Recurrent Neural Networks with Intra-Frame Iterations for Video Deblurring

Seungjun Nah Sanghyun Son Kyoung Mu Lee

Department of ECE, ASRI, Seoul National University, Seoul, Korea

seungjun.nah@gmail.com, {thstkdgus35, kyoungmu}@snu.ac.kr

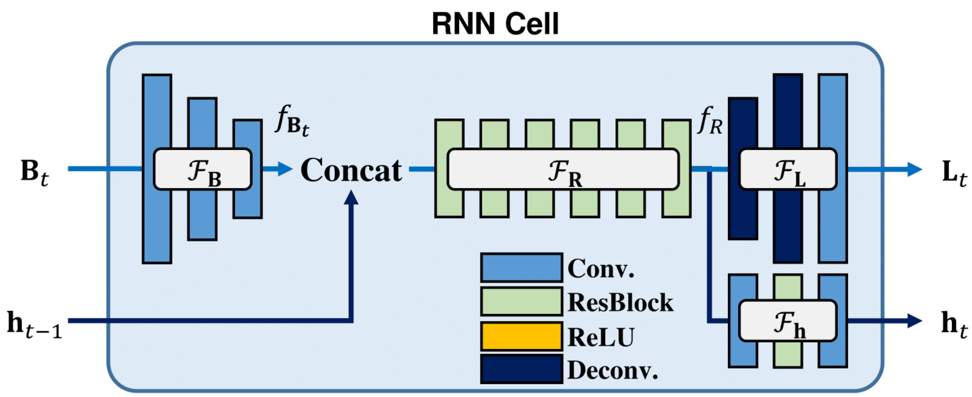


Figure 1: The IFI-RNN cell architecture

1. Introduction

In the main submission manuscript, We proposed IFI-RNN models that use an iterative approach to update hidden states of RNNs. In this supplementary material, we provide more detailed architecture specification. Also, we display more experimental results and comparisons and comparisons with other state-of-the-art video deblurring methods [4, 5, 1].

2. IFI-RNN Model Specification

In Fig. 1, we again show our model cell architecture. In this section, we describe the specific parameter details. Our IFI-RNN cell consists of 4 parts: $\mathcal{F}_\mathbf{B}$, $\mathcal{F}_\mathbf{R}$, $\mathcal{F}_\mathbf{L}$, and $\mathcal{F}_\mathbf{h}$.

The blur feature extraction part, $\mathcal{F}_\mathbf{B}$, contains three convolutional layers without nonlinear activations. It reduces the spatial resolution and effectively increases the receptive field. Given an RGB input size $h \times w$, it produces feature size $60 \times h/4 \times w/4$. Then, the extracted feature is concatenated with a hidden state of size $20 \times h/4 \times w/4$, producing a tensor of shape $80 \times h/4 \times w/4$.

$\mathcal{F}_\mathbf{R}$ is a sequence of the following 6 residual blocks. It generates a feature map having the same size as its input, $80 \times h/4 \times w/4$. Each resblock consists of two convolutional layers with ReLU activation in between. Note that there are no batch-normalization layers in the ResBlocks, following previous image deblurring [3] and super-resolution [2] models.

With the calculated feature from $\mathcal{F}_\mathbf{R}$, $\mathcal{F}_\mathbf{h}$ and $\mathcal{F}_\mathbf{L}$ are located in parallel. Each of them outputs the hidden state and the latent deblurred frame. $\mathcal{F}_\mathbf{h}$ has 2 convolutional layers with a single ResBlock in between. It preserves the resolution of the feature to generate the hidden state of the next time-step. On the other hand, $\mathcal{F}_\mathbf{L}$ increases the resolution and reduces the channels to reconstruct a deblurred frame. It uses a convolution layer after two up-convolutions. In Table 1, we explain the exact kernels and outputs of each layer.

Thus, our single cell method has 0.84M parameters. For dual cell method, we do not need $\mathcal{F}_\mathbf{L}$ for the 1st cell, as we estimate the latent image in the 2nd cell only. Hence, our dual cell method only requires storage for 1.64M parameters. We compare the model size with state-of-the-art methods in Table 2.

Module	layer	kernel	stride	output size	parameters
	input	-	-	$3 \times h \times w$	-
$\mathcal{F}_{\mathbf{B}}$	conv	5×5	1	$20 \times h \times w$	1520
	conv	5×5	2	$40 \times h/2 \times w/2$	20040
	conv	5×5	2	$60 \times h/4 \times w/4$	60060
	hidden state	-	-	$20 \times h/4 \times w/4$	-
	concat	-	-	$80 \times h/4 \times w/4$	-
$\mathcal{F}_{\mathbf{R}}$	ResBlock $\times 6$	3×3	1	$80 \times h/4 \times w/4$	692160
$\mathcal{F}_{\mathbf{h}}$	conv	3×3	1	$20 \times h/4 \times w/4$	14420
	ResBlock	3×3	1	$20 \times h/4 \times w/4$	7240
	conv	3×3	1	$20 \times h/4 \times w/4$	3620
$\mathcal{F}_{\mathbf{L}}$	up-conv	3×3	2	$40 \times h/2 \times w/2$	28840
	up-conv	3×3	2	$20 \times h \times w$	7220
	conv	5×5	1	$3 \times h \times w$	1503

Table 1: Our IFI-RNN cell architecture details. Each component details are shown. $\mathcal{F}_{\mathbf{B}}$, $\mathcal{F}_{\mathbf{R}}$, $\mathcal{F}_{\mathbf{L}}$, and $\mathcal{F}_{\mathbf{h}}$ each has 0.08M, 0.69M, 0.03M, 0.04M parameters, respectively. There are total 0.84M parameters in our single cell model.

Model	# parameters	Storage (MB)
DBN [4]	15.3M	58.4
RDN [5]	16.4M	62.6
OVD [1]	0.90M	3.4
IFI-RNN (single cell)	0.84M	3.2
IFI-RNN (dual cell)	1.64M	6.2

Table 2: Model size comparison with other deep learning based methods. For RDN, we refer to the model and source code provided by the authors of [5], which is different from the paper. In the main paper and this supplementary material, the comparisons are consistently done with the provided model.

3. More Visual Comparisons

In this section, we provide more visual comparisons of deblurred results. We used real blurry videos from YouTube and captured by ourselves. In many examples, our IFI-RNN shows sharper reconstruction results especially on the fine textured area. We notice better reconstructed faces in Fig. 2, 3, 5, 6. Also, our method shows lesser artifact in recovering more easily readable text in Fig. 2, 4, 5. Fine-grained textures are more recognizable in Fig. 3, 7.

References

[1] T. H. Kim, K. M. Lee, B. Scholkopf, and M. Hirsch. Online video deblurring via dynamic temporal blending network. In *ICCV*, 2017. 1, 2, 3, 4, 5

[2] B. Lim, S. Son, H. Kim, S. Nah, and K. M. Lee. Enhanced deep residual networks for single image super-resolution. In *CVPR Workshops*, 2017. 1

[3] S. Nah, T. H. Kim, and K. M. Lee. Deep multi-scale convolutional neural network for dynamic scene deblurring. In *CVPR*, 2017. 1

[4] S. Su, M. Delbracio, J. Wang, G. Sapiro, W. Heidrich, and O. Wang. Deep video deblurring for hand-held cameras. In *CVPR*, 2017. 1, 2, 3, 4, 5

[5] P. Wieschollek, M. Hirsch, B. Scholkopf, and H. P. A. Lensch. Learning blind motion deblurring. In *ICCV*, 2017. 1, 2, 3, 4, 5

(a) Blur

(b) Deblurred (Ours)

(c) Blur

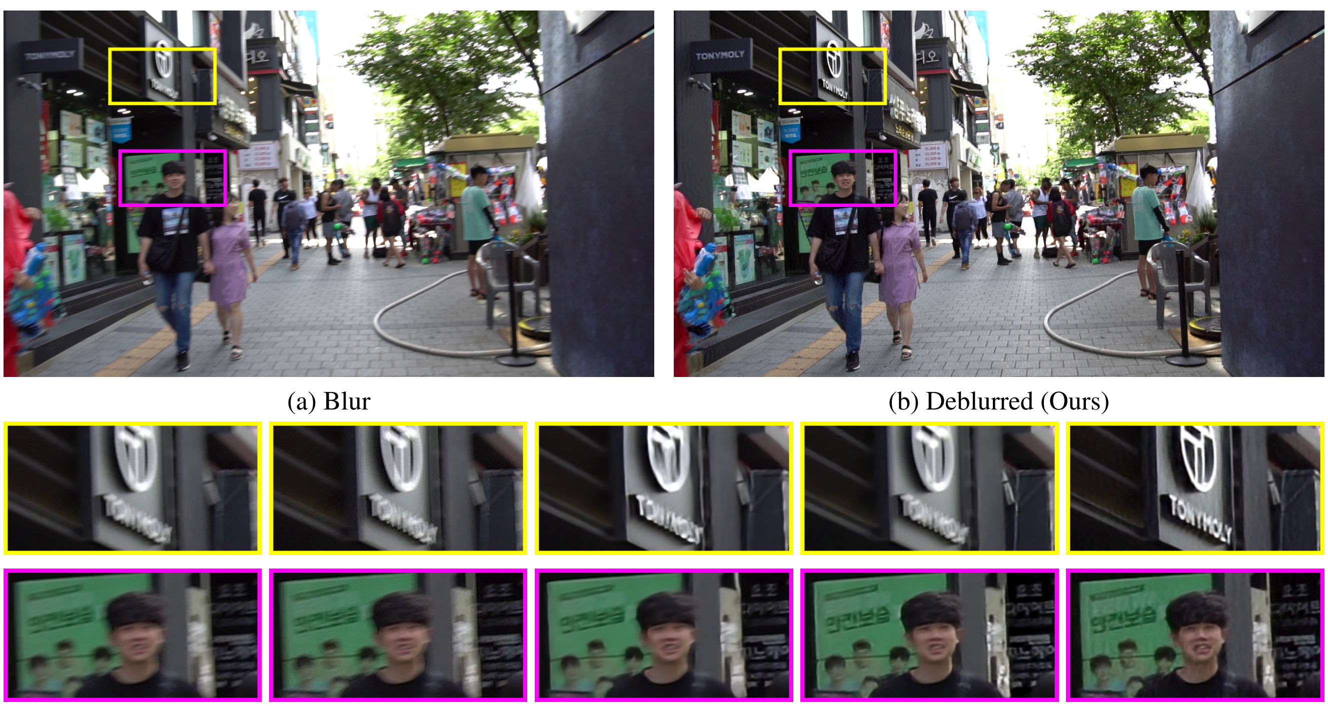
(d) RDN [5]

(e) OVD [1]

(f) DBN+OF [4]

(g) IFI-RNN(C2H3-reg)

Figure 2: Deblurring results of real video.



(a) Blur

(b) Deblurred (Ours)

(c) Blur

(d) RDN [5]

(e) OVD [1]

(f) DBN+OF [4]

(g) IFI-RNN(C2H3-reg)

Figure 3: Deblurring results of real video.

(a) Blur (b) Deblurred (Ours)

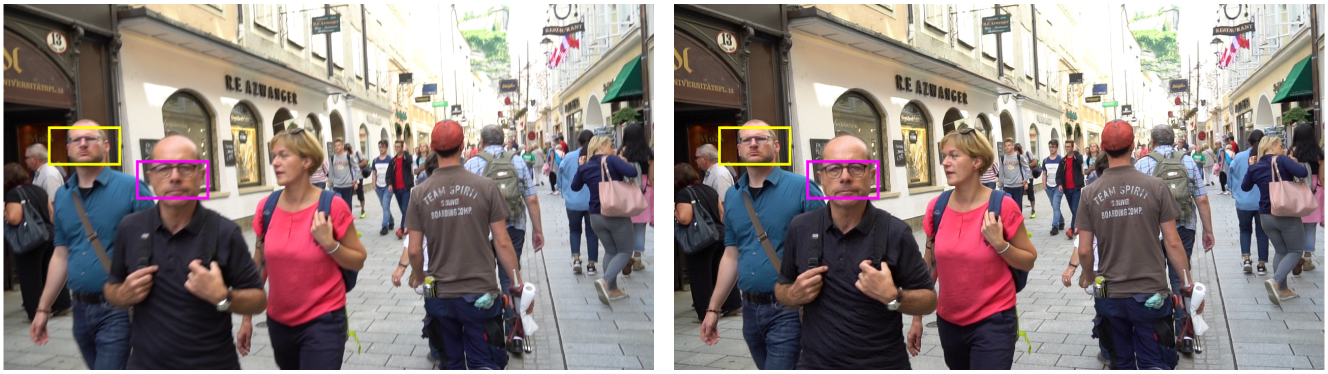
(c) Blur (d) RDN [5] (e) OVD [1] (f) DBN+OF [4] (g) IFI-RNN(C2H3-reg)

Figure 4: Deblurring results of real video.

(a) Blur (b) Deblurred (Ours)

(c) Blur (d) RDN [5] (e) OVD [1] (f) DBN+OF [4] (g) IFI-RNN(C2H3-reg)

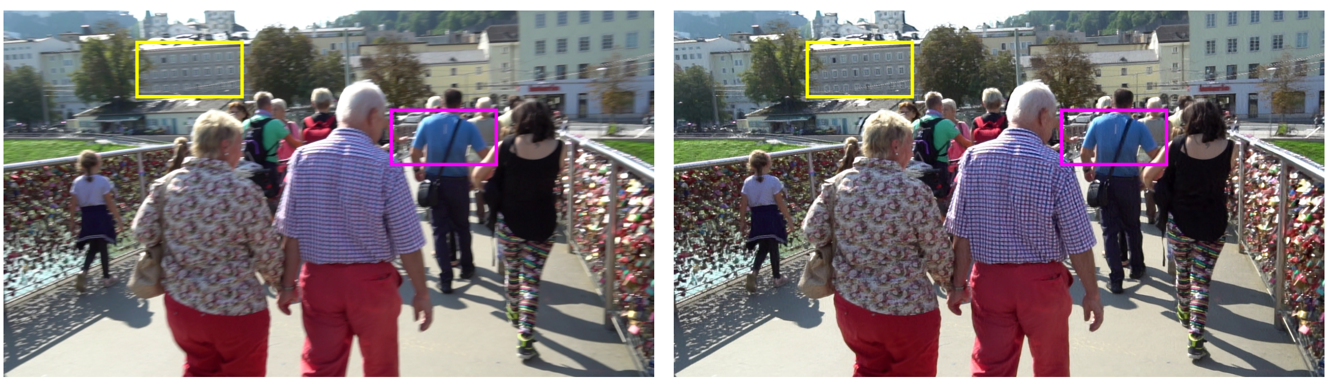
Figure 5: Deblurring results of real video.



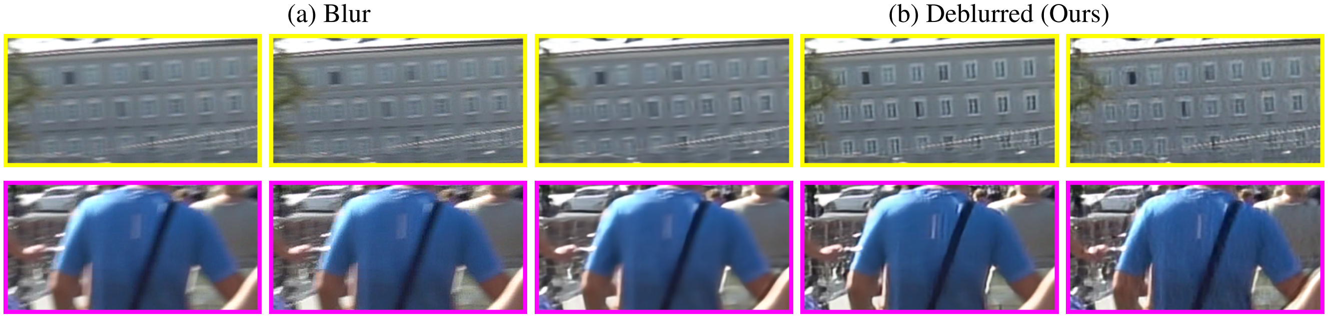
(a) Blur | (b) Deblurred (Ours)

(c) Blur | (d) RDN [5] | (e) OVD [1] | (f) DBN+OF [4] | (g) IFI-RNN(C2H3-reg)

Figure 6: Deblurring results of real video.



(a) Blur | (b) Deblurred (Ours)



(c) Blur | (d) RDN [5] | (e) OVD [1] | (f) DBN+OF [4] | (g) IFI-RNN(C2H3-reg)

Figure 7: Deblurring results of real video.