

# BF-STVSR: B-Splines and Fourier—Best Friends for High Fidelity Spatial-Temporal Video Super-Resolution

## Supplementary Material

### 1. Definition of B-spline function

The B-spline basis function  $\beta^n(x)$  is defined as the  $n$ -fold convolution of  $\beta^0(x)$  with itself, where  $n$  represents the polynomial degree. The function  $\beta^0(x)$  equals 1 when  $|x| < 0.5$  and 0 otherwise. As  $n$  increases, the support expands:  $\beta^1(x)$  spans  $[-1, 1]$ ,  $\beta^2(x)$  spans  $[-1.5, 1.5]$ , and  $\beta^3(x)$  spans  $[-2, 2]$ . In this study, we adopt the third-order B-spline ( $n = 3$ ) for B-spline Mapper. The definitions of  $\beta^3(x)$  and its derivative  $\frac{\partial}{\partial x}\beta^3(x)$  are provided in Eq (1) and Eq (2), respectively.

$$\beta^3(x) = \begin{cases} \frac{1}{6}(2+x)^3 & \text{if } -2 < x \leq -1; \\ \frac{1}{6}(4-6x^2-3x^3) & \text{if } -1 < x \leq 0; \\ \frac{1}{6}(4-6x^2+3x^3) & \text{if } 0 < x \leq 1; \\ \frac{1}{6}(2-x)^3 & \text{if } 1 < x \leq 2; \\ 0 & \text{otherwise.} \end{cases} \quad (1)$$

$$\frac{\partial}{\partial x}\beta^3(x) = \begin{cases} \frac{1}{2}(2+x)^2 & \text{if } -2 < x \leq -1; \\ -2x-1.5x^2 & \text{if } -1 < x \leq 0; \\ -2x+1.5x^2 & \text{if } 0 < x \leq 1; \\ -\frac{1}{2}(2-x)^2 & \text{if } 1 < x \leq 2; \\ 0 & \text{otherwise.} \end{cases} \quad (2)$$

### 2. Detailed Architectures of Mappers

We present the detailed network architecture of B-spline Mapper and Fourier Mapper in Fig 1. Both mappers take encoder features as input and capture temporal/spatial information using three-layer SIRENs [4]. In B-spline Mapper, a linear layer is used for each coefficient estimator, predicting the coefficients, knots, and dilations for B-spline representations. In Fourier Mapper, linear and convolutional layers follow the SIREN layers to estimate the amplitudes and frequencies of the Fourier coefficients. Finally, the B-spline or Fourier representations are passed through a linear layer to produce motion vectors or spatial features.

### 3. Different Position Encodings

We further compare our model with other position encoding techniques in Table 1. Specifically, we compare our model with Fourier Encoding (FE) [3], Thin-Plate Spline (TPS) [7], which is proved effective in image animation, and Gaussian function ( $\simeq \infty$ -order B-spline). For FE, we concatenate encoded coordinates with INR input features.

For fair comparison, we follow the training scheme of MoTIF [1], which includes the optical flow supervision from the RAFT [5]. For Gaussian, similar to ours, we learn scaling, mean, and variance factor. Ours model achieves the best performance, with other basis functions outperformed FE that solely relies on coordinates rather than leveraging priors of basis functions to focus on informative video details.

Table 1. Performance comparison of different position encodings on GoPro and Adobe240 datasets. Results are evaluated using PSNR (dB) and SSIM metrics. All frames are interpolated by a factor of  $\times 4$  in the spatial axis and  $\times 8$  in the temporal axis. FE denotes Fourier Encoding and TPS denotes Thin-Plate Spline. **Red** indicates the best performance.

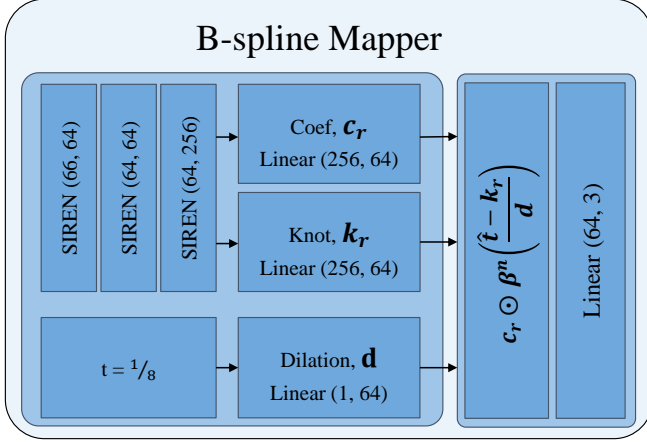
Method	Test Dataset			
	Adobe-Average	Adobe-Center	GoPro-Average	GoPro-Center
MoTIF (w FE)	30.02 / 0.8781	30.73 / 0.8855	30.08 / 0.8780	31.01 / 0.8877
MoTIF (w learnable FE)	30.05 / 0.8789	30.76 / 0.8862	30.11 / 0.8786	31.06 / 0.8884
MoTIF (w TPS)	30.02 / 0.8786	30.70 / 0.8856	30.10 / 0.8783	31.03 / 0.8880
BF-STVSR (w gaussian)	30.06 / 0.8787	30.71 / 0.8855	30.11 / 0.8777	31.01 / 0.8870
BF-STVSR + $\mathcal{L}_{RAFT}$ (Ours)	<b>30.14 / 0.8808</b>	<b>30.84 / 0.8877</b>	<b>30.20 / 0.8799</b>	<b>31.14 / 0.8893</b>
BF-STVSR (Ours)	30.12 / <b>0.8808</b>	30.83 / <b>0.8880</b>	<b>30.22 / 0.8802</b>	<b>31.17 / 0.8898</b>

### 4. Different Basis Function Configurations

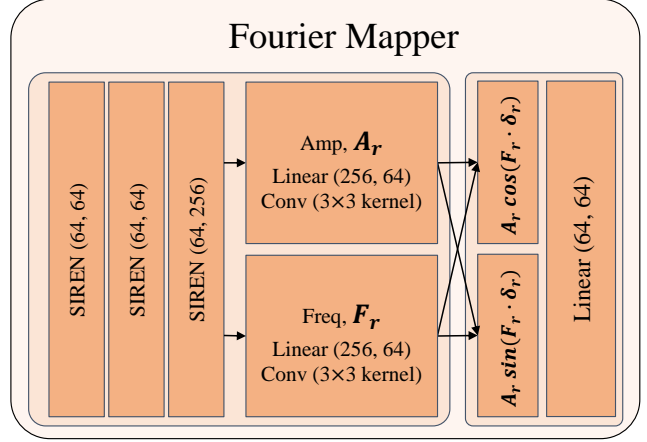
We conduct additional experiments by varying the basis functions used for both axes in Table 2. In all configurations, the basis function model the spatial and temporal axes, while other components (e.g., linear projection for intermediate motion vectors and reliability maps) remain the same. Here, we follow the training scheme of MoTIF [1], which incorporating the optical flow supervision in Eq. (5) of the main paper. Although the performance differences among these configurations are minimal, we adopt the configuration using B-spline for temporal representation and Fourier for spatial representation, as it achieves the highest performance.

Table 2. Performance comparison on the cross basis function configurations. Results are evaluated using PSNR (dB) and SSIM metrics. “Ours-cross” uses Fourier Mapper as temporal basis function and B-spline Mapper as spatial basis function. “Ours-FF” applies the Fourier Mapper for both spatial and temporal axes and “Ours-BB” employs the B-spline Mapper for both axes. **Red** indicates the best performance.

	GoPro-Center	GoPro-Average	Adobe-Center	Adobe-Average
Ours	<b>31.14 / 0.8893</b>	<b>30.20 / 0.8799</b>	<b>30.84 / 0.8877</b>	<b>30.14 / 0.8808</b>
Ours-cross	31.10 / 0.8892	30.17 / 0.8796	30.80 / 0.8876	30.10 / 0.8804
Ours-FF	31.12 / <b>0.8894</b>	30.18 / 0.8798	<b>30.84 / 0.8879</b>	30.13 / <b>0.8809</b>
Ours-BB	31.07 / 0.8885	30.16 / 0.8793	30.77 / 0.8865	30.09 / 0.8797



(a) Architecture of B-spline Mapper



(b) Architecture of Fourier Mapper

Figure 1. Detailed Architectures of B-spline Mapper and Fourier Mapper.

Table 3. Performance comparison on the C-STVSR baselines on Vimeo90K dataset. Results are evaluated using PSNR (dB) and SSIM metrics. All frames are interpolated by a factor of  $\times 4$  in the spatial axis and  $\times 6$  in the temporal axis. **Red** indicates the best performance.

Method	Test Dataset		
	Vimeo-Fast	Vimeo-Medium	Vimeo-Slow
VideoINR [2]	25.79 / 0.7980	28.37 / 0.8553	29.25 / 0.8727
MoTIF [1]	26.03 / 0.7909	28.68 / 0.8574	29.47 / 0.8739
BF-STVSR + $\mathcal{L}_{RAFT}$ (Ours)	<b>26.53 / 0.8079</b>	28.76 / <b>0.8594</b>	29.53 / 0.8748
BF-STVSR (Ours)	26.52 / 0.8054	<b>28.77 / 0.8590</b>	<b>29.55 / 0.8750</b>

## 5. Additional Results on Vimeo90K

We further evaluate C-STVSR baselines, including VideoINR [2], MoTIF [1], BF-STVSR +  $\mathcal{L}_{RAFT}$  and BF-STVSR, all trained on Adobe240 dataset, using the Vimeo90K dataset. Vimeo90K, a widely used benchmark for video enhancement, comprises 64,612 clips, each containing seven frames. Following [6], we categorize the test-set into three motion groups: Fast, Medium, and Slow. The evaluation results, presented in Table 3, show that our model achieves higher PSNR across all motion categories. Notably, it outperforms VideoINR by approximately 0.7 dB and MoTIF by 0.5 dB on the Vimeo-Fast subset, highlighting its superior ability to effectively handle motion dynamics, even in challenging scenarios.

Additionally, we train the C-STVSR models (VideoINR, MoTIF, and BF-STVSR +  $\mathcal{L}_{RAFT}$ ) using the Vimeo90K trainset until 450K training iterations. Similar to the Adobe240 dataset, we select seven consecutive frames from each video clip, use the 1<sup>st</sup> and 7<sup>th</sup> frames as input reference frames, and randomly sample three intermediate frames as ground-truth. These models are then evaluated on GoPro and Adobe240 datasets, following the -Center

Table 4. Performance comparison on the C-STVSR baselines on GoPro and Adobe240 datasets. Baseline models are trained on Vimeo90K septuplet dataset. Results are evaluated using PSNR (dB) and SSIM metrics. All frames are interpolated by a factor of  $\times 4$  in the spatial axis and  $\times 8$  in the temporal axis. **Red** indicates the best performance.

Method	Test Dataset			
	Adobe-Average	Adobe-Center	GoPro-Average	GoPro-Center
VideoINR [2]	28.95 / 0.8527	29.58 / 0.8603	29.43 / 0.8657	30.20 / 0.8751
MoTIF [1]	28.81 / 0.8495	29.47 / 0.8580	29.44 / 0.8649	30.24 / 0.8742
BF-STVSR + $\mathcal{L}_{RAFT}$ (Ours)	<b>29.30 / 0.8611</b>	<b>29.88 / 0.8677</b>	<b>29.75 / 0.8709</b>	<b>30.53 / 0.8796</b>

and -Average evaluation protocols detailed in the main paper. The results, presented in Table 4, demonstrate that our model consistently achieves higher PSNR across all datasets, with improvements of approximately 0.4 dB over VideoINR and 0.3 dB over MoTIF.

These results suggest that our model is both robust and versatile. While trained on specific datasets, it shows promising generalization across diverse datasets, achieving competitive performance in reconstructing high-quality frames. In comparison, previous methods exhibit some challenges in adapting to unseen data, highlighting the potential advantages of our approach.

## 6. Additional Qualitative Results

We provide more qualitative comparison between BF-STVSR and prior methods (VideoINR [2], MoTIF [1]), all trained on Adobe240 dataset. Fig 2 and Fig 3 present temporal interpolation results (with the spatial scale fixed at 1) on the DAVIS dataset across all temporal coordinates, which were not shown in the main paper, at both the out-of-distribution scale ( $\times 6$ ) and in-distribution scale ( $\times 8$ ), respectively. Fig 4 shows the spatial-temporal interpolation results on GoPro dataset at in-distribution temporal

scale ( $\times 8$ ) and in-distribution spatial scale ( $\times 4$ ). Fig 5 shows the temporal interpolation results on the Vimeo-Medium testset at the out-of-distribution scale ( $\times 6$ ). Fig 6 and Fig 7 show spatial-temporal interpolation results on the Vimeo-Fast testset at the out-of-distribution temporal scale ( $\times 6$ ) and both out-of-distribution and in-distribution spatial scales ( $\times 2$ ,  $\times 4$ ), respectively.

## References

- [1] Yi-Hsin Chen, Si-Cun Chen, Yen-Yu Lin, and Wen-Hsiao Peng. Motif: Learning motion trajectories with local implicit neural functions for continuous space-time video super-resolution. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 23131–23141, 2023. 1, 2
- [2] Zeyuan Chen, Yinbo Chen, Jingwen Liu, Xingqian Xu, Vidit Goel, Zhangyang Wang, Humphrey Shi, and Xiaolong Wang. Videoinr: Learning video implicit neural representation for continuous space-time super-resolution. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022. 2
- [3] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *European Conference on Computer Vision*, pages 405–421, 2020. 1
- [4] Vincent Sitzmann, Julien N.P. Martel, Alexander W. Bergman, David B. Lindell, and Gordon Wetzstein. Implicit neural representations with periodic activation functions. In *NeurIPS*, 2020. 1
- [5] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*, pages 402–419. Springer, 2020. 1
- [6] Xiaoyu Xiang, Yapeng Tian, Yulun Zhang, Yun Fu, Jan P. Allebach, and Chenliang Xu. Zooming slow-mo: Fast and accurate one-stage space-time video super-resolution. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3370–3379, 2020. 2
- [7] Jian Zhao and Hui Zhang. Thin-plate spline motion model for image animation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3657–3666, 2022. 1



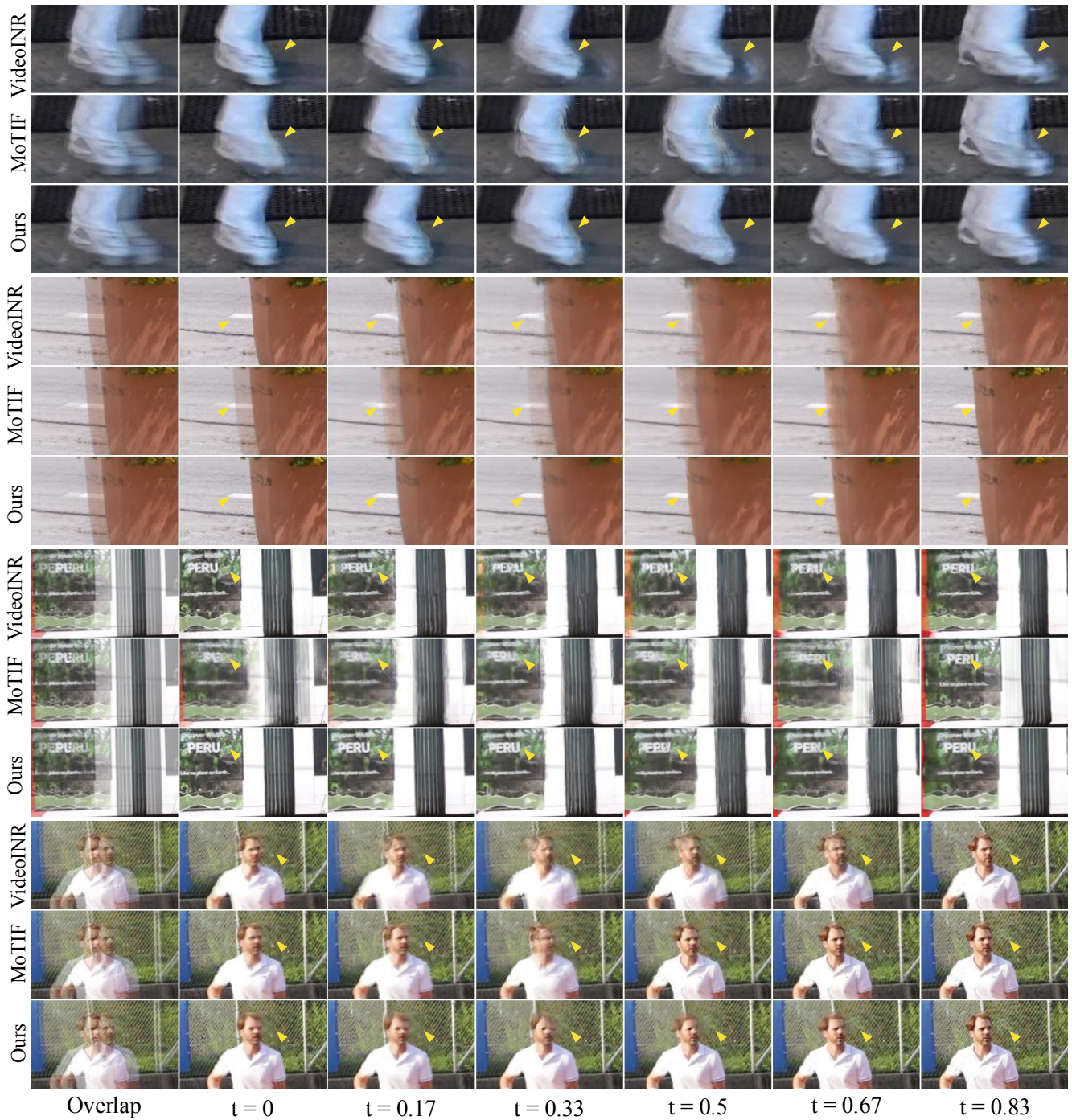


Figure 2. Qualitative comparison on arbitrary scale temporal interpolation on out-of-distribution ( $\times 6$ ) with all time coordinates. We use the DAVIS dataset for evaluation. “Overlap” refers to the averaged image of two input frames ( $t = 0, 1$ ), and the following images are interpolation results at  $t = (0, 1)$ .





Figure 3. Qualitative comparison on arbitrary scale temporal interpolation on in-distribution ( $\times 8$ ) with all time coordinates. We use the DAVIS dataset for evaluation. “Overlap” refers to the averaged image of two input frames ( $t = 0, 1$ ), and the following images are interpolation results at  $t = (0, 1)$ .





Figure 4. Qualitative comparison on arbitrary scale temporal interpolation on in-distribution ( $\times 8$ ) for temporal scale and in-distribution ( $\times 4$ ) for spatial scale with all time coordinates. We use the DAVIS dataset for evaluation. “Overlap” refers to the averaged image of two input frames ( $t = 0, 1$ ), and the following images are interpolation results at  $t = (0, 1)$ .



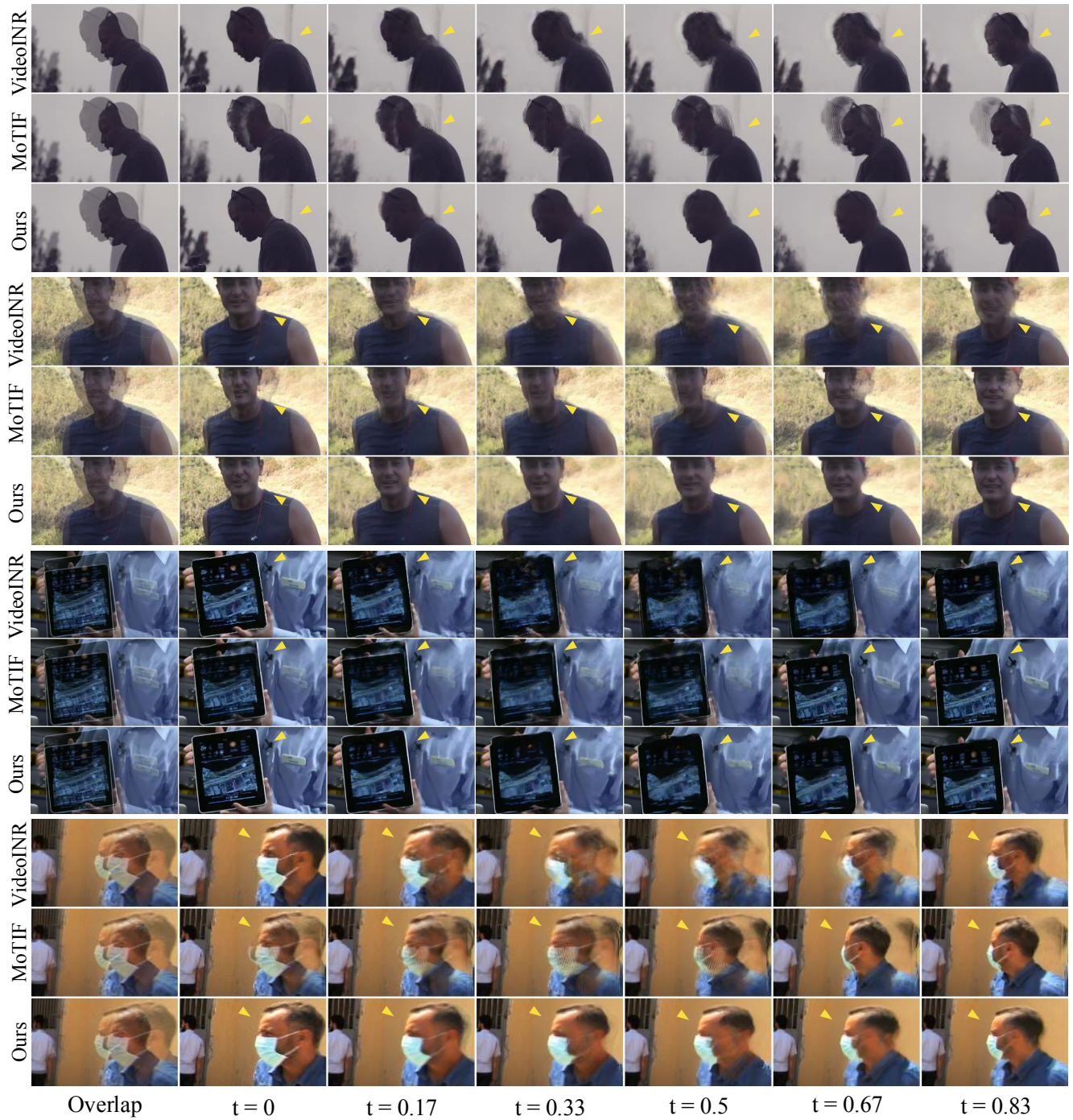


Figure 5. Qualitative comparison on arbitrary scale temporal interpolation on out-of-distribution ( $\times 6$ ) with all time coordinates. We use the Vimeo-medium testset for evaluation. “Overlap” refers to the averaged image of two input frames ( $t = 0, 1$ ), and the following images are interpolation results at  $t = (0, 1)$ .



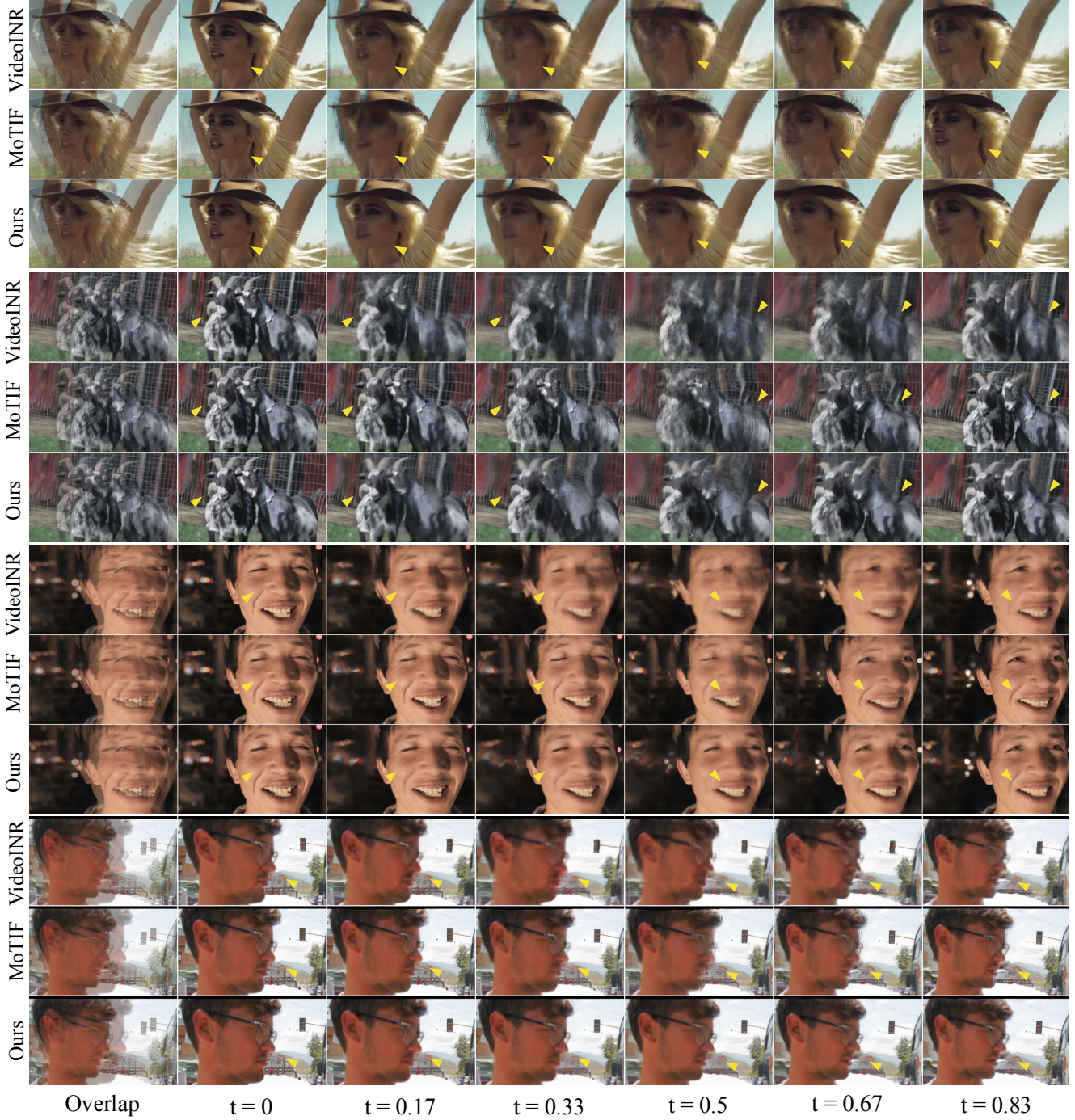


Figure 6. Qualitative comparison on arbitrary scale spatial-temporal interpolation on out-of-distribution ( $\times 6$ ) for temporal scale and out-of-distribution ( $\times 2$ ) for spatial scale with all time coordinates. We use the Vimeo-Fast testset for evaluation. “Overlap” refers to the averaged image of two input frames ( $t = 0, 1$ ), and the following images are interpolation results at  $t = (0, 1)$ .





Figure 7. Qualitative comparison on arbitrary scale spatial-temporal interpolation on out-of-distribution ( $\times 6$ ) for temporal scale and out-of-distribution ( $\times 4$ ) for spatial scale with all time coordinates. We use the Vimeo-Fast testset for evaluation. “Overlap” refers to the averaged image of two input frames ( $t = 0, 1$ ), and the following images are interpolation results at  $t = (0, 1)$ .