Space-Time Distillation for Video Super-Resolution
—— Supplementary Material ——

Zeyu Xiao Xueyang Fu Jie Huang Zhen Cheng Zhiwei Xiong
University of Science and Technology of China

This supplementary document is organized as follows:
Sec. 1 provides the details of the ConvLSTM.
Sec. 2 provides the details of the FastDVDnet.
Sec. 3 provides quantitative comparisons in terms of SSIM and more qualitative results.

1. Details of ConvLSTM

The input sequence of the ConvLSTM consists of the feature maps $F^{SR}_{[t-k:t+k]}$ for each frame. For each time step, the key equations are shown below (the feature maps of student and teacher can be represented by adding `S` and `T` subscripts respectively, and we omit these subscripts here):

$$\begin{aligned}
\mathbf{i}_t =& \sigma\left(\mathbf{W}_{fi} * F^{SR}_t + \mathbf{W}_{hi} * \mathbf{H}_{t-1} + \mathbf{W}_{ci} \circ \mathbf{C}_{t-1} + \mathbf{b}_i\right), \\
\mathbf{f}_t =& \sigma\left(\mathbf{W}_{ff} * F^{SR}_t + \mathbf{W}_{hf} * \mathbf{H}_{t-1} + \mathbf{W}_{cf} \circ \mathbf{C}_{t-1} + \mathbf{b}_f\right), \\
\mathbf{C}_t =& \mathbf{f}_t \circ \mathbf{C}_{t-1} + \mathbf{i}_t \circ \tanh\left(\mathbf{W}_{fc} * F^{SR}_t + \mathbf{W}_{hc} * \mathbf{H}_{t-1} + \mathbf{b}_c\right), \\
\mathbf{o}_t =& \sigma\left(\mathbf{W}_{fo} * F^{SR}_t + \mathbf{W}_{ho} * \mathbf{H}_{t-1} + \mathbf{W}_{ho} \circ \mathbf{C}_t + \mathbf{b}_o\right), \\
\mathbf{H}_t =& \mathbf{o}_t \circ \tanh\left(\mathbf{C}_t\right),
\end{aligned} \tag{1}$$

where $\circ$ denotes the Hadamard product, $*$ denotes the convolution operator, σ is the sigmoid activation function and the activation of input gate i_t controls whether the new input of the current time step will be engaged in the memory cell. $\mathbf{f}_t$ controls how much information will be kept from the past status $\mathbf{C}_{t-1}$. $\mathbf{o}_t$ decides the propagation from $\mathbf{C}_t$ to the hidden state $\mathbf{H}_t$. $\mathbf{W}_*$ and $\mathbf{b}_*$ denote the trainable parameters of the kernels and bias at the corresponding convolution layers. We employ the memory state of the final time step as the distillation item, which contains temporal information.

We also investigate the impact of initializing the ConvLSTM in time distillation (TD) by: (a) Randomly initialize the ConvLSTM and optimize its parameters during training; (b) pretrain a ConvLSTM for the task of video prediction and then freeze its parameters during training; and (c) use the pretrained ConvLSTM in (b) and continue to update its parameters during training. The results are shown in Table 1 and it is clear that the ConvLSTM in TD is not sensitive to initialization. The ConvLSTM is designed for modeling long-term temporal correspondence, which plays an important role in capturing temporal information in TD. Whether it is initialized randomly or not, the information extracted by the ConvLSTM is not optimal. Only when we optimize the parameters of the student network and the ConvLSTM synchronously, the ConvLSTM can extract features suitable for VSR.

Table 1: Analysis on the impact of initializing the ConvLSTM.

w/ TD	(a) Random	(b) Pretrain+freeze	(c) Pretrain+optimize
Vid4-Average	25.87	25.79	25.86

2. Details of FastDVDnet

FastDVDnet [5] is originally proposed for video denoising in which five adjacent noisy inputs are used to reconstruct the intermediate clear frame. Our experiment in Sec. 4 of the paper is mainly based on FastDVDnet. We change its structure to make it suitable for the VSR task.

The architecture of FastDVDnet used in our experiment is shown in Fig. 1. Given seven consecutive video frames $I_{[t-3:t+3]}$, we aim to super-resolve the intermediate frame and get the result SR_t. (I_{t-3}, I_t, I_{t+3}), (I_{t-2}, I_t, I_{t+2}) and (I_{t-1}, I_t, I_{t+1}) are respectively sent to the three subnets which share weights. Then, the output feature maps of the three branches are input into the following subnet after concating operation, and the reconstructed result is obtained after the pixel-shuffle [4] operation with the convolutional layer. The number of channels is the same as the original FastDVDnet.

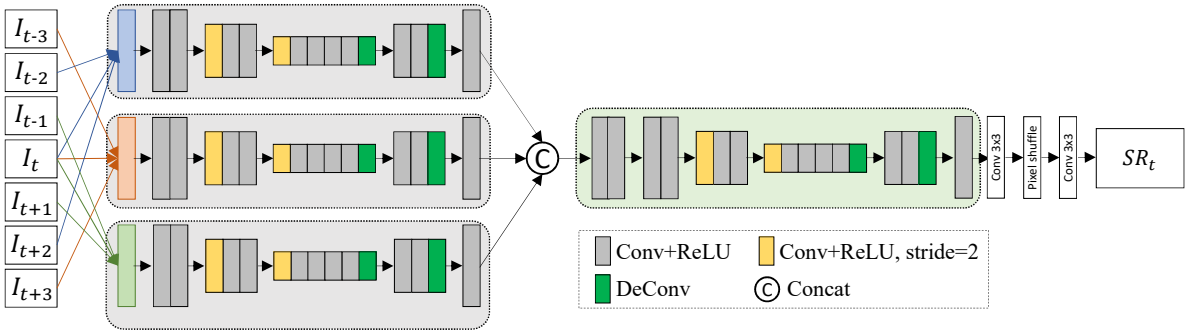


Figure 1: The architecture of FastDVDnet used for VSR. Seven consecutive frames are used to reconstruct the intermediate frame. The skip-connections are omitted in the figure.

We use FastDVDnet as our main compact VSR network, the reasons are two-fold:

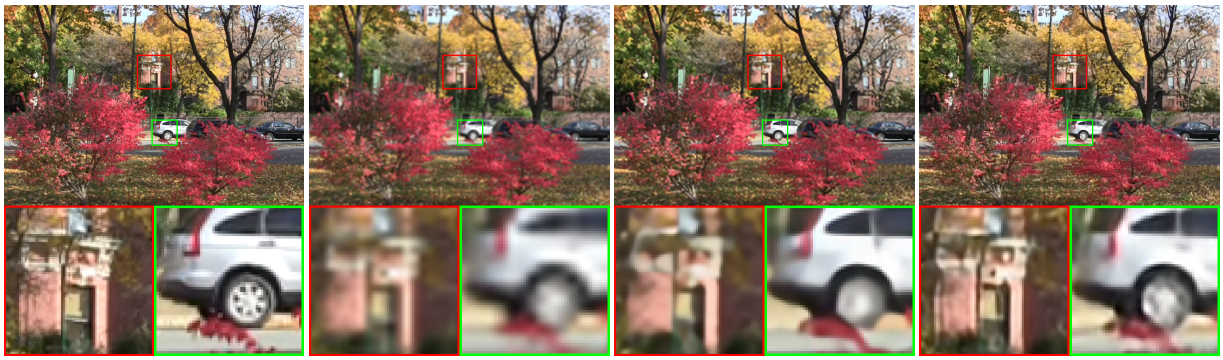
- To the best of our knowledge, the VSR methods published in top venues in the past two years mainly focused on improving SR performance (*i.e.*, PSNR). These methods CANNOT be regarded as "really compact" student networks.
- The recent FastDVDnet (CVPR 2020) has been proven effective and lightweight on video denoising. With simple modification, it outperforms existing compact VSR networks such as VSRNet [2] and VESPCN [1]. Therefore, we adopt it as a representative student network in our experiments.

3. More Experimental Results

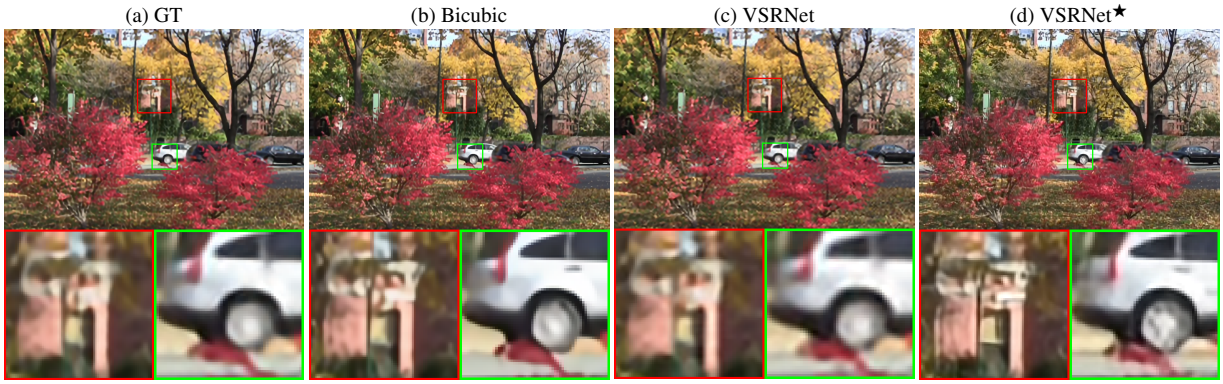
In addition to the PSNR (dB) metric discussed in the paper, we further use SSIM metric for evaluation in Table 2. We also show more visual results in Fig. 2 and Fig. 3.

Table 2: Quantitative comparisons of different methods on Vid4 and Vimeo90K-Test for $4\times$ upscaling in terms of SSIM. Results are evaluated on the Y (luminance) channel. 'Frames' means the number of input frames of the network. 'FLOPs' (T, 10^{12}) is calculated on a frame with the spatial resolution of 180×120. 'Time' is the average running time (ms) which is measured on Vid4-Walk in a per-frame manner. ★ means the student network is trained with our STD scheme and ♣ means the student network is trained with the scheme proposed in [3].

Method	Frames	Network performance		Vid4					Vimeo			
		FLOPs	Time	Calendar	City	Foliage	Walk	Average	Fast	Medium	Slow	Average
Bicubic	1	-	-	0.5720	0.6028	0.5666	0.7974	0.6347	0.8930	0.8592	0.8212	0.8568
TOFlow	7	0.81	632.0	0.7273	0.7446	0.7118	0.8799	0.7659	0.9420	0.9250	0.8890	0.9202
VSR-DUF	7	0.62	496.0	0.8110	0.8235	0.7709	0.9141	0.8318	0.9490	0.9430	0.9090	0.9369
EDVR	7	0.93	86.0	0.8056	0.8313	0.7695	0.9077	0.8286	0.9602	0.9477	0.9192	0.9438
VDSR	1	0.22	11.5	0.6351	0.6473	0.6250	0.8400	0.6869	0.9233	0.8969	0.8647	0.8945
VDSR♣	1	0.22	11.5	0.6501	0.6623	0.6406	0.8517	0.7012	0.9223	0.8969	0.8694	0.8952
VDSR★	1	0.22	11.5	0.6683	0.6640	0.6390	0.8546	0.7065	0.9321	0.9088	0.8792	0.9064
VESPCN	3	0.26	21.3	0.6685	0.6785	0.6435	0.8523	0.7157	0.9344	0.9078	0.8787	0.9060
VESPCN★	3	0.26	21.3	0.7057	0.6847	0.6577	0.8699	0.7293	0.8409	0.9121	0.8805	0.9101
VSRNet	7	0.23	11.3	0.6200	0.6425	0.6295	0.8354	0.6819	0.8979	0.9006	0.8685	0.8979
VSRNet★	7	0.23	11.3	0.6479	0.6553	0.6394	0.8468	0.6973	0.9281	0.9048	0.8734	0.9020
FastDVDnet	7	0.06	17.5	0.7534	0.7811	0.7234	0.8738	0.7829	0.9364	0.9177	0.8890	0.9148
FastDVDnet★	7	0.06	17.5	0.7731	0.8105	0.7401	0.8879	0.8029	0.9545	0.9375	0.9108	0.9348



(a) GT (b) Bicubic (c) VSRNet (d) VSRNet★



(e) VESPCN (f) VESPCN★ (g) FastDVDnet (h) FastDVDnet★

Figure 2: Visual comparisons of different methods on $4\times$ upscaling. ★ means student networks trained with our STD scheme. The frame is from Vid4-Foliage. Please zoom in for better visualization.

(a) GT (b) Bicubic (c) VSRNet (d) VSRNet★

(e) VESPCN (f) VESPCN★ (g) FastDVDnet (h) FastDVDnet★

Figure 3: Visual comparisons of different methods on $4\times$ upscaling. ★ means student networks trained with our STD scheme. The frame is from Vid4-Walk. Please zoom in for better visualization.

References

[1] Jose Caballero, Christian Ledig, Andrew Aitken, Alejandro Acosta, Johannes Totz, Zehan Wang, and Wenzhe Shi. Real-time video super-resolution with spatio-temporal networks and motion compensation. In *CVPR*, 2017. 2

[2] Armin Kappeler, Seunghwan Yoo, Qiqin Dai, and Aggelos K Katsaggelos. Video super-resolution with convolutional neural networks. *IEEE Transactions on Computational Imaging*, 2(2):109–122, 2016. 2

[3] Wonkyung Lee, Junghyup Lee, Dohyung Kim, and Bumsub Ham. Learning with privileged information for efficient image super-resolution. *ECCV*, 2020. 2

[4] Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *CVPR*, 2016. 2

[5] Matias Tassano, Julie Delon, and Thomas Veit. Fastdvdnet: Towards real-time deep video denoising without flow estimation. In *CVPR*, 2020. 1