# Enhancing Video Super-Resolution via Implicit Resampling-based Alignment

Kai Xu Ziwei Yu Xin Wang Michael Bi Mi Angela Yao
kxu@comp.nus.edu.sg

## A. Methods

### A.1. Bilinear / Bicubic Interpolation

Bilinear / Bicubic Interpolation estimates the resampled value as a weighted sum of the 4 (bilinear) or 16 (bicubic) discrete neighbours around $(a, b)$, which we denote in short form as $\lfloor a, b \rceil$:

$$\mathbf{X}_r(a,b) = \sum_{(x,y)\in\lfloor a,b\rceil_{bi}} w_{xy} \cdot \mathbf{X}[x,y], \tag{1}$$

where $w_{xy}$ are the associated weighting coefficients based on either a linear (bilinear) or quadratic (bicubic) interpolation of $a$ and $b$ with respect to the neighbouring coordinates.

For bilinear interpolation:

$$w_{xy} = |a-x| * |b-y| \tag{2}$$

For bicubic interpolation, please refer to [4] for detailed definition.

### A.2. Network Structures

We employed a single-layer fully connected layer as the encoder for $F_q$, $F_k$, and $F_v$ in our experiments. These experiments were conducted on both CNN-based backbones, following the architecture of BasicVSR [3], and Transformer-based backbones, following the PSRT-recurrent model [8]. Alignment was applied for either first-order or second-order bidirectional propagation, with a default window size of 2x2 unless explicitly stated.

The detailed network structure for IA-CNN is provided in Fig. 1, while that for IA-RT is presented in Fig. 2.

For IA-CNN, we adapted first-order bidirectional propagation following BasicVSR [8]. The number of channels was set to 64, and each propagation branch comprised 30 residual blocks. The IA module had 64 channels, and the attention module for implicit alignment included 8 heads. We note that only 0.01M of the increase is due to alignment alignment, with the remaining 2.2M coming from the requirement for feature computation.

For IA-RT, two consecutive second-order bidirectional propagation blocks were employed, following the PSRT-recurrent model [8]. The number of channels for embedding was 120, and each propagation branch contained 18 Multi-Frame Self-Attention Blocks (MFSABs) [8] with shortcuts every 6 blocks. The IA module had 120 channels, and the attention module included 6 heads.

## B. Experimental Details

### B.1. Alignment Study

For the synthetic data, we partitioned the training videos from the clean data track of Sintel [1] into 20 training and 3 testing videos, and we present the results based on the testing set comprising *ambush5*, *market6*, and *mountain*$_1$.

All alignment methods evaluated share a consistent network structure and training parameters. We employed the Adam optimizer for training, running for 100,000 iterations at a learning rate of 2e-4, with a batch size of 8.

For alignment studies on the Sintel dataset, we utilized first-order backward propagation due to the availability of only backward optical flow ground truth. Each propagation branch contains 9 MFSABs with shortcuts every three blocks. The number of channels for embedding is set to 60. The total training iterations are 100,000, and the learning rate starts at 2e-4, with a cosine learning rate decay to 1e-7 towards the end of training. The batch size remains 8 throughout.

### B.2. Comparison with State-of-the-Art:

All experiments were conducted using bicubic 4X downsampling. The training dataset includes the REDS [7] and Vimeo-90K [10] datasets, while the testing dataset comprises REDS4 [7], Vid4 [6], and Vimeo-90K-T [10].

For IA-CNN on the REDS dataset, we trained for 300k iterations using 15 input frames, with a learning rate of 2e-4 and a cosine learning rate decay to 1e-7. The batch size was set to 8. Subsequently, we fine-tuned the model on the Vimeo-90K dataset for 300k iterations using the pre-trained weights from the REDS training.

For IA-RT on the REDS dataset, we trained for 300k

| Method | Param. (M) | FLOPs (T) | Runtime (ms) |
|---|---|---|---|
| DUF | 5.8 | 2.34 | - |
| RBPN | 12.2 | 8.51 | - |
| EDVR [9] | 20.6 | 2.95 | - |
| VSRT [2] | 32.6 | 1.60 | - |
| VRT [5] | 35.6 | 1.30 | - |
| PSRT-recurrent [8] | 13.4 | 1.50 | 2020† |
| IA-RT (ours) | 13.4 | 1.62 | 2105 |

Table 1. The comparison of the parameters, FLOPs and the runtime for VSR models. †The runtime is re-estimated on RTX-A5000 GPU for fair comparison.

iterations using 16 input frames, with a learning rate of 2e-4 and a cosine learning rate decay to 1e-7. The batch size remained 8.

When training IA-RT on the Vimeo-90K dataset, we first conducted 300k iterations with 14 input frames with flip sequence, we then train model on 7 input frames with flip sequence, using a learning rate of 2e-4 and a cosine learning rate decay to 1e-7. We initialized the weights using the model trained on the REDS dataset. The batch size remained 8.

Test results for the REDS model are reported on the REDS4 dataset, while test results for the Vimeo-90K model are reported on Vimeo-90K-T and Vid4.

**Evaluation metrics:** We calculate PSNR and SSIM on the RGB channel for REDS4 and Y channel for Vimeo-90K-T and Vid4.

**Ablation Studies:** For ablation studies on the REDS dataset, a model with a single second-order bi-directional propagation block is employed. Each propagation branch consists of 9 MFSABs with shortcuts every 3 blocks. The total training iterations are set to 200,000, with a learning rate initialized at 2e-4 and subjected to a cosine learning rate decay, reaching 1e-7 at the end of training. The batch size used for these experiments is 4.

### B.3. FLOPS and Runtime Comparison

We conducted an analysis of the FLOPs, as presented in Tab. 1. The overall complexity of a single IA module is 14.93 GFLOPs with an input low-resolution (LR) frame size of $180 \times 320$. In comparison with PSRT-recurrent, eight propagations with IA are performed per frame on average. Thus, the FLOPs for Implicit Alignment with Recurrent Transformers (IA-RT) is calculated as follows:

$$\text{FLOPs}_{\text{IA-RT}} = \text{FLOPs}_{\text{PSRT}} + 14.93\ \text{G} \times 8 = 1.62\ \text{T}$$

For a fair runtime comparison, we estimated the inference time for PSRT-recurrent and IA-RT on the same hardware, namely the RTX-A5000.

### B.4. Training Time

The training time on an RTX-A5000 for IA-RT is 6.7 ms/iter for 16 input training frames and 2.6 ms/iter for 6 input training frames. In comparison, the training time for PSRT-recurrent on RTX-A5000 is 6.4 ms/iter for 16 input training frames and 2.5 ms/iter for 6 input training frames.

### B.5. Different Down-sampling Degradation Comparison with PSRT

We add ablation studies on video super resolution for Blur Down-sampling Degradation **(BD)** and JPEG and MPEG Compression Down-sampling Degradation **(JPEG-D, MPEG-D)**, using data from the REDS official website and models from PSRT and IA-RT trained on REDS BI-degradation. Our results demonstrate that our method still outperforms the current state-of-the-art PSRT, although the performance gap narrows as optical flow accuracy decreases. This is attributed to our method's feature of sub-pixel information reconstruction, which benefits from precise optical flow.

| | BD | JPEG-D | MPEG-D |
|---|---|---|---|
| PSRT | 24.917 | 23.149 | 24.619 |
| IA-RT | 24.931 | 23.280 | 24.635 |

Table 2. Comparison with PSRT on different down-sample degradation

## C. More Visual Comparison

We offer additional visual comparisons in Fig. 3 and Fig. 4.

## References


[1] Daniel J Butler, Jonas Wulff, Garrett B Stanley, and Michael J Black. A naturalistic open source movie for optical flow evaluation. In *Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part VI 12*, pages 611–625. Springer, 2012. 1

[2] Jiezhang Cao, Yawei Li, Kai Zhang, and Luc Van Gool. Video super-resolution transformer. *arXiv preprint arXiv:2106.06847*, 2021. 2

[3] Kelvin CK Chan, Xintao Wang, Ke Yu, Chao Dong, and Chen Change Loy. Basicvsr: The search for essential components in video super-resolution and beyond. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4947–4956, 2021. 1

[4] Robert Keys. Cubic convolution interpolation for digital image processing. *IEEE transactions on acoustics, speech, and signal processing*, 29(6):1153–1160, 1981. 1

[5] Jingyun Liang, Jiezhang Cao, Yuchen Fan, Kai Zhang, Rakesh Ranjan, Yawei Li, Radu Timofte, and Luc Van Gool. Vrt: A video restoration transformer. *arXiv preprint arXiv:2201.12288*, 2022. 2

[6] Ce Liu and Deqing Sun. On bayesian adaptive video super resolution. *IEEE transactions on pattern analysis and machine intelligence*, 36(2):346–360, 2013. 1

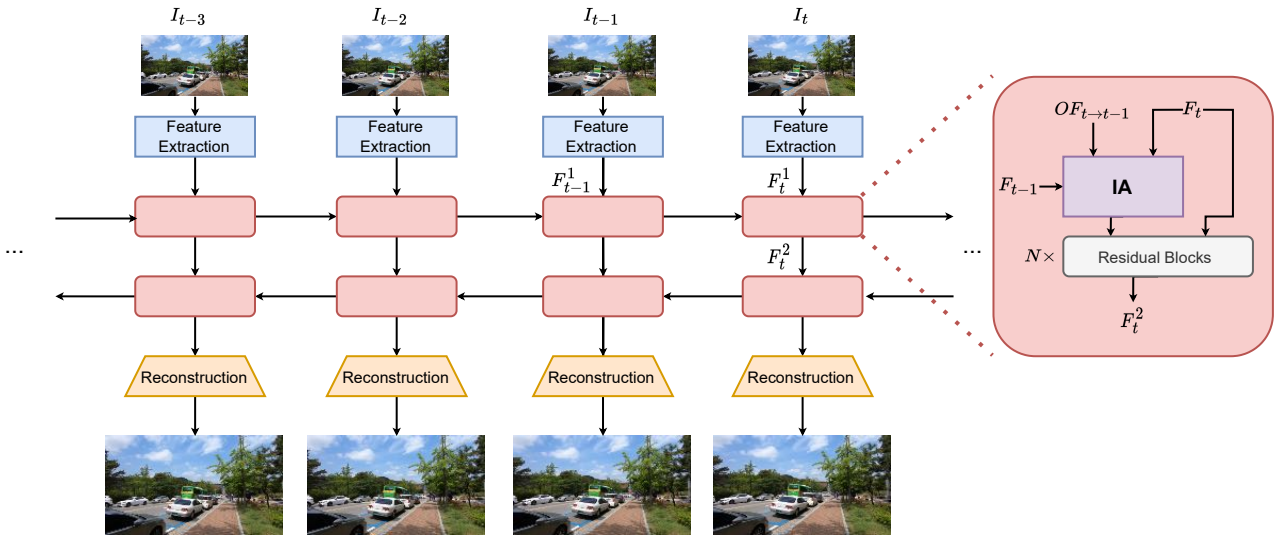


Figure 1. Network Structure for IA-CNN.

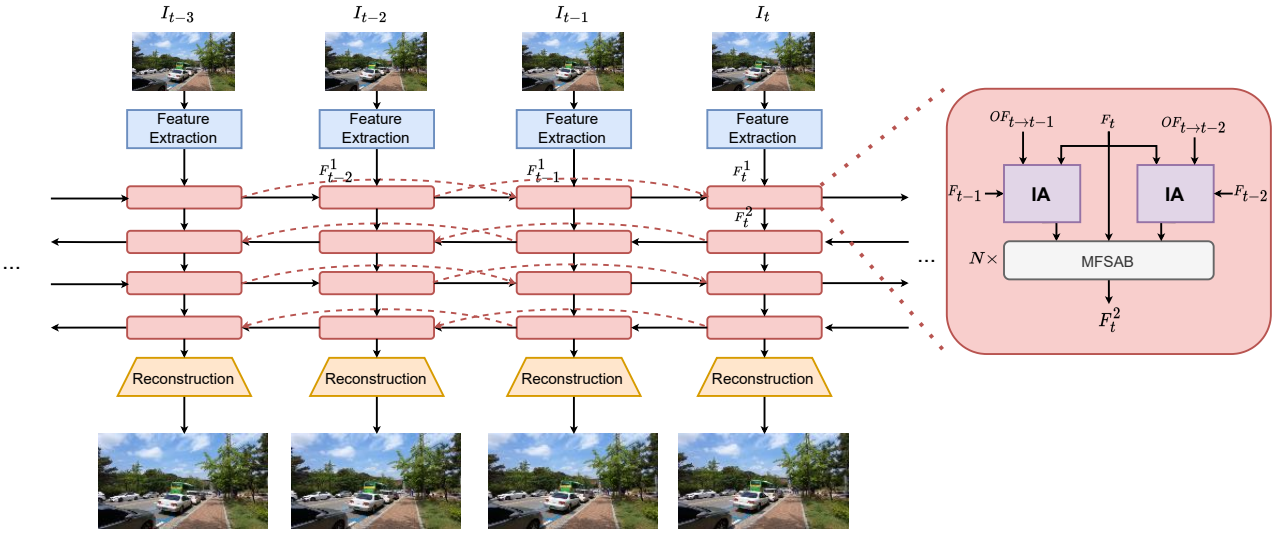


Figure 2. Network Structure for IA-RT.


[7] Seungjun Nah, Sungyong Baik, Seokil Hong, Gyeongsik Moon, Sanghyun Son, Radu Timofte, and Kyoung Mu Lee. Ntire 2019 challenge on video deblurring and super-resolution: Dataset and study. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019. 1

[8] Shuwei Shi, Jinjin Gu, Liangbin Xie, Xintao Wang, Yujiu Yang, and Chao Dong. Rethinking alignment in video super-resolution transformers. *arXiv preprint arXiv:2207.08494*, 2022. 1, 2

[9] Xintao Wang, Kelvin CK Chan, Ke Yu, Chao Dong, and Chen Change Loy. Edvr: Video restoration with enhanced deformable convolutional networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019. 2

[10] Tianfan Xue, Baian Chen, Jiajun Wu, Donglai Wei, and William T Freeman. Video enhancement with task-oriented flow. *International Journal of Computer Vision*, 127:1106–1125, 2019. 1

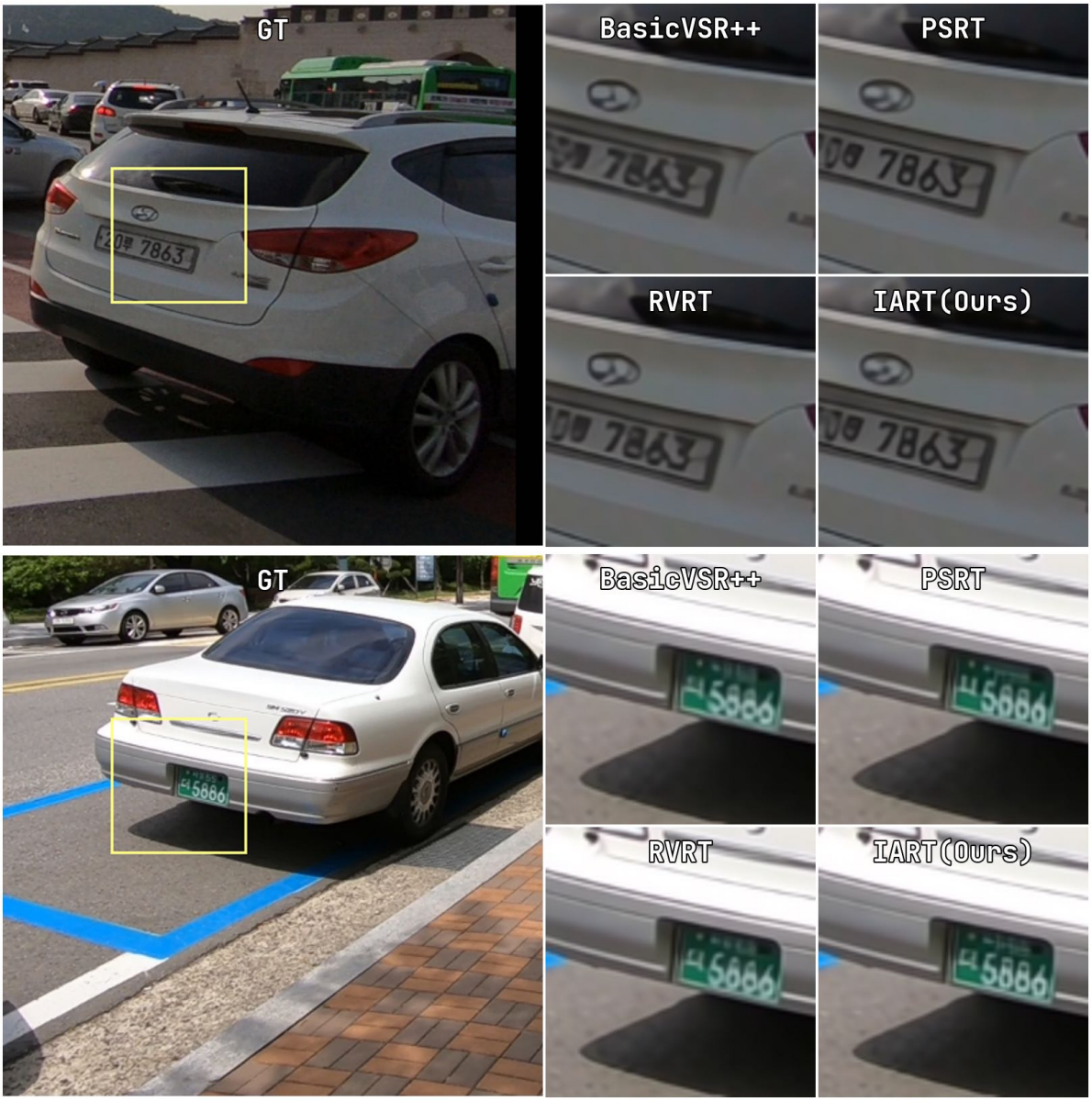


Figure 3. Visual Comparison on REDS4.


Figure 4. Visual Comparison on REDS4.