XVFI: eXtreme Video Frame Interpolation – Supplementary Material –

Hyeonjun Sim* Jihyong Oh* Munchurl Kim†

Korea Advanced Institute of Science and Technology

{flhy5836, jhoh94, mkimee}@kaist.ac.kr

(a) 9.9 (b) 140.6 (c) 122.6 (e) 2.1

(f) 12.3 (g) 33.7 (h) 29.7 (i) 66.3

(j) 166.8 (k) 30.7 (l) 47.3 (m) 7.7

Figure 1. More examples of our X4K1000FPS dataset, which contain diverse motions in 4K-resolution of 1000 fps. The numbers below the examples are the magnitude means of optical flows between two input frames in 30 fps. Please refer to the arXiv version to watch this figure as a video clip.

1. Details of Proposed X4K1000FPS Dataset

1.1. Photographing Videos

In order to provide a wide range of object motions and various camera motion types at different speeds in diverse locations, the shooting rules were guided as follows: (i) shooting various objects independently moving while the camera is stationary, (ii) shooting videos from a moving car (fast translated videos), (iii) shooting while walking (moving at normal speed), (iv) shooting with the camera in irregular motion trajectories at non-uniform speeds, (v) shooting with zooming out or in and panning at the same time. Besides, the contents of the videos also include various objects (crowds, cars, trains, plants, animals, boats, traffic signs, signboards, waterfalls, buildings, etc.) in various types of places such as stadiums, stations, beaches, markets, parks, rivers, playgrounds, etc. Fig. 1 shows some representative thumbnails of spatiotemporally down-sampled 4K video at 15 fps for a visualization purpose. As shown in Fig. 1 in the main paper and Fig. 1 in this Supplementary Material, very extreme scenes including various camera motions, zooming, translations, speeds, occlusions and objects are contained in the X4K1000FPS dataset.

1.2. Test Dataset: X-TEST

We manually select the consecutive 32 frames for each test scene by considering the degrees of occlusion, optical flow magnitudes and diversity among 5,000 frames. We

*Both authors contributed equally to this work.

†Corresponding author.

compose nonuplets by sampling every 4 frames from 32 frames of each test scene. Since the frame rates of videos are often given in multiples of 30 in VFI benchmark datasets [8, 9] or real-world industries, we approximate that our test videos are set to 960 fps (= 32×30 fps) instead of 1,000 fps. Therefore, two input frames that are 32 frames apart are regarded as part of a 30 fps (= 960 / 32) video and converted to 240 fps by $\times 8$ multi-frame interpolation in the evaluation phase.

1.3. Train Dataset: X-TRAIN

To compose a valuable training dataset from our enormous videos for video frame interpolation, we select training samples based on the value of the occlusion map estimated by IRR-PWC [3]. The occlusion maps are approximated on the spatially down-sampled ($\times 1/4$) and temporally sub-sampled ($\times 1/32$) frames of the original 4K-resolution 1000 fps videos. Each target frame is fed into IRR-PWC with the previous and the next frames of the target frame, respectively. The resulting two occlusion maps are averaged to get the bidirectional occlusion map.

Then, we divide the 4K-resolution frame into overlapping patches of 768×768, which forms an 81×31 grid, except for the boundary of the 4K-resolution frame. This is because the boundary patches have undesirably large occlusion values when there are translation motions. Similarly, about 5,000 frames are divided into overlapping 154 clips of 65 consecutive frames except for the boundary period in the temporal dimension of the scenes. Thus, about 386K (= $81 \times 31 \times 154$) candidate training samples per scene of the patch size of 768×768 and the lengths of 65 frames are extracted from a 4K-resolution 1000 fps video. After that, the candidate training samples whose bidirectional occlusion value is the top 10% of those of all candidate training samples remained, and the others are discarded. Finally, total 4,408 training samples are sparsely selected as training data to prevent similar samples from being selected to maintain the diversity of the training samples.

2. Details of Architecture of XVFI-Net

In addition to Fig. 3 and 4 in the main paper, we present the detailed architectures of sub-networks of XVFI-Net in the case of the module scale factor $M = 4$ in Table 1 to Table 5. The series of rows represents the consecutive operations. The first column represents each layer's operation, and H,W and C indicate the spatial ratio with respect to bicubicly downsampled ($\times 1/2^s$) input frames I_i^s for each scale s and the number of channels of the output tensors, respectively. The last column denotes the names of some output tensors, which are worth mentioning. We omit the names of the output tensors if they are just intermediate tensors in the sub-networks. When the multiple tensors are input to each layer, they are concatenated channel-wise. 'res-block' represents a residual block which consists of *conv2d - relu - conv2d - identity addition*. The stride of the convolutional layer is set to 1, if not mentioned. The convolution filter sizes are 3×3 and 4×4 for the strides of 1 and 2, respectively.

As shown in Table 1, the Feature Extraction Block is a simple residual block-based sub-network. On the other hand, the flow estimation sub-networks, the BiFlownet and TFlownet, have a simple auto-encoder architecture to enlarge the receptive field as shown in Table 2, 3 and 4. The Refinement Block has a U-Net [6]-based architecture as in Table 5. The parameters of each sub-network are shared for all scale levels except for the BiFlownet at the lowest scale depth S, which is isolated in Table 2. The bidirectional flows are estimated directly from two input features C_0^S, C_1^S at the lowest scale level S, because there does not exist any provided initial flow.

Efficiency of XVFI-Net During Inference. The BiOF-T module can start from any down-scaled level, while the BiOF-T module can be skipped in the down-scaled levels ($s = 1, \ldots, S_{tst}$) as described in Fig. 3 in the main paper. By doing so, our XVFI-Net framework can accelerate run time about 22% faster compared to the full-recursion framework where both BiOF-I and -T modules are processed together in all scale levels for 4K video, when $S_{tst} = 5$. Besides, the additional runtimes induced by the smaller down-scaled levels ($s > 0$) are negligible since the runtimes of the BiOF-I module at down-scaled levels are much smaller than those of BiOF-I and BiOF-T modules at the original scale ($s = 0$).

As shown in the first scene with even extreme back and forth motions of a propeller with zoom-out, our XVFI-Net can surprisingly capture such a complex motion for VFI, but the SOTAs fail to interpolate the sophisticated tips of the propeller pointed by the yellow arrows. For the second scene, even while riding a fast-moving car, XVFI-Net better captures far tiny structures such as electric wires seen at the left part and a closer pole with a large pixel displacement pointed by the yellow arrows. For the third scene, the rightmost front car moves very fast, so all the previous methods fail to capture it, denoted by the yellow arrow, yielding severe artifacts (structural distortions). On the other hand, XVFI-Net precisely captures the especially right edges of the rightmost car. Finally, in the last scene even with the *extremely* hand-shaken frames, the XVFI-Net can also synthesize repeating similar stairs but all SOTAs tend to generate baggy artifacts. As a result, XVFI-Net significantly better handles large pixel displacements due to extreme motion and huge spatial resolutions.

3. Additional Qualitative Results

Visual Comparisons for VFI methods. We provide additional qualitative results on X-TEST (4K) in Fig. 2,

Operation	H,W	C	Remarks
input I_i^0	1	3	I_i^0
conv2d - relu	1	64	-
conv2d - relu	1/2	64	-
conv2d	1/4	64	Feat$_i$
resblock ($\times 2$) - add to Feat$_i$	1/4	64	C_i^0
conv2d ($\times s$)	1/4 $\times$ $1/2^s$	64	C_i^s

Table 1. The detailed architecture of the Feature Extraction Block of XVFI-Net. C_i^s is obtained by applying the last convolutional layer to C_i^0 s times recurrently. The parameters are temporally shared for the two input frames ($i = 0, 1$).

Operation	H,W ($\times$ $1/2^S$)	C	Remarks
input [C_0^S, C_1^S]	1/4	64$\times$2	-
conv2d (stride 2) - relu	1/8	128	-
conv2d (stride 2) - relu	1/16	256	-
NN upscale - conv2d - relu	1/8	128	-
NN upscale - conv2d - relu	1/4	64	-
conv2d	1/4	2+2+1+1	F_{01}^S, F_{10}^S, z_{01}^S, z_{10}^S

Table 2. The detailed architecture of the auto-encoder-based BiFlownet of XVFI-Net at the lowest scale depth.

Adobe240fps [8] (HD) in Fig. 3 and Vimeo90K [9] in Fig. 4 by each setting described in the main paper.

Visualization of Components of XVFI-Net. Fig. 5 shows the visualization of optical flows and occlusion masks of XVFI-Net$_v$. As expected, estimated flows at the upper level seem finer than those at the lower level (F_{t0}^1) as shown in Fig. 5. The coarse-to-fine structure gradually helps the whole XVFI framework boost the final VFI performance at original scale $s = 0$ based on the occlusion masks and the iteratively updated flows that are all *learned from scratch*.

4. Failure Cases

Since we delicately select the scenes to compose extremely challenging X-TEST, there exist inevitably failure patches *within the same 4K frame result*, where all compared methods including XVFI-Net (S$_{tst} = 5$), fail to accurately interpolate the intermediate frames. Fig. 6 shows the 4K failure results ($t = 0.5$) including several failure patches of ours because the input videos have very large magnitude means of optical flows (196.5) attributed to large camera shaking with the fast moving cars. Fig. 7 shows the failure cropped patches. First, the tiny electric line, which is hard to be distinguishable from static background, is failed to be accurately interpolated by all methods including ours, as indicated by a red arrow. Second, rotations of fast moving car's wheels are also challenging to be delicately synthesized while considering the degree of rotations, as pointed by two green arrows. On top of these, blurriness and abrupt brightness or color change in the input frames would also make all VFI methods challenging.

Please note that we have also provided all interpolated results for all compared methods of both original and retrained versions on X-TEST to be publicly available at https://github.com/JihyongOh/XVFI for easier comparison.

Acknowledgement. We specially thank Sung-Jun Yoon and Hyun-Ho Kim for photographing 4K videos on the spot.

References

[1] Wenbo Bao, Wei-Sheng Lai, Chao Ma, Xiaoyun Zhang, Zhiyong Gao, and Ming-Hsuan Yang. Depth-aware video frame interpolation. In *CVPR*, pages 3703–3712, 2019. 7

[2] Shurui Gui, Chaoyue Wang, Qihua Chen, and Dacheng Tao. Featureflow: Robust video interpolation via structure-to-texture generation. In *CVPR*, pages 14004–14013, 2020. 7

[3] Junhwa Hur and Stefan Roth. Iterative residual refinement for joint optical flow and occlusion estimation. In *CVPR*, pages 5754–5763, 2019. 2, 5

[4] Hyeongmin Lee, Taeoh Kim, Tae-young Chung, Daehyun Pak, Yuseok Ban, and Sangyoun Lee. Adacof: Adaptive collaboration of flows for video frame interpolation. In *CVPR*, pages 5316–5325, 2020. 7

[5] Junheum Park, Keunsoo Ko, Chul Lee, and Chang-Su Kim. Bmbc: Bilateral motion estimation with bilateral cost volume for video interpolation. In *ECCV*, 2020. 7

[6] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, pages 234–241. Springer, 2015. 2, 5

Operation	H,W ($\times\ 1/2^s$)	C	Remarks
input [C_0^s, $\tilde{C}_{01}^s$]	1/4	64×2	-
conv2d	1/4	64	$\hat{C}_{01}^s$
input [C_1^s, $\tilde{C}_{10}^s$]	1/4	64×2	-
conv2d	1/4	64	$\hat{C}_{10}^s$
input [$\hat{C}_{01}^s$, $\hat{C}_{10}^s$,$\tilde{F}_{01}^s$, $\tilde{F}_{10}^s$]	1/4	64×2+2×2	-
conv2d (stride 2) - relu	1/8	128	-
conv2d (stride 2) - relu	1/16	256	-
NN upscale - conv2d - relu	1/8	128	-
NN upscale - conv2d - relu	1/4	64	-
conv2d - add to [$\tilde{F}_{01}^s$, $\tilde{F}_{10}^s$]	1/4	2+2+1+1	F_{01}^s, F_{10}^s, z_{01}^s, z_{10}^s

Table 3. The detailed architecture of the auto-encoder-based BiFlownet of XVFI-Net except for the lowest scale depth.

Operation	H,W ($\times\ 1/2^s$)	C	Remarks
input [C_0^s, C_1^s, $\tilde{C}_{t0}^s$, $\tilde{C}_{t1}^s$,$\tilde{F}_{t0}^s$, $\tilde{F}_{t1}^s$]	1/4	64×4+2×2	-
conv2d (filter 1×1) - relu	1/4	64	-
conv2d (stride 2) - relu	1/8	128	-
conv2d (stride 2) - relu	1/16	256	-
NN upscale - conv2d - relu	1/8	128	-
NN upscale - conv2d - relu	1/4	64	-
conv2d - add to [$\tilde{F}_{t0}^s$, $\tilde{F}_{t1}^s$]	1/4	2+2+1	F_{t0}^s, F_{t1}^s, m^s

Table 4. The detailed architecture of the auto-encoder-based TFlownet of XVFI-Net.

[7] 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*, pages 1874–1883, 2016. 5

[8] Shuochen Su, Mauricio Delbracio, Jue Wang, Guillermo Sapiro, Wolfgang Heidrich, and Oliver Wang. Deep video deblurring for hand-held cameras. In *CVPR*, pages 1279–1288, 2017. 2, 3

[9] Tianfan Xue, Baian Chen, Jiajun Wu, Donglai Wei, and William T Freeman. Video enhancement with task-oriented flow. *International Journal of Computer Vision*, 127(8):1106–1125, 2019. 2, 3, 7

Operation	H,W ($\times 1/2^s$)	C	Remarks
input ([C_0^s, C_1^s, $\tilde{C}_{t0}^s$, $\tilde{C}_{t1}^s$])	1/4	256	-
pixel-shuffle [7] (↑4)	1	16	PS
input [PS, $F_{t0}^s \uparrow_2$, $F_{t1}^s \uparrow_2$, I_0^s, I_1^s, $\tilde{I}_{t0}^s$, $\tilde{I}_{t1}^s$]	1	16+2×2+3×4	-
conv2d (stride 2) - relu	1/2	64	enc_1
conv2d (stride 2) - relu	1/4	128	enc_2
conv2d (stride 2) - relu	1/8	256	-
conv2d - relu	1/8	256	-
NN upscale - concat to enc_2	1/4	384	-
conv2d - relu	1/4	128	-
NN upscale - concat to enc_1	1/2	192	-
conv2d - relu	1/2	64	-
NN upscale - conv2d	1	1+3	m^s, $\tilde{I}_r^s$

Table 5. The detailed architecture of the U-Net [6]-based Refinement Block of XVFI-Net.

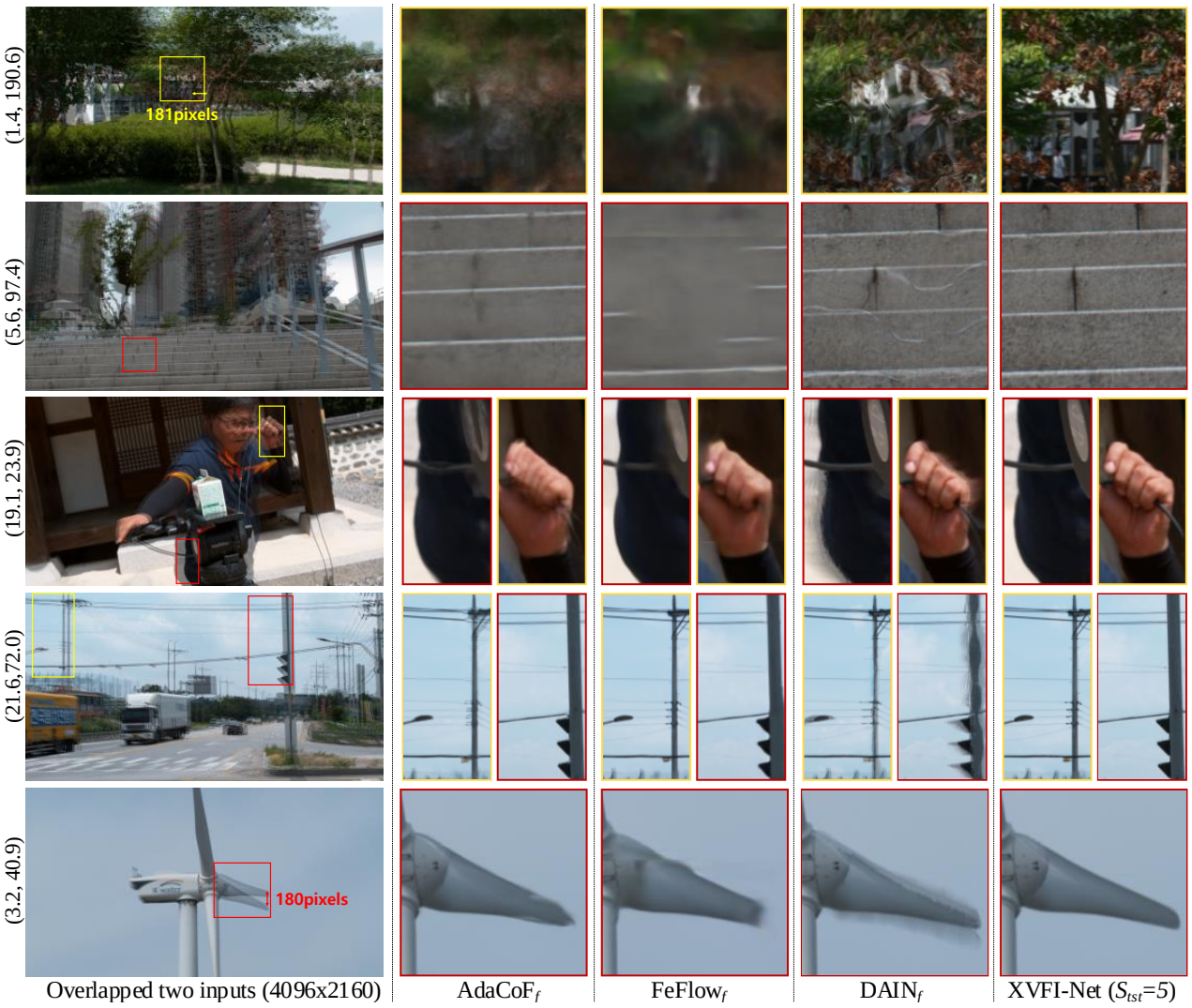

Figure 2. Visual comparisons for VFI results ($t = 0.5$) on X-TEST for our and *retrained* SOTA methods with X-TRAIN. (*,*): occlusions and optical flow magnitudes between the two input frames measured by IRR-PWC [3], respectively. *Best viewed in zoom.*

Figure 3. Visual comparisons for VFI results ($t = 0.5$) on Adobe240fps for our and *retrained* SOTA methods with X-TRAIN. *Best viewed in zoom.*

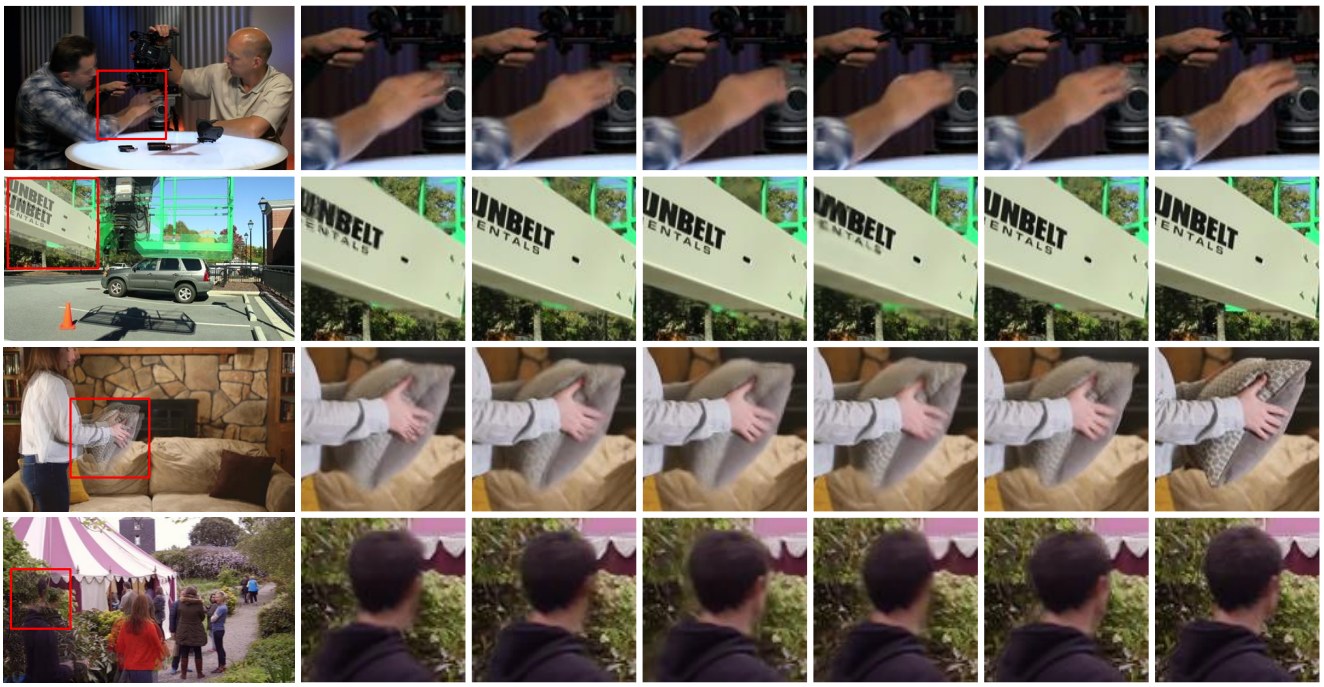

Figure 4. Visual comparisons of AdaCoF [4], DAIN [1], BMBC [5], FeFlow [2], our XVFI-Net$_v$ and the corresponding ground truth on the testset of Vimeo90K [9] triplet. *Best viewed in zoom.*

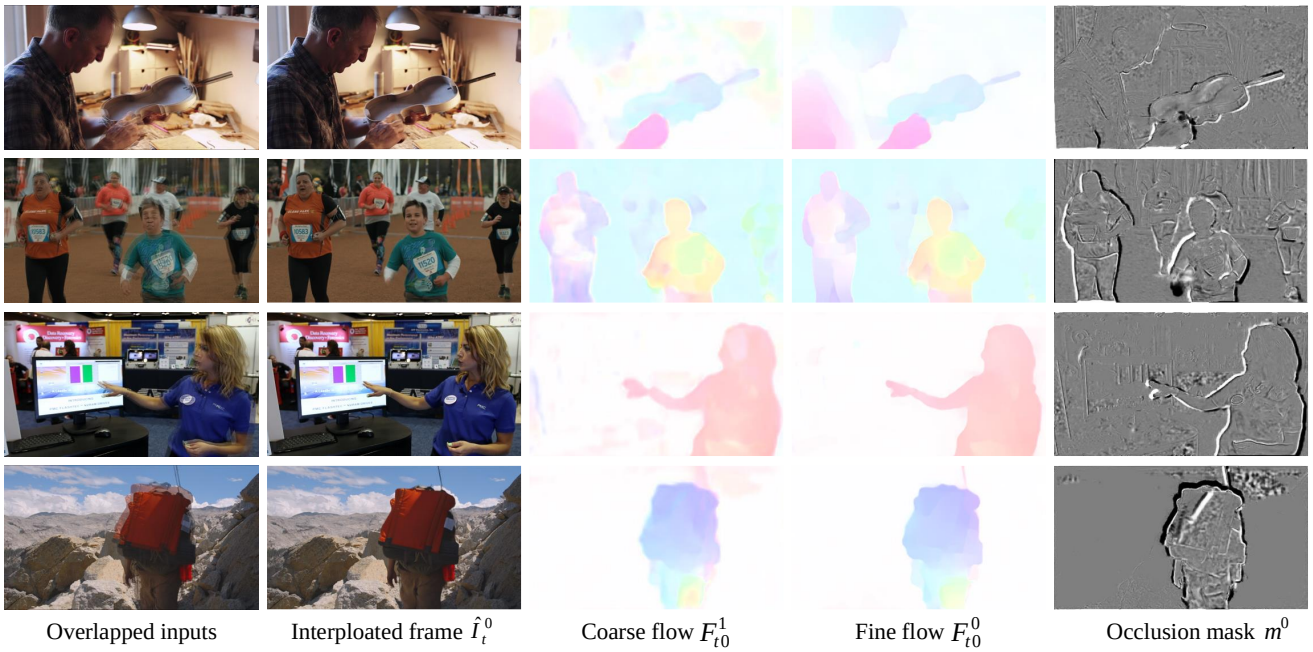

Figure 5. Visualization of optical flows and occlusion masks of XVFI-Net$_v$. The coarse and fine flows are extracted at scale 1 and 0, respectively.

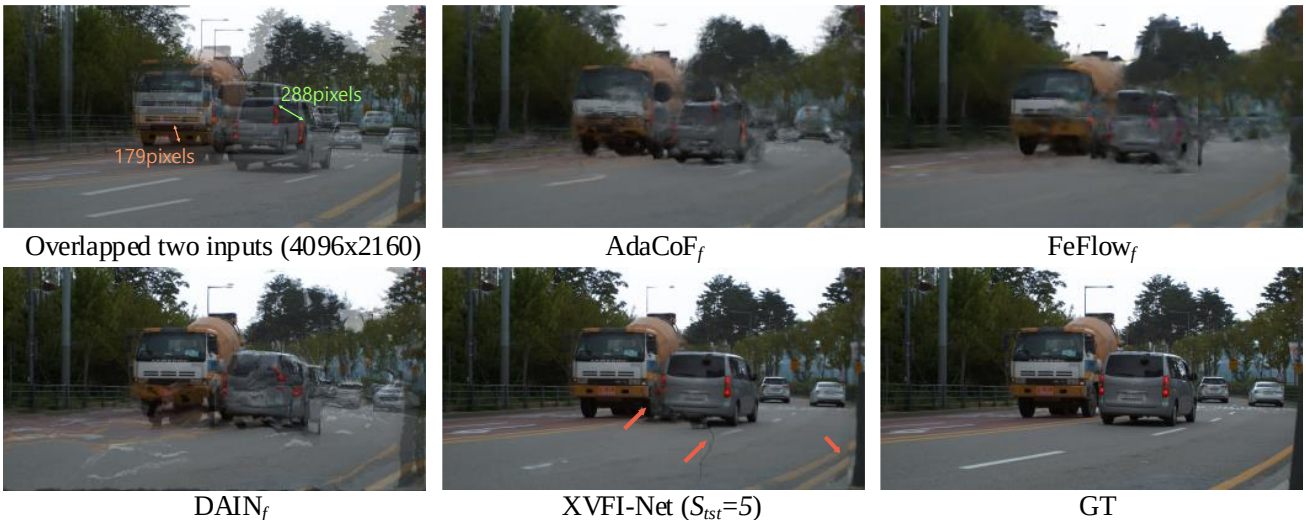

Figure 6. Failure cases of 4K result ($t = 0.5$) on X-TEST for our and *retrained* SOTA methods with X-TRAIN, including the corresponding ground truth. *Best viewed in zoom.*

Figure 7. Failure cases of cropped results ($t = 0.5$) on X-TEST for our and *retrained* SOTA methods with X-TRAIN, including the corresponding ground truth. *Best viewed in zoom.*