An Implicit Alignment for Video Super-Resolution

Kai Xu[1] Ziwei Yu[1] Xin Wang[2] Michael Bi Mi[2] Angela Yao[1]
[1]National University of Singapore, [2]Huawei International Pte Ltd, Singapore
{kxu,ayao}@comp.nus.edu.sg, yuziwei@u.nus.edu
wangxin237@huawei.com, michaelbimi@yahoo.com

Abstract

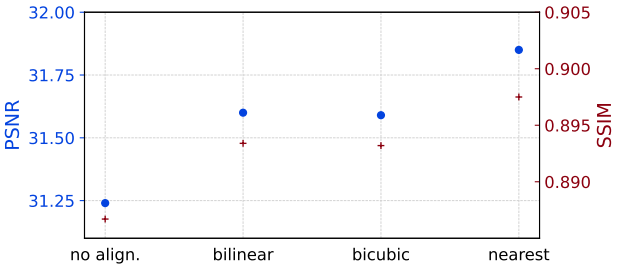
Video super-resolution commonly uses a frame-wise alignment to support the propagation of information over time. The role of alignment is well-studied for low-level enhancement in video, but existing works have overlooked one critical step – re-sampling. Most works, regardless of how they compensate for motion between frames, be it flow-based warping or deformable convolution/attention, use the default choice of bilinear interpolation for re-sampling. However, bilinear interpolation acts effectively as a low-pass filter and thus hinders the aim of recovering high-frequency content for super-resolution.

This paper studies the impact of re-sampling on alignment for video super-resolution. Extensive experiments reveal that for alignment to be effective, the re-sampling should preserve the original sharpness of the features and prevent distortions. From these observations, we propose an implicit alignment method that re-samples through a window-based cross-attention with sampling positions encoded by sinusoidal positional encoding. The re-sampling is implicitly computed by learned network weights. Experiments show that the proposed implicit alignment enhances the performance of state-of-the-art frameworks with minimal impact on both synthetic and real-world datasets. [1]

1. Introduction

Video super-resolution (VSR) recovers a spatially high-resolution sequence of frames from a low-resolution sequence. While image super-resolution can be applied naively to each frame individually, the temporal correlations across the frames give an extra source of information to improve the super-resolved output. As such, the main difference in designing architectures for video versus image super-resolution lies in the exploration of temporal dependencies. Previous works [25, 3, 28, 10] have shown that spatial alignment is an essential pre-processing step for effective information exchange across the frames. Given the frame-to-frame motions, either of the camera or the objects in the scene, alignment provides indications for extra sub-pixel information.

(a) Comparison with the original image and the optical flow warped image from the previous frame with bilinear interpolation and nearest interpolation. Bilinear interpolation smooths out the image pattern (red arrow) while nearest interpolation produces sharper results but introduces distortions (blue arrow).



(b) Video super-resolution results with different re-sampling methods on the Sintel dataset [1]. We perform first-order backward image alignments from $t+1$ to t using ground-truth optical flow.

Frame-wise alignment has two steps: motion estimation and motion compensation. The standard alignment [19, 3, 31] estimates motion via optical flow and compensates the motion with a backward warp. In the warping, the default option is to use bilinear interpolation for re-sampling for any non-integer flow values. A separate line of work [18, 4, 28] merges motion estimation and compensation into a single step, in the form of deformable convolutions and deformable attention. These works use multiple offsets, which can be viewed as a multi-reference ensemble (versus the single-reference setting of optical flow). Never-

[1] https://github.com/kai422/IART

theless, bilinear interpolation is also used when re-sampling the pixel/feature values for convolution or attention operations.

As other studies have shown, high-frequency components are crucial for the learning of super-resolution tasks [15]. Yet bilinear interpolation smooths out pixels or feature values, directly contradicting the aim of recovering high-frequency information for super-resolution. In fact, [25] recently observed that even a crude nearest-neighbour interpolation can perform better for patch alignment on a VSR transformer.

This paper takes a deep dive into re-sampling in alignment for video super-resolution. One key challenge to study re-sampling is the confounding effect from the estimated motions. The accuracy of motion estimation affects the reconstruction performance [3], and different re-sampling methods have different levels of robustness to this accuracy. Therefore, it is challenging to evaluate and compare various re-sampling methods under real-world optical flow conditions. To better understand the effect of re-sampling on video super-resolution, we design a series of experiments on a synthetic dataset with ground-truth optical flow. This allows us to isolate and remove the influence of motion estimation for alignment.

Fig. 1a shows the warping results on image alignment using bilinear and nearest interpolation, where bilinear interpolation preserves the original image structure but smooths out the pattern; nearest interpolation produces sharper results but introduces distortions. Fig. 1b plots video super-resolution PSNR and SSIM. On a first-order image alignment with ground-truth optical flow for synthetic dataset Sintel [1], the nearest interpolation performs better than bilinear interpolation and bicubic interpolation.

Based on these observations, we aim to design a balanced alignment. Ideally, we target interpolation that does not have any smoothness prior when reconstructing re-sampled pixels, which is the case for bi-linear interpolation. At the same time, we wish to avoid distortions by providing accurate sub-pixel information aggregation.

Inspired by recent image neural implicit representations [7, 30], we propose a new alignment module with an *implicit* re-sampling. The re-sampling is achieved through a local cross-attention module, applied to a feature window based on the optical flow offset. Rather than re-sampling on the target frame explicitly, *i.e*., solving for the sub-pixel feature value via interpolation, the values are aggregated via an affinity matrix. This matrix is computed by similarity comparison with feature encodings and positional encodings. The attention-based aggregation imposes no smoothness constraints on the re-sampling process. Furthermore, we encode the sub-pixel coordinate information from the flow estimate with a sinusoidal positional encoding. As the result, the implicit alignment largely surpasses optical flow with the nearest interpolation and exceeds the state-of-the-art alignment method for the VSR transformer model on both synthetic and real-world datasets.

The work closest related to ours in spirit is [25], where they point out that inaccurate optical flow and the bilinear re-sampling methods could corrupt the feature patterns and reconstruction results. As a result, they propose patch alignment to handle the inaccurate optical flow. Our work further investigates this effect by experimenting with re-sampling methods under different optical flow scenarios for both image and feature alignments. Moreover, our proposed implicit alignment is as robust as patch alignment when the offset estimation is inaccurate, since it refers to a window of pixel features. Our implicit alignment also provides more precise sub-pixel information when the offset estimation is accurate, as it embeds the exact offsets into the positional encoding. In contrast, patch alignment averages the offsets indiscriminately.

We summarize our contribution as follows:

- To better understand alignment for VSR, we design a series of experiments to disentangle the effects of motion estimation from re-sampling. Our findings reveal that an effective re-sampling method should impose no smoothness prior and introduce no distortions.
- We propose an implicit alignment method where estimated motion is encoded by a sinusoidal positional encoding and re-sampling is implicitly performed by window-based attention.
- The proposed implicit alignment outperforms current the state-of-the-art alignment methods on both a synthetic dataset and a real-world dataset.

2. Related Work

Video Super-Resolution: Video super-resolution aims to recover a spatially high-resolution sequence from low-resolution frames. Its difference with image super-resolution lies in the use of temporal information. Early methods [11, 9, 12, 13] did not consider spatial alignment from frame to frame. Initially, VSR methods applied optical flow-based warping to the images of temporal neighbours [14, 31]. However, inaccurate flows lead to degradation for CNN backbone. To mitigate the impact of inaccurate optical flow, BasicVSR [3] aligns feature maps instead of the input image, while others use the flow to guide deformable convolutions [28, 4, 17] and attention schemes [18].

To increase the robustness toward inaccurate optical flow, [25] propose patch alignment which aligns blocks by averaging the motions within predefined grids. The design of our implicit alignment differs from them as we use

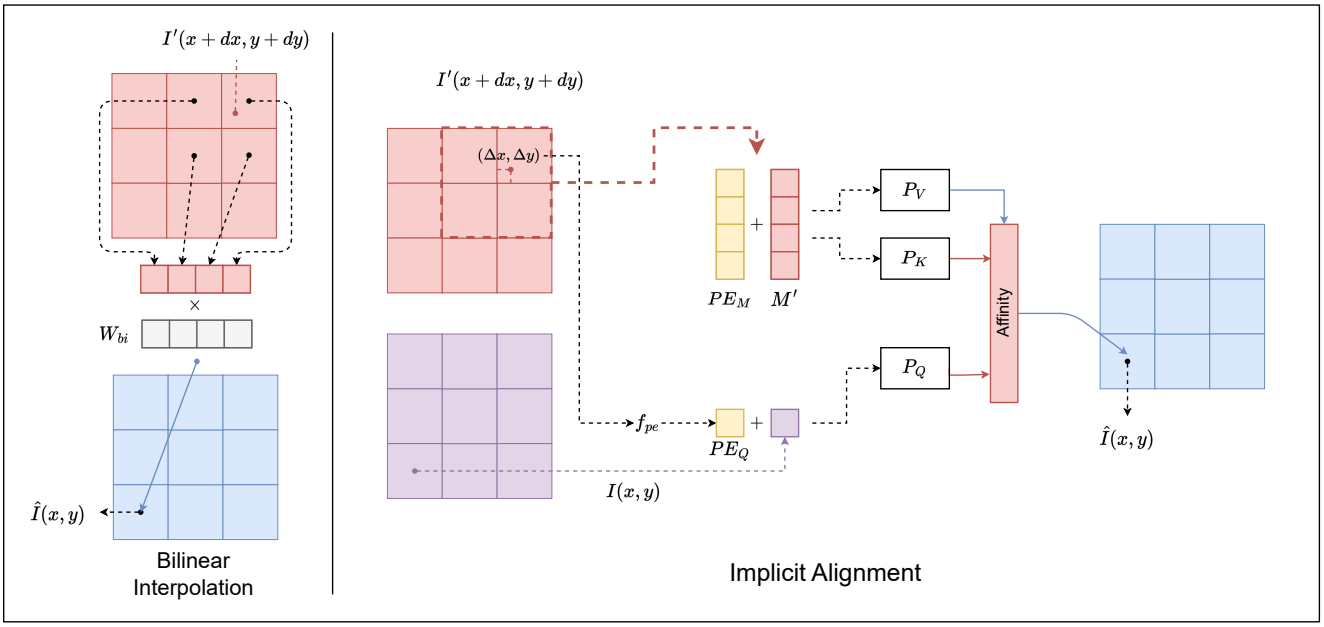


Figure 2: A comparison diagram between bilinear interpolation and our implicit alignment. Bilinear interpolation fixes aggregation weight W_{bi}. Implicit alignment learns affinity through the cross-attention module to calculate the final result. Red grids denote the source frame, purple grids denote the target frame, and blue grids denote the aligned frame.

a flexible window, meaning that each pixel has a different reference window computed by its optical flow.

Image Re-sampling: Aligning the reference frame to the destination frame requires re-sampling sub-pixel values on a discrete reference image. Nearest-neighbour interpolation requires no continuity. Linear interpolation guarantees an L0-smoothness on the resulting image intensity surface [8]; higher orders of polynomials will be required for higher smoothness. It is also well-established that bilinear interpolation will result in blurry outputs [29].

The recent work [7] studied arbitrary-scale image super-resolution. They proposed to learn a continuous representation from the discrete image with an MLP. The neighbouring pixels and the 2D coordinates in the continuous image domain are fed into the network to estimate the pixel value. [30] further improves the representation by adding positional encoding to represent the 2D coordinates and enhance the learning of the high-frequency components.

Our alignment module is also an implicit image representation. The key difference is that our focus is on modelling the temporal aggregation between different frames through cross-attention.

3. Preliminaries

In video super-resolution, inter-frame propagation is a key component to enhance the temporal information aggregation. To facilitate effective propagation, one should first spatially align the source frame with the target frame. the aligned frame $\hat{I}$ is then concatenated with target frame I and together they are fed into the subsequent network block.

Alignment can be performed both on input images and intermediate features. In the following context, we unify the image and feature alignment here as frame-wise alignment and discuss the various implementations.

Frame-wise alignment estimates the spatially matched features in a target frame $I \in \mathbb{R}^{H\times W\times C}$ from a source frame $I' \in \mathbb{R}^{H\times W\times C}$, where C, H and W denote the channels, height and width of the feature vector. For image alignment, C is the image channel. Frame-wise alignment has two steps: motion estimation and motion compensation. Motion estimation predicts the motion offsets $F \in \mathbb{R}^{H\times W\times 2}$, where $F(x,y) = (dx, dy)$ represents the motion offset of each pixel from frame I to frame I'. Given predicted motion offsets, the aligned frame $\hat{I}$ is computed by motion compensation, which is defined by:

$$\hat{I} = \mathcal{W}(I', F) \tag{1}$$

Optical Flow Warping: The most common alignment method is optical flow warping, which is used in [3, 31, 19]. Given an optical flow $F \in \mathbb{R}^{H\times W\times 2}$, where $F(x,y) = (dx, dy)$. The backward warping process is often chosen to propagate pixel/feature values from source frame I' to target frame I. Specifically, the following calculation is it-

erated on all spatial locations:

$$\hat{I}(x,y) = \mathcal{W}_{of}(I', dx, dy) = \sum_{(i,j)} G\big((i,j),(x+dx,y+dy)\big) I'(i,j) \quad (2)$$

The need for re-sampling arises as (dx, dy) is continuous but $I(\cdot,\cdot)$ is defined in discrete space. Common re-sampling methods are bilinear, bicubic and nearest interpolation, where the interpolated pixels are obtained by aggregating neighbouring reference pixels. $G((i,j),\cdot)$ defines the interpolation kernel, which stores the weights to aggregate pixel $I'(i,j)$. For example, a bilinear interpolation kernel is

$$G\big((i,j),(x+dx,y+dy)\big) = g(i,x+dx)\cdot g(j,y+dy), \quad \text{where } g(a,b) = max(0,1-|a-b|). \quad (3)$$

There is no prior knowledge on how the original discrete image is sampled from the real-world. Assumptions on smoothness are often imposed to construct the interpolation kernel. Specifically, nearest interpolation has no smoothness requirements. Bilinear interpolation enforces L0 smoothness and bicubic interpolation enforces L1 smoothness.

Existing works often adopt bilinear interpolation for re-sampling. Yet by enforcing L0 smoothness, the interpolation is effectively applying a low-pass filter to the source frame's features. While nearest-neighbour interpolation does not have such a low-pass effect, it introduces distortions by shifting the sampled position to the nearest pixel grid value.

4. Methodology

4.1. Implicit Alignment

The process of bilinear interpolation or bicubic interpolation can be viewed as imposing a low-pass filter on the original data[32]. We aim to propose an implicit alignment function where the re-sampled value is not obtained by any explicit interpolation function, *e.g.*, a bilinear or bicubic interpolation. The implicit alignment takes the offsets and original values, and outputs a warped value through a small neural network block.

For each pixel $I(x,y)$ in the target frame t, and $I'(x,y)$ in the source framet', given the estimated motion f from t to t', where $f(x,y) = (dx,dy)$ is the offset of the pixel from t to t' frame. Then

$$\hat{I}(x,y) = \mathcal{W}_{ia}(I', x+dx, y+dy) = A(Q,K')V', \quad (4)$$

where A is the affinity matrix which is computed by:

$$A(Q,K') = \text{softmax}\left(\frac{QK'^T}{\sqrt{C_p}}\right). \quad (5)$$

$Q \in \mathbb{R}^{1\times C_q}$ is the encoded query for $I(x,y)$; $K' \in \mathbb{R}^{h\times w\times C_q}$ and $V' \in \mathbb{R}^{h\times w\times C_v}$ are the encoded key and value for source frame I'. h and w are the height and width of the attention window. C_q is the channel number for the query and key encoding and C_v is the channel number for the value encoding. The softmax operation is applied along all the pixels in the windows $h \times w$. The computation of query encoding, key encoding and value encodings are defined as follows.

Query Encoding: The query Q is computed by a query encoder P_Q on the original value $I(x,y)$ and the positional encoding of sub-pixel offset of motion estimation (dx,dy). (see Sec. 4.2):

$$Q = P_Q\left(I(x,y) + PE_Q\right). \quad (6)$$

Key and Value Encoding: A global cross-attention scheme is to choose $K' \in \mathbb{R}^{H\times W\times C_p}$ to encode all pixels in the source frame I', where H and W are the height and width of the feature. However, this choice has quadratic computation complexity with respect to the input image resolution $H \times W$.

We apply a local attention scheme and encode only the pixels in a window around the query position $(x+dx, y+dy)$. Let w be the chosen window size. The selected pixels $M' \in \mathbb{R}^{h\times w\times C}$ are obtained by iterating over $i \in \{0,1...,h\}$ and $j \in \{0,1...,w\}$:

$$M'(i,j) = I'(\lfloor x+dx\rfloor - \lfloor\frac{h}{2}\rfloor + i, \lfloor y+dy\rfloor - \lfloor\frac{w}{2}\rfloor + j). \quad (7)$$

The key K' and value V' are computed by encoding M' and its positional encoding into key space with P_K and P_V:

$$K' = P_K(M + PE_M) \quad (8)$$

$$V' = P_V(M + PE_M), \quad (9)$$

where P_k and P_v are the key and value encoders respectively. We use a single fully connected layer for P_q, P_k and P_v.

4.2. Positional Encoding

Recent works [30, 21] have shown that positional encoding can enhance the network in the high-frequency domain by expanding 2D coordinates into a high-dimensional periodic positional encoding. In order to represent the sub-pixel offset information, we use sinusoidal functions to encode the offsets into the positional encoding $PE_Q \in \mathbb{R}^{1\times C}$ and $PE_M \in \mathbb{R}^{N\times C}$.

The positional encoding for the query takes the sub-pixel offset into the C dimensional sinusoidal representation:

$$PE_Q = f(\Delta x, \Delta y), \tag{10}$$

where

$$\Delta x = dx - \lfloor dx \rfloor, \Delta y = dy - \lfloor dy \rfloor. \tag{11}$$

The positional encoding takes the grid index inside the window to the C dimensional sinusoidal representation:

$$PE_M(i + wj) = f(i - \lfloor \frac{w}{2} \rfloor, j - \lfloor \frac{w}{2} \rfloor). \tag{12}$$

Above, $f(x, y) : \mathbb{R}^2 \rightarrow \mathbb{R}^C$ is the 2D positional encoding function which projects 2D continuous coordinates into high-dimensional space as follows:

$$f(x,y)^{(c)} := \begin{cases} \sin(\omega_k \cdot x) & \text{if } c = 2k \quad \text{and } c \leq C/2, \\ \cos(\omega_k \cdot x) & \text{if } c = 2k+1 \text{ and } c \leq C/2, \\ \sin(\omega_k \cdot y) & \text{if } c = 2k \quad \text{and } c > C/2, \\ \cos(\omega_k \cdot y) & \text{if } c = 2k+1 \text{ and } c > C/2, \end{cases} \tag{13}$$

where c denotes the encoding index. We first normalize (x, y) into $[-\pi/2, \pi]$. The angular speed is defined as:

$$\omega_k = \frac{1}{T^{4k/C}}. \tag{14}$$

In natural language processing, positional encodings were first proposed to represent the discrete text index. Suggested parameter values for the temperature value were in the range of $T = 10000$. This forms a geometric progression from 2π to 10000π on the wavelength [27]. The objective of the positional encoding here is to represent the continuous coordinates and preserve the high-frequency content. As such, set $T = 0.01$, forming a geometric progression from 2π to 0.01π on the wavelength, can represent more precise sub-pixel position information.

Conceptually, the computation of PE_Q and PE_M take $(\lfloor x + dx \rfloor, \lfloor y + dy \rfloor)$ as the anchor and compute the relative offsets. The PE_Q represents the relative sub-pixel offsets for $I'(x + dx, y + dy)$; we add this positional encoding to $I(x, y)$ under the assumption that the query point $I'(x + dx, y + dy)$ has a similar value with $I(x, y)$, given the estimated motion.

4.3. Network Structure

We choose a single-layer fully connected layer as the encoder for P_q, P_k and P_v. The implicit alignment module can directly replace the optical flow warping operation in the existing backbone. Specifically, we conducted experiments on both CNN-based backbone following BasicVSR [3] and Transformer-based backbone following PSRT-recurrent [25]. Alignment is used for either first-order or second-order bidirectional propagation. All implicit alignment modules share the same parameters, which means features of different depths are aligned through one implicit alignment module. The width and height of the attention window are set to 2 for all the experiments unless explicitly stated.

5. VSR on Synthetic Dataset

5.1. Sintel Dataset

As mentioned in [3], image alignment suffers from inaccurate optical flow. In order to focus on the study of interpolation effectiveness and exclude the impact of optical flow accuracy, we propose using a synthetic dataset where the ground-truth optical flow is provided. In order to study the robustness of different alignment methods under different optical flow accuracy, we generated two different optical flows in addition to ground truth optical flow, namely RAFT [26] and SPyNet [24].The former is a precise but slow method, the latter is an efficient method.

The Sintel clean data track is composed by 23 training videos, which we split into 20 training videos and 3 testing videos [2] and report the testing results. We generate the training data by bicubic down-sampling the high-resolution image. As only first-order forward backward optical flow $(t \rightarrow t + 1)$ ground-truth is provided, we perform image alignment from $t + 1 \rightarrow t$ and concatenate with original frame before feeding into residual blocks and the reconstruction network.

5.2. Experimental Settings

We chose VSR transformer as the backbone, where the network consists of a convolution module and two Multi-Freame Self-Attention Blocks [25] followed by an upsampling module. For image alignment, propagation is done on image inputs. For feature alignment, propagation is done after the first SwinIR module. All propagated features are concatenated with the current frame's features and then input to the next network layer. All alignment methods tested have consistent network structure and training parameters, where use Adam optimizer to learn 100k iterations at a learning rate of 2e-4, with a batch size of 8.

We tested three different optical flow settings under the following different alignment methods : 1) without propagation, which is degraded to image super-resolution. 2) Duplicate: propagation without alignment. 3) optical flow warping with bilinear and nearest interpolations. 4) FGDC [28]: flow-guided deformable convolution. 5) FGDA [18]: glow-guided deformable attention. 6) PA [25]: patch alignment. 7) IA: our implicit alignment.

[2] $\texttt{ambush}_5, \texttt{market}_6, \texttt{mountain}_1$

Alignment	Params (M)	Re-samp.	GT Flow (EPE=0)		RAFT Flow (EPE=1.467)		SpyNet Flow (EPE=3.710)	
			Img.	Feat.	Img.	Feat.	Img.	Feat.
-	1.35	-	31.24/0.8867	31.24/0.8867	31.24/0.8867	31.24/0.8867	31.24/0.8867	31.24/0.8867
Duplicate		-	31.50/0.8902	31.42/0.8883	31.50/0.8902	31.42/0.8883	31.50/0.8902	31.42/0.8883
OF Warp	1.35	bilinear	31.60/0.8934	31.92/0.9000	31.60/0.8934	31.87/0.8991	31.60/0.8934	31.85/0.8987
		nearest	31.85/0.8975	31.84/0.8977	31.81/0.8971	31.87/0.8982	31.75/0.8958	31.78/0.8967
FGDC [28]	1.60	bilinear	-	32.08/0.9026	-	31.99/0.9009	-	31.98/0.9005
FGDA [18]	1.56	bilinear	-	32.03/0.9017	-	31.91/0.8998	-	31.94/0.8998
PA [25]	1.35	nearest	-	31.81/0.8971	-	31.85/0.8976	-	31.82/0.8969
IA (ours)	1.36	-	-	**32.14/0.9034**	-	**32.03/0.9018**	-	**32.05/0.9014**

Table 1: PSNR/SSIM Comparison of VSR transformers on Sintel datasets with optical flow of different accuracies for alignment. The best score is marked in **bold**.

5.3. Results Analysis

Tab. 1 shows PSNR/SSIM comparisons on different alignment methods. Unlike [3, 25], we do not find any degradation on image alignment for the synthetic dataset. Even for propagation without alignment, the accuracy also increases. This is likely due to the small displacement between frames in the dataset. Using optical flow warping for alignment further improves the results, but different re-sampling results perform differently and we discuss these results in the following context.

The effect of re-sampling: An interesting finding is that nearest-neighbour interpolation for image alignment results in better performance than bilinear interpolation, but the opposite is true for feature alignment. We think there are two reasons for this phenomenon: 1) Image alignment can provide the most discriminative comparison on the re-sampling methods because the frequency components are not yet smoothed out by convolution layers. In this scenario, the preservation of high-frequency components by nearest interpolation outweighs the distortions introduced by it. 2) Feature alignment shows fewer differences towards high-frequency components due to the bias towards low-frequency signals of neural networks [23, 6]. In this case, the distortions of nearest interpolation contributes to the performance difference.

Based on this observation, we conclude that an effective re-sampling method should combine the advantages of nearest interpolation and bilinear interpolation. Specifically, one should not impose any smoothness constraint on the interpolation to avoid a smoothing effect on the high-frequency components, and avoid distortions caused by coordinates quantization.

Comparison with SOTA alignment methods: We compare implicit alignment with FGDC, FGDA, and PA for feature alignment. Patch Alignment (PA) is more robust than others against inaccurate optical flow, showing no accuracy drops when optical flow degrades. However, it sacrifices the overall accuracy because patch alignment cannot utilize sub-pixel or even sub-grid motions. FGDA and FGDC are essentially ensemble modules for multiple optical flow warping, which provide extra improvements but still suffer from the disadvantage of bilinear re-sampling. Implicit alignment outperforms all three SOTA alignment methods because it learns the re-sampling weights implicitly, while the others adapt bilinear and nearest interpolation which introduces smoothing prior or distortions.

6. VSR on Real-World Dataset

6.1. Experimental Settings

To show the effectiveness of implicit alignment on large-scale datasets, we incorporate it into state-of-the-art CNN-based (BasicVSR [3]) and Transformer-based backbones (PSRT-recurrent [25]). We follow the training configurations of BasicVSR: 300K training iterations on the REDS datasets for BI degradation, and 150K iterations on the Vimeo-90K datasets for BD and BI degradation. Results are reported on REDS4 for the REDS BI model in RGB. For the Vimeo-90K BI model, we report accuracy on Vimeo-90K-T and Vid4. For the Vimeo-90K BD model, we report accuracy on UDM10, Vimeo-90K-T and Vid4. For PSRT-recurrent, we follow the configurations from the original paper, where the model is trained with 6 input frames for 300k iterations.

6.2. Comparison with SOTA

Tab. 2 shows a quantitative comparison with SOTA methods. For CNN-based models, Implicit alignment (IA-CNN) outperforms BasicVSR using the same network backbone. The increased parameters (2.2M) come from the computation to calculate the current features, which is required by its implicit alignment module (BasicVSR computes each frame's features recurrently which already include information from other frames). This implicit alignment module increase only 0.01M parameters. For Transformer-based models, implicit alignment restoration transformer (IA-RT) achieves the highest accuracy for the

Method	Frames REDS/Vimeo	Params (M)	REDS4 PSNR	REDS4 SSIM	Vimeo-90K-T PSNR	Vimeo-90K-T SSIM	Vid4 PSNR	Vid4 SSIM
TOFlow [31]	5/7	-	27.98	0.7990	33.08	0.9054	25.89	0.7651
EDVR [28]	5/7	20.6	31.09	0.8800	37.61	0.9489	27.35	0.8264
MuCAN [16]	5/7	-	30.88	0.8750	37.32	0.9465	-	-
BasicVSR [3]	15/14	6.3	31.42	0.8909	37.18	0.9450	27.24	0.8251
IA-CNN (ours)	15/14	8.5	31.68	0.8959	37.34	0.9463	27.42	0.8315
VSR-T [2]	5/7	32.6	31.19	0.8815	37.71	0.9494	27.36	0.8258
VRT [17]	6/-	30.7	31.60	0.8888	-	-	-	-
PSRT-sliding [25]	5/-	14.8	31.32	0.8834	-	-	-	-
PSRT-recurrent [25]	6/-	10.8	31.88	0.8964	-	-	-	-
IA-RT (ours)	6/-	13.4	**32.15**	**0.9010**	-	-	-	-
BasicVSR++ [4]	30/14	7.3	32.39	0.9069	37.79	0.9500	27.79	0.8400
VRT	16/7	35.6	32.19	0.9006	38.20	0.9530	27.93	0.8425
RVRT [18]	30/14	10.8	32.75	0.9113	38.15	0.9527	27.99	0.8462
PSRT-recurrent [25]	16/14	13.4	32.72	0.9106	38.27	0.9536	28.07	0.8485
IA-RT (ours)	16/-	13.4	**32.90**	**0.9138**	-	-	-	-

Table 2: Quantitative comparison on the REDS4 [22] dataset, Vid4 [20], Vimeo-90K-T [31] dataset for $4\times$ VSR task. The best score is marked in **bold**.

Figure 3: Qualitative comparison on REDS4 dataset. We highlight the detail regions with yellow boxes. Compared with BasicVSR, IA-CNN provides more details on the wall and more uniform patterns on the window.

6 training input frames configuration. Note that implicit alignment module only increases parameters by 0.04M compared to the released model by [25], which is 13.37; however, we follow the original paper and report 10.8 in the table for PSRT-recurrent.

6.3. Ablation Studies

We conduct ablation experiments using a smaller-scale model and training setup on REDS dataset. The supplementary material provides more details.

Positional Encoding: Tab. 3 shows the results of ablation studies on the components of the positional encoding. The positional encoding improves the PSNR by 0.28 compared

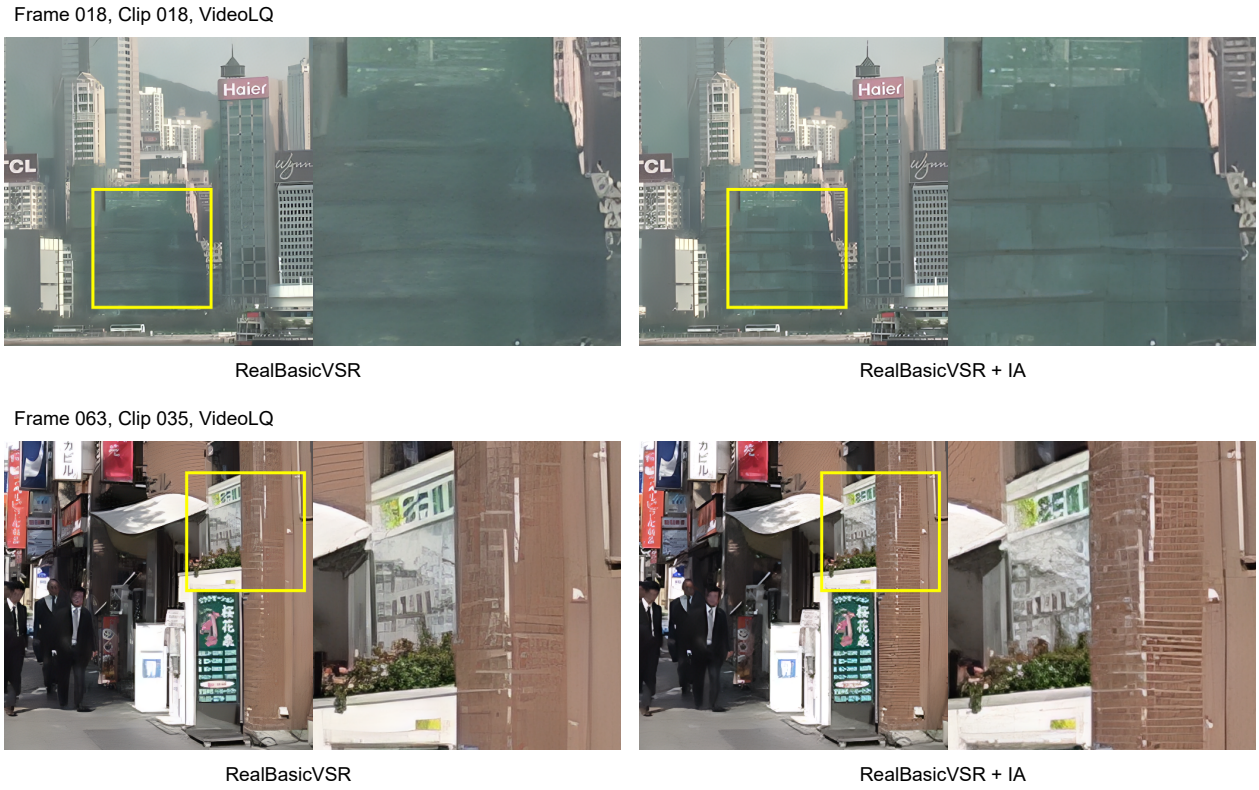


Figure 4: Qualitative comparison on VideoLQ dataset. Our proposed IA method recovers the building details and the brick textures, which ReadlBasicVSR does not recover. We highlight the detail regions with yellow boxes.

to the naive window-based cross-attention. Adding positional encodings only to key and value slightly enhances the performance, which can be interpreted as a case where the estimated motion is quantized to integers. However, adding positional encodings to key and value encoders leads to model collapse, as the relative positional information for the neighboring pixels is missing.

Window Size: The PSNR/SSIM results for different window sizes of the cross-attention operation are presented in Tab. 4. We observe a larger window have a bigger receptive field, but they also reduce the alignment quality due to additional noise. A window size of 4 causes model divergence during training. However, we discovered that for Real-world VSR where predicting accurate optical flow is difficult, a larger window size will improve to increase the robustness of the model.

6.4. Qualitative Results

Sec. 5.3 shows qualitative comparisons between BasicVSR and IA-CNN on the REDS4 dataset. The implicit alignment can propagate more high-frequency contents and reconstruct finer patterns than the baseline. See Supplementary for more qualitative results.

PE_Q	PE_M	PSNR	SSIM
✗	✗	30.43	0.8700
✓	✗	28.71	0.8184
✗	✓	30.54	0.8730
✓	✓	30.71	0.8776

Table 3: Ablations on positional encodings. The positional encoding improves the alignment effectiveness compared to the naive window-based cross-attention.

Window Size	2	3	4
PSNR/SSIM	30.71/0.8776	30.67/0.8775	*diverge*

Table 4: PSNR/SSIM for different window sizes.

6.5. Real-World Video Super-Resolution

For real-world VSR with unknown degradation, existing methods tend to produce overly smoothed results due to the lack of high-frequency content. Since the degradation is unknown, there is no fixed re-sampling method that can be effective. We demonstrate that our methods can enhance the high-frequency content when applied on top of RealBasicVSR [5]. As the optical flow computed on the degraded

image is often inaccurate, we use a window size of 4 to increase the receptive field and thus the information aggregation. Fig. 4 shows qualitative comparison. When embedded in RealBasicVSR, our implicit alignment can produce more realistic and fine-grained results. Please refer to the Supplementary for more qualitative results.

7. Conclusion

This paper studies the impact of re-sampling on alignment for video super-resolution through experiments on a synthetic dataset utilizing ground-truth optical flow. We find that a re-sampling should preserve the original sharpness of the features and prevent distortions for an effective alignment. We further propose an implicit alignment through window-based cross-attention with estimated motions encoded into positional encoding. Our method outperforms SOTA alignments on both synthetic and real-world datasets. One of the limitations of implicit alignment is that it cannot be combined with multiple offsets, making it hard to benefit from ensemble motion predictions. The possible feature work could be embedding multiple motion offsets into positional encoding for more accurate motion estimation.

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

[2] Jiezhang Cao, Yawei Li, Kai Zhang, and Luc Van Gool. Video super-resolution transformer. *arXiv preprint arXiv:2106.06847*, 2021. 7

[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, 2, 3, 5, 6, 7

[4] Kelvin CK Chan, Shangchen Zhou, Xiangyu Xu, and Chen Change Loy. Basicvsr++: Improving video super-resolution with enhanced propagation and alignment. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5972–5981, 2022. 1, 2, 7

[5] Kelvin CK Chan, Shangchen Zhou, Xiangyu Xu, and Chen Change Loy. Investigating tradeoffs in real-world video super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5962–5971, 2022. 8

[6] Yuanqi Chen, Ge Li, Cece Jin, Shan Liu, and Thomas Li. Ssd-gan: measuring the realness in the spatial and spectral domains. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 1105–1112, 2021. 6

[7] Yinbo Chen, Sifei Liu, and Xiaolong Wang. Learning continuous image representation with local implicit image function. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8628–8638, 2021. 2, 3

[8] Neil Anthony Dodgson. Image resampling. Technical report, University of Cambridge, Computer Laboratory, 1992. 3

[9] Dario Fuoli, Shuhang Gu, and Radu Timofte. Efficient video super-resolution through recurrent latent space propagation. In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, pages 3476–3485. IEEE, 2019. 2

[10] Muhammad Haris, Gregory Shakhnarovich, and Norimichi Ukita. Recurrent back-projection network for video super-resolution. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3897–3906, 2019. 1

[11] Yan Huang, Wei Wang, and Liang Wang. Video super-resolution via bidirectional recurrent convolutional networks. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):1015–1028, 2017. 2

[12] Takashi Isobe, Xu Jia, Shuhang Gu, Songjiang Li, Shengjin Wang, and Qi Tian. Video super-resolution with recurrent structure-detail network. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XII 16*, pages 645–660. Springer, 2020. 2

[13] Takashi Isobe, Fang Zhu, Xu Jia, and Shengjin Wang. Revisiting temporal modeling for video super-resolution. *arXiv preprint arXiv:2008.05765*, 2020. 2

[14] Tae Hyun Kim, Mehdi SM Sajjadi, Michael Hirsch, and Bernhard Scholkopf. Spatio-temporal transformer network for video restoration. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 106–122, 2018. 2

[15] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4681–4690, 2017. 2

[16] Wenbo Li, Xin Tao, Taian Guo, Lu Qi, Jiangbo Lu, and Jiaya Jia. Mucan: Multi-correspondence aggregation network for video super-resolution. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part X 16*, pages 335–351. Springer, 2020. 7

[17] 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, 7

[18] Jingyun Liang, Yuchen Fan, Xiaoyu Xiang, Rakesh Ranjan, Eddy Ilg, Simon Green, Jiezhang Cao, Kai Zhang, Radu Timofte, and Luc Van Gool. Recurrent video restoration transformer with guided deformable attention. *arXiv preprint arXiv:2206.02146*, 2022. 1, 2, 5, 6, 7

[19] Jing Lin, Xiaowan Hu, Yuanhao Cai, Haoqian Wang, Youliang Yan, Xueyi Zou, Yulun Zhang, and Luc Van Gool. Unsupervised flow-aligned sequence-to-sequence learning for

video restoration. In *International Conference on Machine Learning*, pages 13394–13404. PMLR, 2022. 1, 3

[20] Ce Liu and Deqing Sun. On bayesian adaptive video super resolution. *IEEE transactions on pattern analysis and machine intelligence*, 36(2):346–360, 2013. 7

[21] 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. *Communications of the ACM*, 65(1):99–106, 2021. 4

[22] 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. 7

[23] Nasim Rahaman, Aristide Baratin, Devansh Arpit, Felix Draxler, Min Lin, Fred Hamprecht, Yoshua Bengio, and Aaron Courville. On the spectral bias of neural networks. In *International Conference on Machine Learning*, pages 5301–5310. PMLR, 2019. 6

[24] Anurag Ranjan and Michael J Black. Optical flow estimation using a spatial pyramid network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4161–4170, 2017. 5

[25] 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, 5, 6, 7

[26] 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. 5

[27] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 5

[28] 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. 1, 2, 5, 6, 7

[29] G Wolberg. Digital image warping: Ieee computer society, 1990. 3

[30] Xingqian Xu, Zhangyang Wang, and Humphrey Shi. Ultrasr: Spatial encoding is a missing key for implicit image function-based arbitrary-scale super-resolution. *arXiv preprint arXiv:2103.12716*, 2021. 2, 3, 4

[31] 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, 2, 3, 7

[32] Abdou Youssef. Analysis and comparison of various image downsampling and upsampling methods. In *Proceedings DCC'98 Data Compression Conference (Cat. No. 98TB100225)*, page 583. IEEE, 1998. 4