k) Ours (l) Ground truth

Figure 11: Visual comparisons on video deblurring dataset [6]. Our method generates clearer image.

4.2. Real-world Blurry Videos

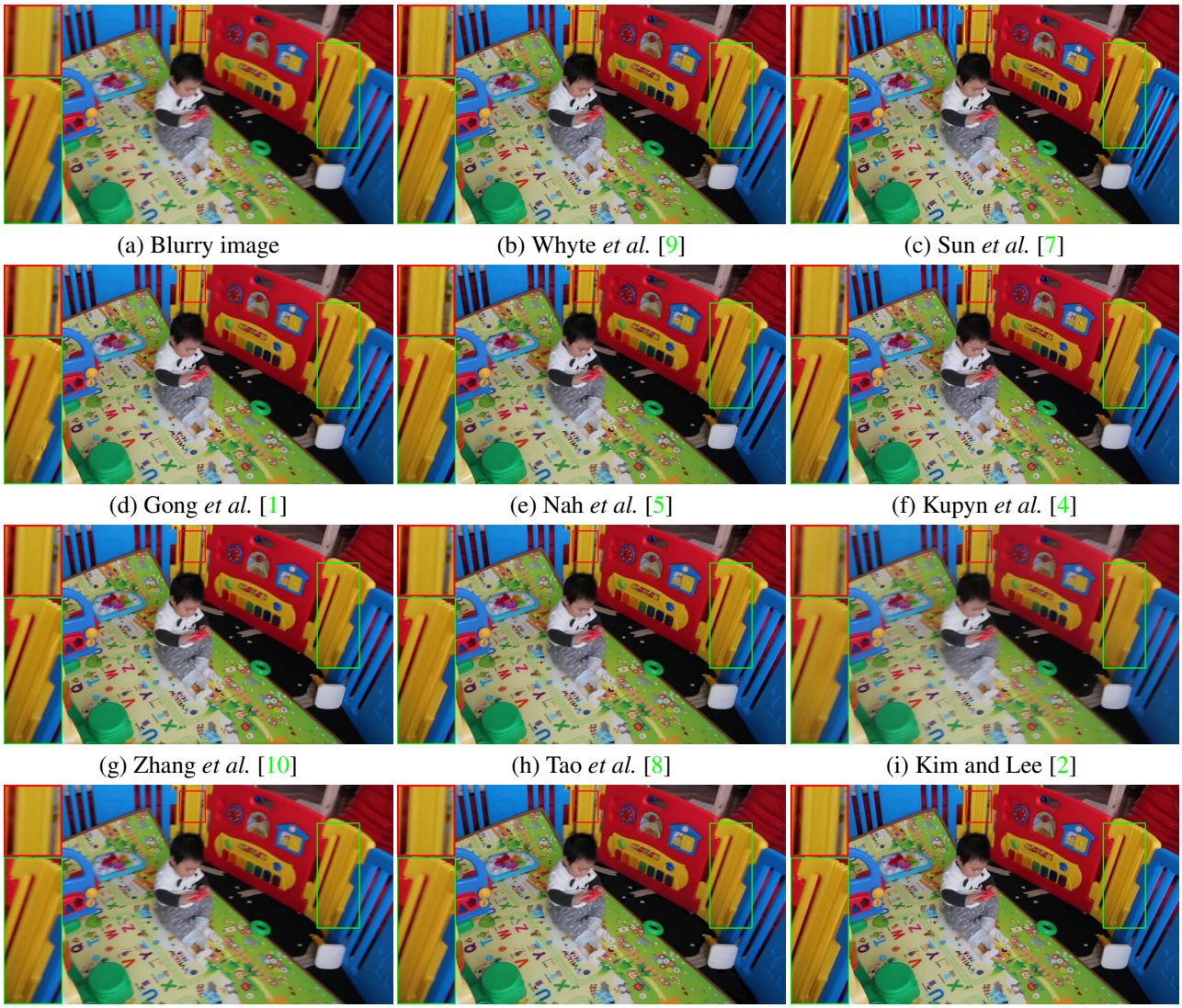
(a) Blurry image (b) Whyte *et al.* [9] (c) Sun *et al.* [7]

(d) Gong *et al.* [1] (e) Nah *et al.* [5] (f) Kupyn *et al.* [4]

(g) Zhang *et al.* [10] (h) Tao *et al.* [8] (i) Kim and Lee [2]

(j) Kim *et al.* [3] (k) Su *et al.* [6] (i) Ours

Figure 12: Visual comparisons on real-world blurry videos. Our method generates clearer image.



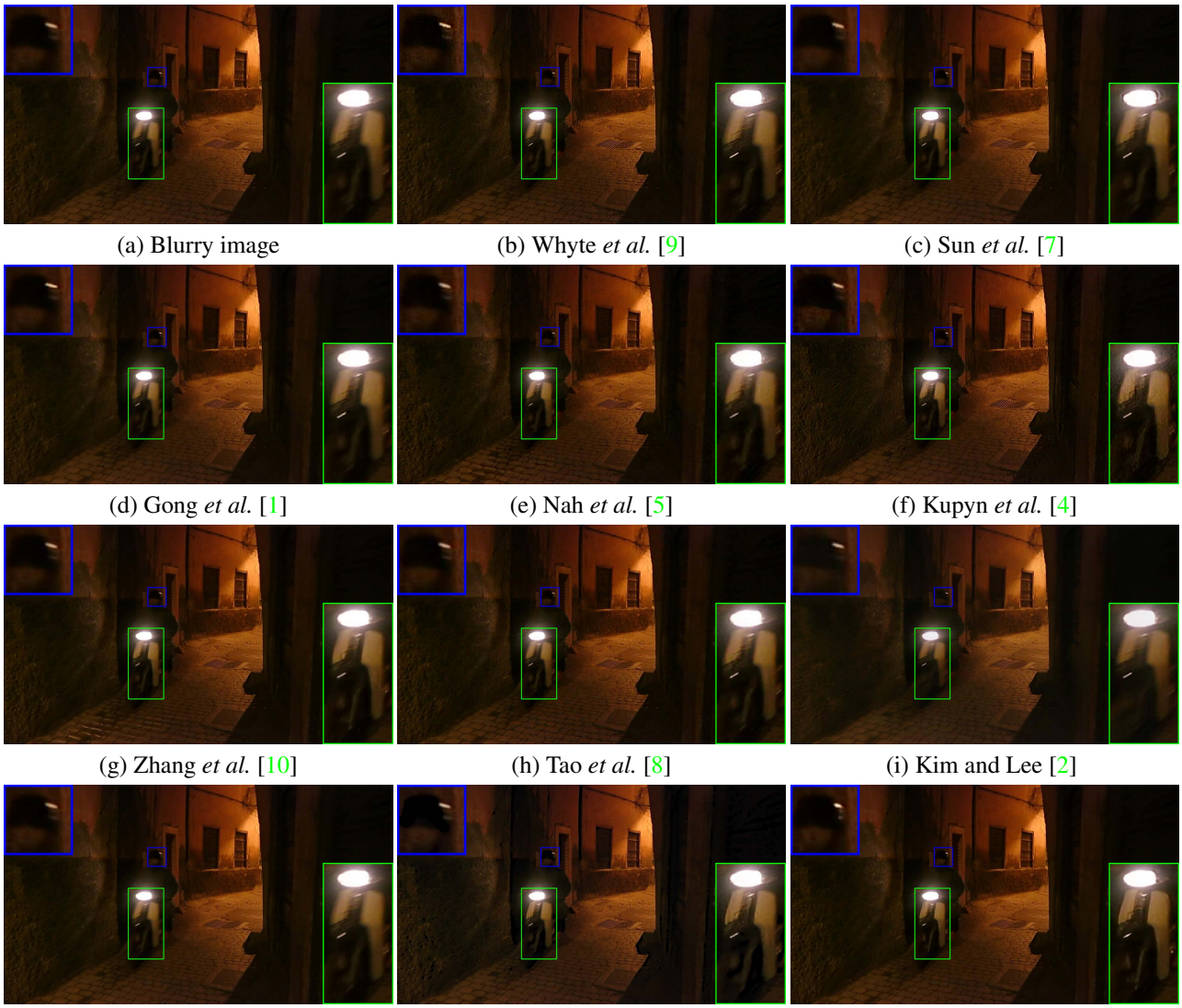
(a) Blurry image (b) Whyte *et al.* [9] (c) Sun *et al.* [7]

(d) Gong *et al.* [1] (e) Nah *et al.* [5] (f) Kupyn *et al.* [4]

(g) Zhang *et al.* [10] (h) Tao *et al.* [8] (i) Kim and Lee [2]

(j) Kim *et al.* [3] (k) Su *et al.* [6] (i) Ours

Figure 13: Visual comparisons on real-world blurry videos. Our method generates clearer image.



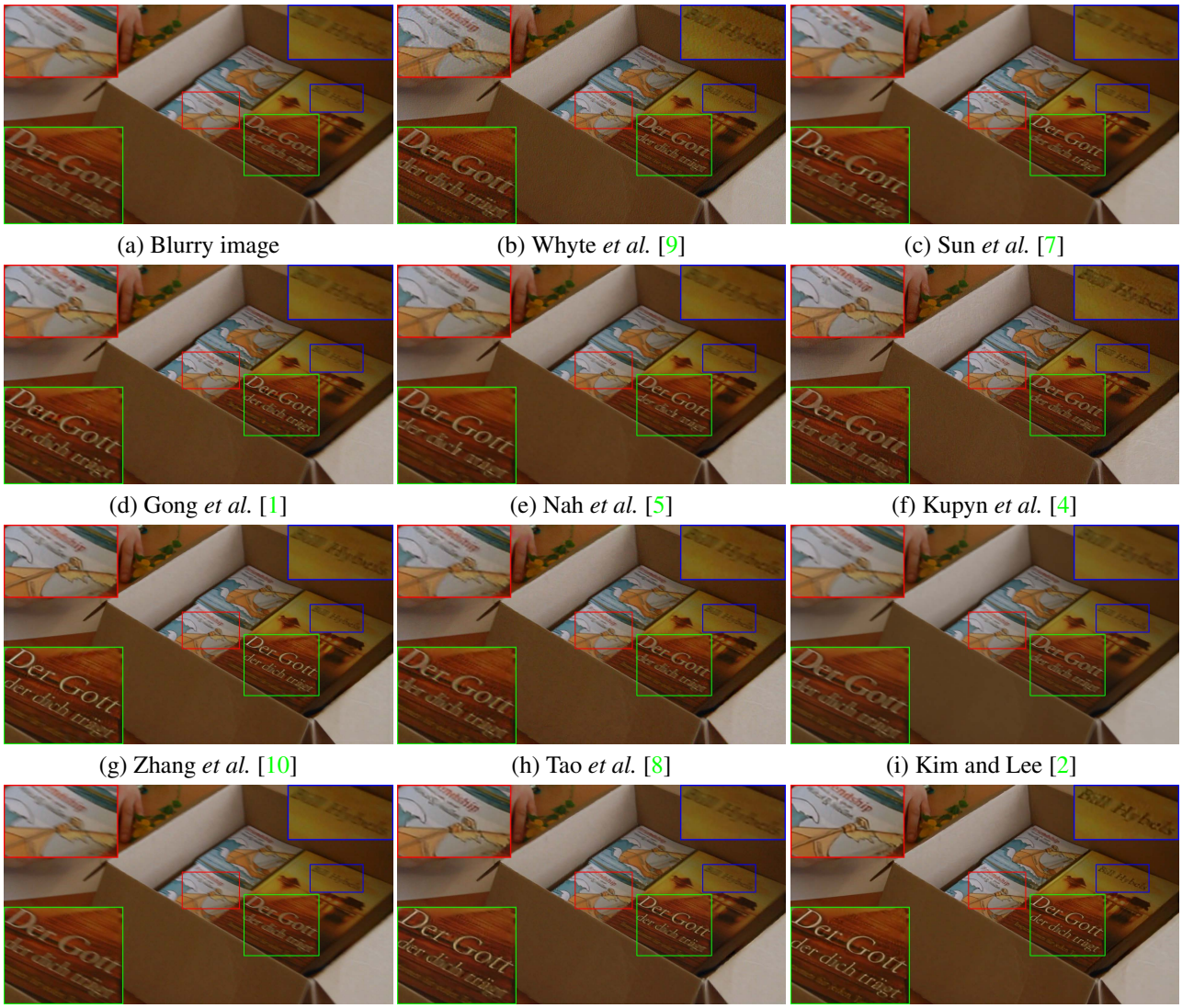
(a) Blurry image (b) Whyte *et al.* [9] (c) Sun *et al.* [7]

(d) Gong *et al.* [1] (e) Nah *et al.* [5] (f) Kupyn *et al.* [4]

(g) Zhang *et al.* [10] (h) Tao *et al.* [8] (i) Kim and Lee [2]

(j) Kim *et al.* [3] (k) Su *et al.* [6] (i) Ours

Figure 14: Visual comparisons on real-world blurry videos. Our method generates clearer image.



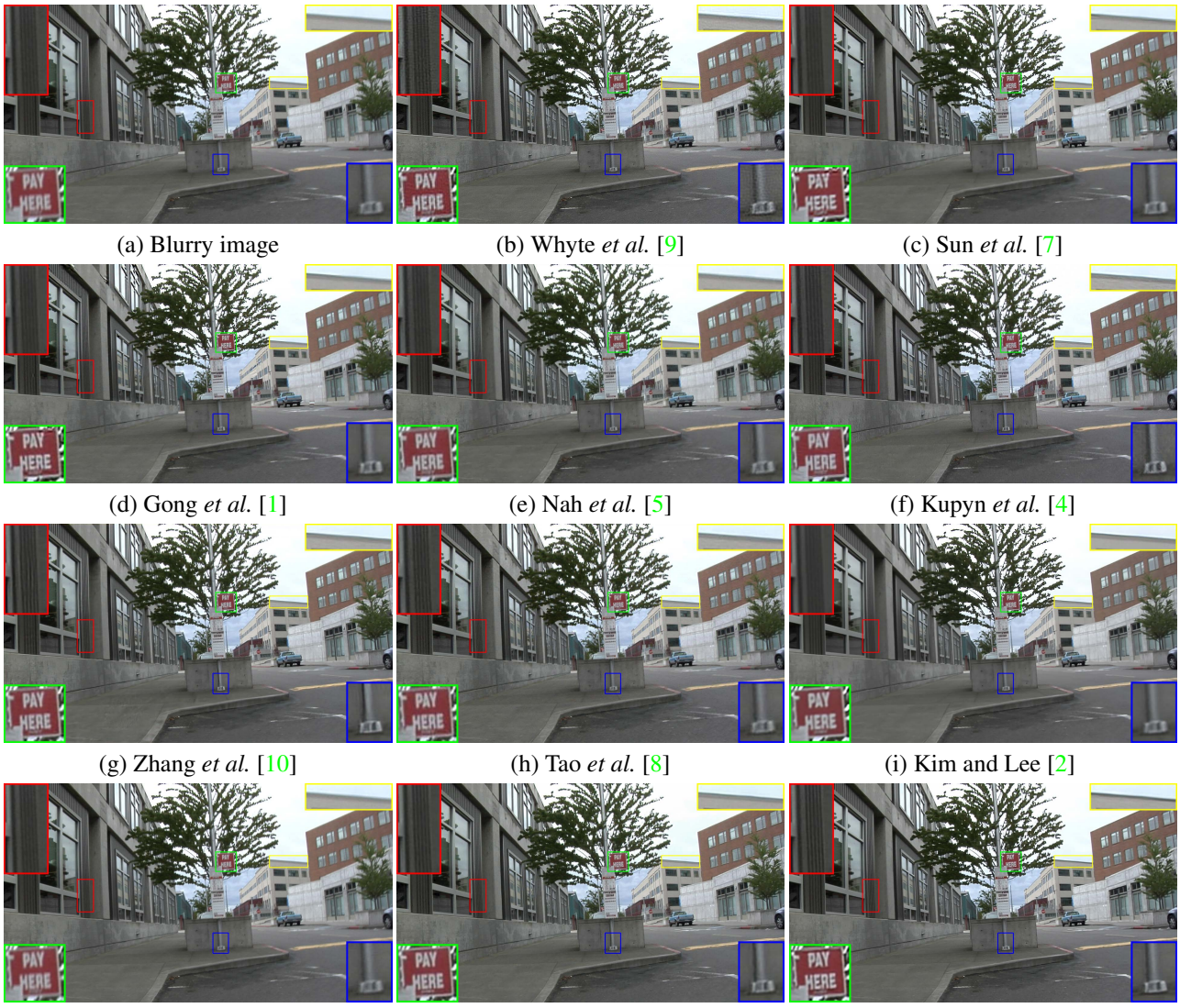
(a) Blurry image (b) Whyte *et al.* [9] (c) Sun *et al.* [7]

(d) Gong *et al.* [1] (e) Nah *et al.* [5] (f) Kupyn *et al.* [4]

(g) Zhang *et al.* [10] (h) Tao *et al.* [8] (i) Kim and Lee [2]

(j) Kim *et al.* [3] (k) Su *et al.* [6] (i) Ours

Figure 15: Visual comparisons on real-world blurry videos. Our method generates clearer image.



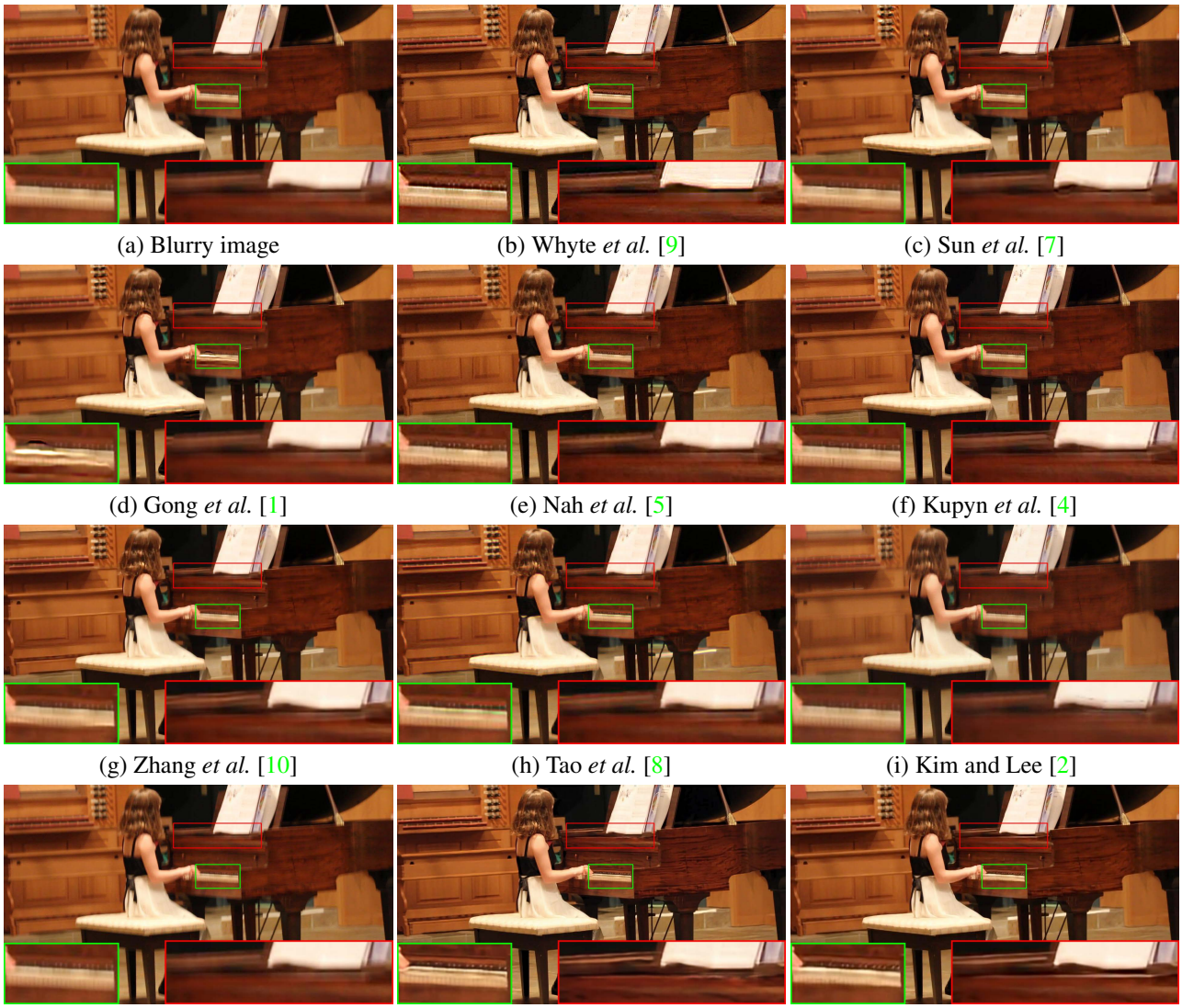
(a) Blurry image (b) Whyte *et al.* [9] (c) Sun *et al.* [7]

(d) Gong *et al.* [1] (e) Nah *et al.* [5] (f) Kupyn *et al.* [4]

(g) Zhang *et al.* [10] (h) Tao *et al.* [8] (i) Kim and Lee [2]

(j) Kim *et al.* [3] (k) Su *et al.* [6] (i) Ours

Figure 16: Visual comparisons on real-world blurry videos. Our method generates clearer image.



(a) Blurry image (b) Whyte *et al.* [9] (c) Sun *et al.* [7]

(d) Gong *et al.* [1] (e) Nah *et al.* [5] (f) Kupyn *et al.* [4]

(g) Zhang *et al.* [10] (h) Tao *et al.* [8] (i) Kim and Lee [2]

(j) Kim *et al.* [3] (k) Su *et al.* [6] (i) Ours

Figure 17: Visual comparisons on real-world blurry videos. Our method generates clearer image.

(a) Blurry image (b) Whyte *et al.* [9] (c) Sun *et al.* [7]

(d) Gong *et al.* [1] (e) Nah *et al.* [5] (f) Kupyn *et al.* [4]

(g) Zhang *et al.* [10] (h) Tao *et al.* [8] (i) Kim and Lee [2]

(j) Kim *et al.* [3] (k) Su *et al.* [6] (i) Ours

Figure 18: Visual comparisons on real-world blurry videos. Our method generates clearer image.

References

[1] D. Gong, J. Yang, L. Liu, Y. Zhang, I. D. Reid, C. Shen, A. Van Den Hengel, and Q. Shi. From motion blur to motion flow: A deep learning solution for removing heterogeneous motion blur. In *CVPR*, 2017.

[2] T. Hyun Kim and K. Mu Lee. Generalized video deblurring for dynamic scenes. In *CVPR*, 2015.

[3] T. Hyun Kim, K. Mu Lee, B. Scholkopf, and M. Hirsch. Online video deblurring via dynamic temporal blending network. In *CVPR*, 2017.

[4] O. Kupyn, V. Budzan, M. Mykhailych, D. Mishkin, and J. Matas. Deblurgan: Blind motion deblurring using conditional adversarial networks. In *CVPR*, 2018.

[5] S. Nah, T. H. Kim, and K. M. Lee. Deep multi-scale convolutional neural network for dynamic scene deblurring. In *CVPR*, 2017.

[6] S. Su, M. Delbracio, J. Wang, G. Sapiro, W. Heidrich, and O. Wang. Deep video deblurring for hand-held cameras. In *CVPR*, 2017.

[7] J. Sun, W. Cao, Z. Xu, and J. Ponce. Learning a convolutional neural network for non-uniform motion blur removal. In *CVPR*, 2015.

[8] X. Tao, H. Gao, X. Shen, J. Wang, and J. Jia. Scale-recurrent network for deep image deblurring. In *CVPR*, 2018.

[9] O. Whyte, J. Sivic, A. Zisserman, and J. Ponce. Non-uniform deblurring for shaken images. *IJCV*, 98(2):168–186, 2012.

[10] J. Zhang, J. Pan, J. Ren, Y. Song, L. Bao, R. W. Lau, and M.-H. Yang. Dynamic scene deblurring using spatially variant recurrent neural networks. In *CVPR*, 2018.