Satellite Video Super-Resolution via Multiscale Deformable Convolution Alignment and Temporal Grouping Projection

Yi Xiao, Xin Su, Qiangqiang Yuan, *Member, IEEE*, Denghong Liu, Huanfeng Shen, *Senior Member, IEEE*, and Liangpei Zhang, *Fellow, IEEE*

***Abstract*— As a new earth observation tool, satellite video has been widely used in remote-sensing field for dynamic analysis. Video super-resolution (VSR) technique has thus attracted increasing attention due to its improvement to spatial resolution of satellite video. However, the difficulty of remote-sensing image alignment and the low efficiency of spatial–temporal information fusion make poor generalization of the conventional VSR methods applied to satellite videos. In this article, a novel fusion strategy of temporal grouping projection and an accurate alignment module are proposed for satellite VSR. First, we propose a deformable convolution alignment module with a multiscale residual block to alleviate the alignment difficulties caused by scarce motion and various scales of moving objects in remote-sensing images. Second, a temporal grouping projection fusion strategy is proposed, which can reduce the complexity of projection and make the spatial features of reference frames play a continuous guiding role in spatial–temporal information fusion. Finally, a temporal attention module is designed to adaptively learn the different contributions of temporal information extracted from each group. Extensive experiments on Jilin-1 satellite video demonstrate that our method is superior to current state-of-the-art VSR methods.**

***Index Terms*— Deformable convolution, satellite video, super-resolution (SR), temporal attention (TA), temporal grouping projection.**

I. Introduction

HIGH-RESOLUTION (HR) remote-sensing image with rich detailed information has been widely used in object tracking [1] and land-cover classification [2]. However, due to the limitation of sensors and the degradation of data transmission, the spatial resolution of satellite video will be decreased to some extent, which hinders the application of satellite video. Therefore, super-resolution (SR) technique is urgently needed to recover HR images from the corresponding low-resolution (LR) images [3]–[5]. SR can be understood as a post-processing technology, which breaks through the sensor's resolution limitation and algorithmically obtains a higher-resolution image. According to the number of LR images used to reconstruct the HR image, SR is divided into single-image SR (SISR) [6]–[10], multiimage SR (MISR) [11], [12], and video super-resolution (VSR) [13]–[15].

SR is a typical ill-posed inverse problem [16] because one LR image may correspond to multiple HR images, which does not satisfy the uniqueness of the solution. To constrain the solution space, many SISR methods have been proposed and are divided into three categories: interpolation-based methods, reconstruction-based methods, and learning-based methods. Interpolation-based methods include nearest interpolation, bicubic interpolation, and the Lanczos resampling. They are fast in the calculation, but may exhibit fuzzy and jagged artifacts on the edges of the object. The reconstruction-based method constrains the solution space of the HR images by adding external constraint information (such as total variation, gradient prior, and sparse prior) and then solves the SR problem iteratively. A typical reconstruction-based method is to incorporate both SR and regularization parameters into the Bayesian framework [17]. It converts the SR problem into maximizing the probability of obtaining HR image under the condition of the existence of LR image. Although the reconstruction-based method can obtain sharper texture information, their computational complexity is very high. The learning-based method constructs a large number of LR–HR image pairs and learns the mapping relationship between LR images and HR images from the sample database. A typical method is sparse coding [18], [19]. LR images are sparsely coded on the LR dictionary to obtain sparse coefficients. Based on the assumption that the manifold space of LR and HR images is consistent, the sparse coefficients learned from the LR dictionary are applied to the HR dictionary to reconstruct an HR image. With the rise of deep learning, convolutional neural networks (CNNs) are widely used in remote-sensing image processing [20], such as cloud removal [21], image denoising [22], hyperspectral image (HSI) restoration [23], [24], anomaly detection [25], and classification [26]. For the

Manuscript received May 12, 2021; revised July 11, 2021 and August 5, 2021; accepted August 18, 2021. This work was supported in part by the National Natural Science Foundation of China under Grant 41922008 and Grant 61971319. *(Corresponding author: Xin Su.)*

Yi Xiao, Qiangqiang Yuan, and Denghong Liu are with the School of Geodesy and Geomatics, Wuhan University, Wuhan 430079, China (e-mail: xiao_yi@whu.edu.cn; yqiang86@gmail.com; dhliu77@whu.edu.cn).

Xin Su is with the School of Remote Sensing and Information Engineering, Wuhan University, Wuhan 430079, China (e-mail: xinsu.rs@whu.edu.cn).

Huanfeng Shen is with the School of Resource and Environmental Sciences, Wuhan University, Wuhan 430079, China (e-mail: shenhf@whu.edu.cn).

Liangpei Zhang is with the State Key Laboratory of Information Engineering in Surveying, Mapping, and Remote Sensing, Wuhan University, Wuhan 430079, China (e-mail: zlp62@whu.edu.cn).

Digital Object Identifier 10.1109/TGRS.2021.3107352

SR tasks, the CNN has brought SR performance to new heights due to its powerful ability of fitting nonlinear relationships. By building a dataset with a large number of LR–HR image pairs, the neural network can learn the complex nonlinear mapping relationship between LR and HR images from the training set.

Compared to SISR, video SR (VSR) is more complex. The information used by the SISR method is limited to the spatial domain of the LR image, while the VSR needs to consider both spatial and temporal information. Therefore, the key to VSR is how to use the redundant information among frames. Specifically, given an LR frame which needs to be super-resolved (reference frame) and its surrounding LR neighboring frames (supporting frames), the VSR methods need to model the spatial–temporal relationship between frames simultaneously. In particular, accurately modeling the temporal relationship between the reference frame and the supporting frames is critical for the success of the VSR. Thus, the primary task of VSR is to align the supporting frames to the reference frame such that the redundant information in the supporting frames can be fully exploited. Currently, the state-of-the-art (SOTA) methods employ optical flow to achieve explicit [27]–[30] alignment and deformabale convolution (DConv) to achieve implicit [4], [31]–[33] alignment. Specifically, explicit alignment performs at the image level. The optical flow between a supporting frame and the reference frame is first calculated to complete the motion estimation. Then the optical flow is used to warp the supporting frames to complete the motion compensation. Implicit alignment performs at the feature level. Sampling parameters are learned from the features of supporting frames and the reference frame and then are fed into DConv together with the features of supporting frames to achieve adaptive alignment.

Despite the current VSR methods have achieved significant results in natural videos, they are still not suitable for direct application on satellite video. First, due to the limitation of sensor resolution, the resolution of satellite video frames is lower compared to natural video. The satellite video frame lacks sufficient texture and detail information, making feature extraction more difficult. Second, the remote-sensing image has a larger width. The frame is of higher complexity due to moving objects with various scales. Finally, the satellite video contains scarce motion information. Moving objects only occupy very few pixels in remote-sensing images, resulting in the difficulty of capturing motion information. These three points make it difficult to achieve precise alignment and effective spatial–temporal information fusion in satellite VSR. Specifically, optical flow still suffers from large motion or motion with various scales due to the requirement of small motions as a hypothesis. Also, optical flow calculation is computationally intensive and sometimes independent of network parameters, such as PyFlow [30] used in RBPN [29]. In this case, the misalignment caused by incorrect optical flow estimation will affect the subsequent fusion. To a certain extent, DConv can solve the alignment of multiscale moving objects through adaptive alignment. But existing DConv-based methods adopt convolution with a limited receptive field to generate sampling parameters or a pyramid structure with a large amount of computation to increase the receptive fields. The sampling parameters learned in these way is either not accurate enough to model the temporal relationship between supporting frames and the reference frame or computationally excessive. In addition, the strided convolution used by the pyramid structure will further lose detailed information of object boundaries [34], thus reducing the performance of alignment. Therefore, conventional VSR methods suffer severe performance drop when directly applying to satellite video. Besides, most of the current methods directly fuse the aligned features to achieve spatial–temporal information fusion. But in remote-sensing images, due to the small number of pixels occupied by moving objects, the complementary information provided by supporting frames is limited and difficult to be extracted. Thus, spatial information of the reference frame should play a leading role in the fusion of spatial–temporal information. If the aligned features are simply fused, the guiding role of the reference frame will be weakened with the deepening of the network, which is not conducive to the effective integration of spatial–temporal information in satellite video.

To solve the problems mentioned above, this article proposes a VSR network for satellite video based on multiscale deformable (MSD) convolution alignment and temporal grouping projection. The main contributions of this article are as follows:

1) A fusion strategy of *temporal group projection (TGP)* is proposed. Through the projection, the network focuses on learning temporal information and constantly supplements the spatial information of the reference frame, realizing the efficient fusion of spatial–temporal information. Our temporal grouping strategy helps the network learn more complementary information from supporting frames while reduces the complexity of the projection.
2) An *MSD convolution alignment module* is proposed. We designed a *multiscale residual block (MSRB)* as the generator of sampling parameters of deformable convolution kernels to learn more complex motion information for precise alignment of satellite video frames. Finally, we adopted a temporal attention (TA) module to take into account the different contributions of the temporal information.

The remaining part of this article is arranged as follows. In Section II, we introduce the existing works related to SR. The details of our proposed network will be presented in Section III. In Section IV, we present the experimental results on the satellite video data of Jilin-1 and make a meticulous analysis of the results. Finally, we summarize the work of this article in Section V.

II. Related Work

A. Deep Learning-Based Single-Image SR

The earliest SR work based on deep learning was inspired by sparse coding. Dong *et al*. [35] proposed the first end-to-end CNN named SRCNN. Subsequently, to solve the difficult training problem brought by the deepening of network depth,

Kim *et al*. [36] proposed VDSR inspired by the residual network of He *et al*. [37] and introduced residual learning into SISR for the first time, increasing the number of layers of the network to 20. Both SRCNN and VDSR used interpolated LR images as the input of the network. In order to avoid such preprocessing operations to train an end-to-end model directly, Shi *et al*. [16] put forward ESPCN by adding a subpixel convolution at the end of the network to complete the up-sampling operation. Since feature calculations perform in an LR space, ESPCN greatly reduces the computational cost. Lim *et al*. [38] combined the advantages of residual learning and subpixel convolution further proposed an enhanced deep SR network EDSR. The authors improved the structure of the traditional residual block by removing the batch normalization (BN) layer and proved that the presence of BN would reduce the reconstruction performance in SR tasks. Based on EDSR, Yu *et al*. [39] proposed a WDSR using a wide activation strategy, which increases the width of the feature map (the number of channels) before the activation operation, to learn more feature information after the activation operation. In addition to the residual network, the dense network is also an effective structure in SR tasks. The RDN network [40] combined the structure of a residual block and dense block to form a residual dense block. The output of each residual dense block would eventually be concatenated and fused. However, densely connected networks will bring a huge amount of parameters and computational burden. For this reason, Jiang *et al.* [41] proposed a hierarchical dense connection network by designing a hierarchical dense residual block (HDB) to enhance feature representation while saving computational memory. At present, DBPN [42] using back projection and RCAN [43] using residual channel attention have achieved SOTA performance in SISR. DBPN came up with a back projection network, which provides an error feedback mechanism during each iterating up-down sampling. Each up-down sampling module represents the different SR and degradation components of the image. Compared with the traditional feedback network, the back projection mechanism is more in line with human vision law. RCAN proposed a novel residual in the residual structure and introduced channel attention into SR. For more details about these methods, refer to [44].

The SISR method only considers the spatial information of the LR image. Thus, the rich temporal information in the sequence of frames cannot be utilized. In addition, due to the failure to consider the temporal relationship between frames, the recovered video would suffer from the temporal inconsistency problem, which indicates video content flickering across different frames [34]. To make better use of spatial–temporal information, VSR method has been developed gradually.

B. Deep Learning-Based VSR

The early VSR method is based on the optical flow method to achieve explicit motion estimation and motion compensation. Based on ESPCN, VESPCN [28] calculated the optical flow between supporting frames and the reference frame in a coarse-to-fine manner and used optical flow to warp supporting frames to achieve motion compensation. The compensated frames were sent to a series of convolutional layers for feature extraction and fusion, and finally, an HR reference frame is obtained through subpixel up-sampling. Inspired by DBPN, Haris *et al*. [29] put forward a kind of recurrent back projection network named RBPN, which concatenates supporting frames with reference frames and their optical flow and then completes adaptive alignment through residual blocks. Then HR features are obtained through the back projection module. Each supporting frame participates in a projection process once, and finally, the results of each projection are aggregated to obtain an HR reference frame. Considering that the optical flow calculation usually occurs between LR frames, and the resolution conflict between LR optical flow and HR output hinders the restoration of details, Wang *et al*. [45] super-resolve optical flow and LR frames simultaneously. With HR optical flow, alignment happens in HR space, making the alignment more accurate.

Dai *et al*. [46] propose the concept of deformable convolution for the first time, which is introduced into VSR tasks by Tian *et al*. [32]. Specifically, TDAN [32] first concatenates the features of the reference frame and supporting frame to learn the offset through several convolutional layers. The offset is used to guide the deformable convolution to perform implicit alignment on the supporting frame features. Inspired by TDAN, Wang *et al*. [33] proposed an alignment module with Pyramid, Cascading, and Deformable convolution (PCD) to learn multilevel offsets through the pyramid structure and performing a coarse alignment on supporting frame features in three levels. Finally, EDVR cascades a deformable convolution operation in supporting frame features in the first level for fine alignment. After alignment, the authors designed a temporal–spatial attention (TSA) module to fully consider the contribution of the aligned features and finally fuses the aligned features to reconstruct a HR reference frame. Apart from deformable convolution, Yi *et al*. [4] proposed a PFNL that uses a nonlocal structure for implicit alignment. They also proposed a progressive fusion strategy to merge spatial–temporal information more effectively rather than directly fuse the aligned features. A similar idea is also used in the multistage feature fusion network proposed by Song *et al*. [34]. Lin *et al*. [47] proposed a flow-guided deformable module (FDM) to integrate optical flow into deformable convolution, thus solved the problem that deformable alignment methods suffer from fast motion and lack explicit motion constraints. Most of the above methods rely on accuracy motion estimation. Thus, Zhu *et al*. [48] proposed a method named STMN which works in the wavelet domain to reduce the dependence on motion estimation.

Recently, Chen *et al.* [49] begun to study the essential components of VSR framework and proposed the BasicVSR that adopts bidirectional propagation with feature alignment to effectively exploit information from the entire input video. On the basis of this work, they further proposed the BasicVSR++ [50] which adopts second-order grid propagation and flow-guided deformable alignment. At present, the vast majority of methods use either a iterative structure or a recurrent structure to process LR frames, to consider the two

structures simultaneously, Yi *et al.* [51] proposed an hybrid omniscient framework to not only utilize the preceding SR output, but also leverage the SR outputs from the present and future, thus achieved SOTA performance.

C. Satellite VSR

Satellite video not only has high temporal redundancy like the natural video, but also has the wide scene characteristics of traditional remote-sensing imaging. The difficulty of data acquisition and the higher complexity of the scene make the research of SR on satellite video still in its infancy. Satellite video contains multitemporal images with abundant information, and it provides a novel data source and application scenario for VSR.

Early work [52] usually just retrained the VDSR model on remote-sensing images. Inspired by SRCNN, Xiao *et al*. [53] proposed a five-layer end-to-end network without any preprocessing and post-processing and used Jilin-1 satellite video for training. Subsequent methods began to take more account of the characteristics of remote-sensing images. To finely learn the structural information and low-level features in wide scenes, Jiang *et al*. [54] proposed a PECNN network, which learns features at different levels through two subnetworks. Zhang *et al*. [8] proposed a scene-adaptive strategy to learn the features of different scenes and realized multilevel feature extraction through a multiscale activation feature fusion module. Jiang *et al*. [55] used a GAN-based network to enhance high-frequency edge information in satellite video. First, an intermediate result with noise is generated through a subnetwork and then the second subnetwork is used to filter the noise and enhance the edge contour information. Finally, the results of the two subnetworks are merged to obtain a result with sharper edge information. Recently, Lei and Shi [56] proposed a hybrid-scale self-similar network HSENet that simultaneously considers the similarity of single-scale and cross-scale targets in remote-sensing images. Although these methods have made remarkable success, most of them are SISR methods that rely on the constraints of spatial information. Thus, it is necessary to carry out collaborative modeling of spatial–temporal information and develop a VSR method for satellite video. Liu *et al*. [57] proposed a traditional VSR method based on the prediction of nonlocal similarity in the adaptive spatial–temporal domain. By adaptively using the spatial–temporal domain to represent local prior knowledge, implicit motion estimation is completed. The local spatial similarity is integrated into the SR framework to enhance the texture details, and finally, it is solved by iterative reweighted least squares. He and He [58] proposed a network for arbitrary scale SR. First, the feature extracting module accepts multiple LR frames. Then, they use numerous 3-D residual blocks to extract features of these frames and finally use subpixel convolution to enhance the spatial resolution. This method utilizes 3-D convolution to realize the modeling of the spatial–temporal relationship adaptively. Once the input frames are not well aligned, the direct fusion of the spatial–temporal information of multiple frames can easily introduce excessive noise and affect the performance. Therefore, simply stacking 3-D convolution is not sophisticated enough in modeling spatial–temporal relations.

At present, there are still few works in the literature about satellite VSR. The SISR method does not make good use of the highly redundant information in the temporal dimension. The existing VSR method cannot accurately model the spatial–temporal relationship. Thus, it is significant to design a VSR network to precisely realize satellite video frame alignment and spatial–temporal information fusion. In this article, accurate alignment can be achieved by an MSD convolution alignment module, and effective spatial–temporal information fusion can be completed by temporal grouping projection.

III. Methodology

A. Overview of the Proposed Framework

Given continuous $2N+1$ LR frames $\mathrm{I} = \{I_{t-N}^{\mathrm{LR}}, \ldots, I_{t-1}^{\mathrm{LR}}, I_t^{\mathrm{LR}}, I_{t+1}^{\mathrm{LR}}, \ldots, I_{t+N}^{\mathrm{LR}}\}$, we define the center frame $I_t^{\mathrm{LR}} \in \mathbb{R}^{h\times w\times c}$ as the reference frame which needs to be super-resolved, and the remaining $2N$ frames as supporting frames. Here, $c = 3$ represents the RGB channels, and h and w represent the height and width of the LR frame, respectively. The goal of our network f_{Ours} is to obtain an HR reference frame $I_t^{\mathrm{SR}} \in \mathbb{R}^{(h\times r)\times(w\times r)\times c} = \mathbb{R}^{H\times W\times c}$ for the input I so that it is close enough to the ground-truth I_t^{HR}, where r is the scale factor. It can be expressed as

$$I_t^{\mathrm{SR}} = f_{\mathrm{Ours}}(\mathrm{I}). \tag{1}$$

Our network structure diagram is shown in Fig. 1. Take continuous input of five frames as an example and define the center frame I_t^{LR} as the reference frame (marked by a red box). First, we divide the frame sequence into two groups according to the temporal distance from the reference frame. The first group contains the reference frame I_t^{LR} and two supporting frames I_{t-1}^{LR} and I_{t+1}^{LR} (marked by green boxes) closest to I_t^{LR}, and the second group contains I_t^{LR} and the two supporting frames I_{t-2}^{LR} and I_{t+2}^{LR} (marked by blue boxes) that are farthest from I_t^{LR}.

After feature extraction, the features in each group are aligned through our MSD alignment module. Then the aligned features are merged to obtain the LR features $M = \{M_1, \ldots, M_n\}, n \in [1, N]$ that encode the multiframe information.

Subsequently, the network integrates spatial–temporal information through projection module. Each projection will get a projection result T_n. Finally, the HR temporal feature $T = \{T_1, \ldots T_n\}, n \in [1, N]$ and the HR spatial feature S_N^{HR} obtained from the last projection will be sent to a TA module for modulation. Finally, the modulated features $\tilde{S}_N, \tilde{T}$ are fused through a convolutional block to obtain the final HR reference frame, which is expressed as

$$I_t^{\mathrm{SR}} = \mathrm{Conv}\left(\tilde{S}_N, \tilde{T}\right). \tag{2}$$

The configuration details of each module in our network are shown in Table I, and the algorithm of our network is described in Algorithm 1.

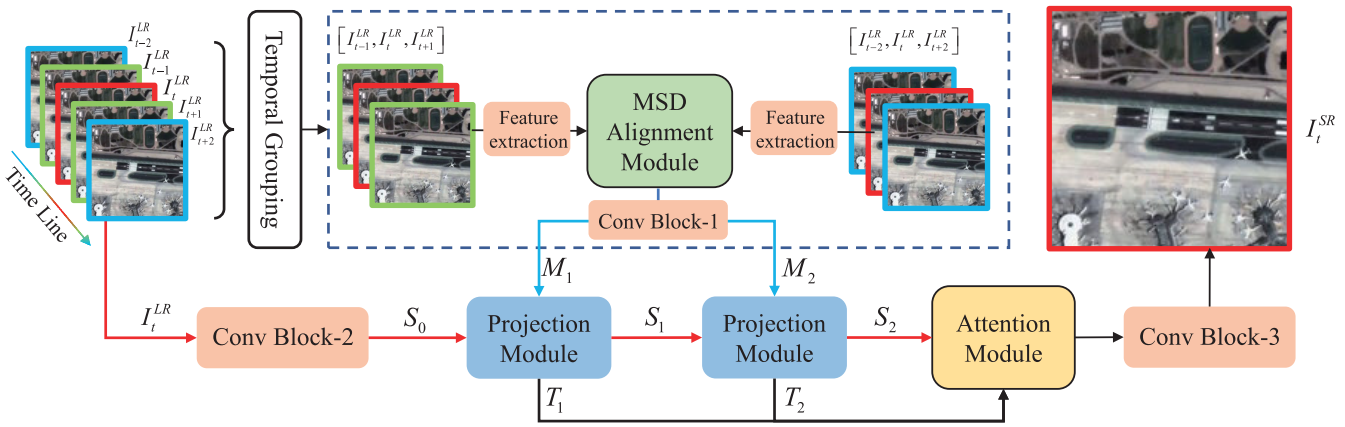


Fig. 1. Overall structure of our proposed network, take $N = 2$ as an example.

TABLE I

CONFIGURATION DETAILS OF OUR NETWORK ARCHITECTURE. $2N + 1$ REPRESENTS THE NUMBER OF FRAMES INPUT, AND h AND w ARE THE HEIGHT AND WIDTH OF LR FRAMES, RESPECTIVELY. H AND W ARE THE HEIGHT AND WIDTH OF HR FRAMES, RESPECTIVELY

Module	Layer		Input Size → Output Size	Kernel Size	Stride	Padding
Feature Extraction	Conv-Block (2D Conv+PReLU)		$(2N+1)\times 3\times h\times w \rightarrow (2N+1)\times 64\times h\times w$	3×3	1	1
	ResBlock-1 (2D Conv1+PReLU+2D Conv2)		$(2N+1)\times 64\times h\times w \rightarrow (2N+1)\times 64\times h\times w$	3×3	1	1
	ResBlock-2 (2D Conv1+PReLU+2D Conv2)		$(2N+1)\times 64\times h\times w \rightarrow (2N+1)\times 64\times h\times w$	3×3	1	1
	ResBlock-3 (2D Conv1+PReLU+2D Conv2)		$(2N+1)\times 64\times h\times w \rightarrow (2N+1)\times 64\times h\times w$	3×3	1	1
Conv-Block-1	2D Conv+PReLU		$(3\times 64)\times h\times w \rightarrow 256\times h\times w$	3×3	1	1
Conv-Block-2	2D Conv+PReLU		$3\times h\times w \rightarrow 256\times h\times w$	3×3	1	1
Conv-Block-3	2D Conv		$((N+1)\times 64)\times H\times W \rightarrow 3\times H\times W$	3×3	1	1
MSD Alignment Module	2D Conv1		$(2\times 64)\times h\times w \rightarrow 64\times h\times w$	3×3	1	1
	MSRB	2D Conv2	$64\times h\times w \rightarrow 64\times h\times w$	3×3	1	1
		2D Conv3+LeakyReLU	$64\times h\times w \rightarrow 64\times h\times w$	5×5	1	2
		2D Conv4+LeakyReLU	$64\times h\times w \rightarrow 64\times h\times w$	7×7	1	3
		2D Conv5+LeakyReLU	$(3\times 64)\times h\times w \rightarrow 64\times h\times w$	3×3	1	1
	Dconv		$64\times h\times w \rightarrow 64\times h\times w$	3×3	1	1
Projection Module	MISR	Net_{res-1} (5 × ResBlock)	$256\times h\times w \rightarrow 256\times h\times w$	3×3	1	1
		2D Transpose Conv+PReLU	$256\times h\times w \rightarrow 64\times H\times W$	8×8	4	2
	SISR	DBPN [43]	$256\times h\times w \rightarrow 64\times H\times W$	-	-	-
	Net_{res-2} (5 × ResBlock)		$64\times H\times W \rightarrow 64\times H\times W$	3×3	1	1
	Net_{res-3} (5 × ResBlock)		$64\times H\times W \rightarrow 64\times H\times W$	3×3	1	1
	Downsample (2D Conv+PReLU)		$64\times H\times W \rightarrow 256\times h\times w$	8×8	4	2
Attention Module	α (2D Conv1)		$64\times H\times W \rightarrow 64\times H\times W$	3×3	1	1
	β (2D Conv2)		$64\times H\times W \rightarrow 64\times H\times W$	3×3	1	1
	$\otimes$		$2\times(64\times H\times W) \rightarrow 1\times H\times W$	-	-	-
	Sigmoid		$1\times H\times \rightarrow 1\times H\times$ W	-	-	-

B. Temporal Grouping Strategy

For the more general case, $2N + 1$ input frames will be regrouped into N groups $\{G_1, \ldots, G_N\}$, each group $G_n = \{I^{\mathrm{LR}}_{t-n}, I^{\mathrm{LR}}_t, I^{\mathrm{LR}}_{t+n}\}$ consists of a reference frame I^{LR}_t and two supporting frames I^{LR}_{t-n} and I^{LR}_{t+n}, which have the same temporal distance n from I^{LR}_t.

Supporting frames at different temporal distances contain different types of motion information. A simple case is shown in Fig. 2, where we observed that supporting frames with the same temporal distance have more similar attributes. The two supporting frames closest to the reference frame have similar motion blur, and the whole fuselage has noticeable ghosting, while the two frames farthest from the target frame have ghosting only at the wings. Obviously, when restoring the tail of the aircraft, the two supporting frames farthest from the reference frame can provide richer information. In this case, the motion pattern in supporting frames shows symmetry on both sides of the reference frame. This inspired us to group supporting frames with similar motion information.

In addition to the observed symmetry, our idea of temporal grouping is inspired by the following two evidence:

1) A similar symmetric phenomenon is also confirmed in [59]. The author claims that symmetry can be used as an intrinsic property to make the problem better

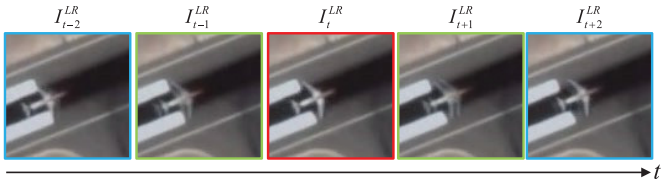


Fig. 2. Example illustrates that there is a case where the motion information is symmetrical about the reference frame (marked by red box) during motion.

constrained, thus improving the quality of SR reconstruction.

2) The idea of grouping has been proved effective in the field of HSI SR. In [23], the author first grouped spectral bands according to their similarity and then performed SR reconstruction. By grouping spectral bands into different groups, the spatial–spectral information can be explored more effectively and ultimately improve the performance of SR. Since the number of frames used in VSR is not as large as the number of bands used in HSI SR, there is no need for complex grouping strategies. We have done ablation experiments (see Table V) for all possible grouping situations and proved that grouping frames according to temporal distance is the most effective.

Generally speaking, the supporting frames closest to the reference frame have higher structural similarity (SSIM) with the reference frame. With the increase of motion time, more motion information will be introduced, and the difference between the supporting frame and the reference frame will become more prominent.

More fundamentally, each group corresponds to a subvideo sequence at a different frame rate. The group containing the frame farthest from the reference frame represents the high frame rate subvideo and the group containing the closest frame to the reference frame represents the low frame rate subvideo. Rearranging the frames in a simple way of grouping by temporal distance, putting together the frames with the same type of complementary information makes it easier for the network to learn these motion patterns from each group. Note that we have added the reference frame to each group to guide the network learn the information which is pivotal to restore the reference frame. We prove the effectiveness of our temporal grouping strategy in Section IV.

C. Feature Extraction

Take the reference frame $I_t^{\mathrm{LR}} \in R^{h\times w\times 3}$ as an example. The feature extraction module completes the initial feature extraction by a 3×3 convolution and then three residual blocks [37] further complete the deeper feature extraction. We adopt parametric rectified linear unit (PReLU) as the activation function. After feature extraction $f_{\mathrm{FE}}(\cdot)$, we obtain $F_t^{\mathrm{LR}} \in R^{h\times w\times 64}$.

Feature extraction is performed for frames in each group. Finally, the features $F_1 = \{F_{t-1}^{\mathrm{LR}}, F_t^{\mathrm{LR}}, F_{t+1}^{\mathrm{LR}}\}$ of G_1 and $F_2 = \{F_{t-2}^{\mathrm{LR}}, F_t^{\mathrm{LR}}, F_{t+2}^{\mathrm{LR}}\}$ of G_2 will be sent to the MSD alignment module for feature alignment. The parameters of feature extraction module are shared.

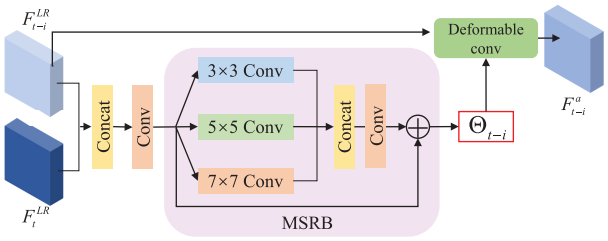


Fig. 3. Schematic of MSD alignment module. The feature F_{t-i}^{LR} of the supporting frame and the feature F_t^{LR} of the reference frame are first concatenated on the channel dimension and then fused through a convolutional layer. The sampling parameters Θ_{t-i} is learned through the proposed MSRB block.

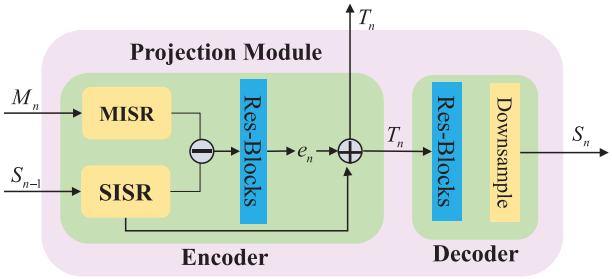


Fig. 4. Projection module used in RBPN [29]. It consists of two parts: encoder and decoder. The MISR process super-resolves multiframe feature M_n obtained from each group into an HR feature. The SISR process super-resolves the LR spatial feature of reference frame into an HR spatial feature. Here SISR is DBPN and it can be replaced by other advanced methods. The residual block in the encoder enhances the residual information. The decoder is responsible for downsampling T_n back to S_n for the next projection.

D. MSD Alignment Module

Unlike natural video, remote-sensing video scenes often contain moving objects with various scales. Therefore, the capture of motion information is complicated. Conventional alignment method based on deformable convolution such as TDA used in TDAN uses a series of single-scale convolutions to learn sampling parameters Θ_{t-i} from the concatenated reference feature F_t^{LR} and supporting feature F_{t-i}^{LR}. It is difficult to learn complex motion information containing more scale motion objects only by relying on a limited receptive field.

For this reason, we propose an MSD alignment module that utilizes *MSRB* as the generator of sampling parameters. The task of our MSD alignment module is to align the features in each group to the feature of the reference frame to complete the spatial–temporal modeling, thus the spatial–temporal information can be better exploited. Specifically, the task of MSRB is to learn sampling parameters from multiscale features, while the task of DConv is to carry out deformable convolution operation on features within each group to complete alignment.

As shown in Fig. 3, we use 3×3, 5×5, and 7×7 convolution in MSRB to extract features of different scales and then concatenate features of different scales. After a convolutional layer, we can learn the offset parameters from these multiscale features. Furthermore, through a residual skip connection, the offset learned under the initial receptive field is added to the offset learned under the multiscale structure to obtain the final sampling parameters. The process

is expressed as

$$\Theta_{t-i} = f_{\text{MSRB}}\left(\text{Conv}\left[F_t^{\text{LR}}, F_{t-i}^{\text{LR}}\right]\right) \tag{3}$$

where $[\cdot]$ represents the concatenation operation. $\Theta_{t-i} = \{\Delta p_k, \Delta m_k\}, k = 1, \ldots, K$, where $K = |p_k \in \{(-1,-1),(-1,0),\ldots,(0,1),(1,1)\}| = 9$ represents the nine sampling positions p_k in the grid of a conventional 3×3 convolution kernel. $\Delta p_k \in R^2$ represents the learned extra offset. $\Delta m_k \in R$ is the modulation coefficient. Under the guidance of the sampling parameters, we introduce the modulation DConv [60] to perform deformable convolution operations on F_{t-i}^{LR} to obtain the aligned feature F_{t-i}^a, which can be written as the following formula:

$$F_{t-i}^a = f_{\text{DConv}}\left(F_{t-i}^{\text{LR}}, \Theta_{t-i}\right). \tag{4}$$

The value of F_{t-i}^a at position p_0 is determined by

$$F_{t-i}^a(p_0) = \sum_{k=1}^{K} \omega_k \cdot F_{t-i}^{\text{LR}}(p_0 + p_k + \Delta p_k) \cdot \Delta m_k \tag{5}$$

where ω_k represents the weight of the kth sampling position. Since $(p_0 + p_k + \Delta p_k)$ may be a decimal, we use the same bilinear interpolation strategy as [32].

The features F_1 and F_2 are sent to our MSD alignment module and finally we get two sets of aligned features $F_1^a = \{F_{t-1}^a, F_t^a, F_{t+1}^a\}$, $F_2^a = \{F_{t-2}^a, F_t^a, F_{t+2}^a\} n \in [1:N]$.

After MSD alignment module, F_1^a and F_2^a are, respectively, merged through a convolutional block to obtain the LR features $M = \{M_1, \ldots, M_n\}, n \in [1, N]$, which can be expressed as follows:

$$M_n = \text{Conv}\left(\left[F_{t-n}^a, F_t^a, F_{t+n}^a\right]\right) \tag{6}$$

where $\text{Conv}(\cdot)$ represents a 3×3 convolution, which maps the concatenated feature $[F_{t-n}^a, F_t^a, F_{t+n}^a] \in R^{h \times w \times (64 \times 3)}$ to $M_n \in R^{h \times w \times 256}$, preparing for the following projection.

E. Projection Module

Our work is built upon [29]. This kind of projection idea is very suitable for remote-sensing images because of the spatial information of the LR reference frame, which is the most critical for restoring the reference frame, will participate in the projection process every time, thus continuing to play a guiding role in the entire network.

The task of projection module is to complete the spatial–temporal information fusion. Specifically, the MISR module fuses the aligned features of each group to obtain HR temporal features. The task of SISR module is to super-resolve the reference frame to obtain HR spatial features. By supplementing the temporal features to the spatial features, the fusion of spatial–temporal information is completed.

Since the vast majority of remote-sensing images are in static areas. For areas that do not contain motion information, the use of MISR methods not only increases the amount of calculation but may also introduce additional interference information due to the misalignment. In fact, an SISR method may be sufficient to achieve good results without the need for complex spatial–temporal modeling. As shown in Fig. 1, the features M_n (see the blue arrows) that encode the temporal information in each group are continuously added to the spatial information (see the red arrows) of the reference frame I_t^{LR}. One projection result T_n as well as LR spatial features S_n^{LR} will be obtained after projection. The projection process can be expressed as

$$\left(S_n^{\text{LR}}, T_n\right) = f_{\text{projection}}\left(S_{n-1}^{\text{LR}}, M_n\right). \tag{7}$$

The structure of the projection module is shown in Fig. 4, which composes of an encoder and a decoder. The task of the encoder is to fuse the spatial features of the reference frame with the temporal features of each group to get the projection output, while the decoder is to downsample the HR projection output back to the LR spatial features for the next projection. Specifically, the encoder receives M_n and the spatial features S_{n-1}^{LR} of reference frame. First, M_n is mapped to HR features through a $\text{Net}_{\text{MISR}}(\cdot)$, which consists of five residual blocks $\text{Net}_{\text{res}-1}$ and a deconvolution. At the same time, HR spatial features can be obtained by an SISR method DBPN [42]. By predicting the residual, the network will be forced to learn the temporal information that is missing in the result of an SISR. Then the residual information e_n enhanced by $\text{Net}_{\text{res}-2}$ is added to the result of an SISR and we will get the projection output T_n. In the decoding part, HR output T_n will be down-sampled back to the LR spatial feature S_n^{LR} through a $\text{Net}_{\text{Decoder}}$ for the next projection. The residual block in the decoder named $\text{Net}_{\text{res}-3}$ consists of five residual blocks and the downsampling uses an 8×8 convolution with $stride = 4$ and $padding = 2$. The previously mentioned process can be represented as

$$e_n = \text{Net}_{\text{res}-2}\left(\text{Net}_{\text{SISR}}\left(S_{n-1}^{\text{LR}}\right) - \text{Net}_{\text{MISR}}(M_n)\right) \tag{8}$$

$$T_n = e_n + \text{Net}_{\text{SISR}}\left(S_{n-1}^{\text{LR}}\right) \tag{9}$$

$$S_n^{\text{LR}} = \text{Net}_{\text{Decoder}}(T_n). \tag{10}$$

Considering the characteristics of remote-sensing images, our method is different from RBPN in the following three aspects:

1) We abandon the PyFlow alignment method which is not suitable for remote-sensing images and independent of the RBPN network. Instead, we introduce the MSD alignment module to provide more accurate alignment features for the projection. Experiments in Section IV prove that the proposed MSD alignment module is superior to the method using optical flow.
2) RBPN allows each supporting frame to participate in a projection once, so the number of projections is $2N$. However, our model adopts the temporal grouping strategy, where each group participates in projection once, so the number of projections is reduced to N, which greatly alleviates the computational complexity of the model. It can be clearly reflected in the testing time results and floating point of operations (FLOPs) comparison in Section IV.
3) In the last projection, RBPN abandons the spatial features of reference frame which are the most important for the recovery of the reference frame and simply fused the results of each projection. In order to make

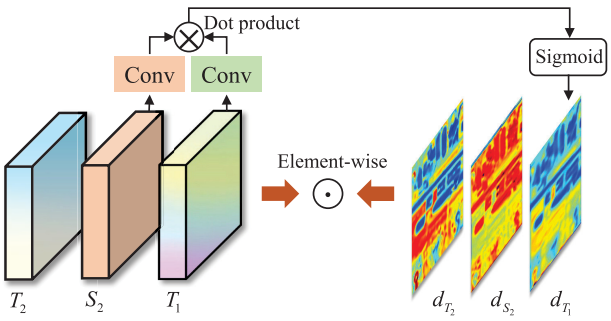


Fig. 5. Proposed TA module. The features enter the embedding space through simple 3×3 convolution filters and then the dot product is used to calculate the similarity. Finally, the attention map is restricted to $(0, 1)$ by the sigmoid function.

the spatial information of the reference frame play a guiding role in the network continuously, we repass the LR spatial features S_N^{LR} through DBPN to obtain the HR spatial features S_N^{HR} and sent it into the final fusion process together with the results of each projection. That means

$$S_N^{\mathrm{HR}} = \mathrm{Net}_{\mathrm{SISR}}\left(S_N^{\mathrm{LR}}\right). \tag{11}$$

In addition, we also introduce a TA mechanism that takes into account the different contributions of different groups to the recovery of the reference frame. Experiments in Section IV demonstrate the effectiveness of the TA mechanism.

F. TA Module

Temporal features that encode different temporal information from supporting frames in each group are not informative in the contribution of reconstructing the reference frames, so they should not be treated equally. Based on this, we adopt a TA module, in which the HR temporal features T that integrate temporal information in each group and the HR spatial features S_N^{HR} obtained from the last projection are fed into a TA module to obtain the modulated HR features $\tilde{S}_N, \tilde{T}$, that is,

$$\left[\tilde{S}_N, \tilde{T}\right] = f_{\mathrm{TA}}\left(S_N^{\mathrm{HR}}, T\right). \tag{12}$$

As shown in Fig. 5, S_2^{HR} and T_1 are converted into the embedding space through two convolutional layers α and β and then carry out dot product operation to obtain features $D_{T_1} \in R^{H \times W \times 1}$ that measures similarity between S_2^{HR} and T_1

$$D_{T_1} = \alpha\left(S_2^{\mathrm{HR}}\right) \otimes \beta(T_1). \tag{13}$$

Finally, the sigmoid function limits D_{T_1} to $(0, 1)$ to stabilize the back propagation of the gradient. So, the attention map d_{T_1} that measures the difference between spatial features S_2^{HR} and temporal features T_1 is

$$d_{T_1}(x, y) = \frac{1}{1 + e^{D_{T1}(x,y)}} \tag{14}$$

where (x, y) represents a position in d_{T_1}. We use the attention map to modulate the feature T_1, and the modulated feature $\tilde{T}_1$ is expressed as

$$\tilde{T}_1 = T_1 \odot d_{T_1} \tag{15}$$

Algorithm 1 Algorithm of Our Network

Input: $2N + 1$ LR frames:
$\mathrm{I} = \left\{I_{t-N}^{LR}, \cdots, I_t^{LR}, \cdots, I_{t+N}^{LR}\right\}$
Output: SR result I_t^{SR} of LR reference frame I_t^{LR}

Temporal Grouping: $G_1, \cdots, G_N = \left\{I_{t-N}^{LR}, I_t^{LR}, I_{t+N}^{LR}\right\}$;
Denote $n \in [1 : N]$, $i = [-n, 0, n]$;
Feature Extraction:
foreach $I_{t+i}^{LR} \in G_n$ **do**
  $F_{t+i}^{LR} = f_{FE}\left(I_{t+i}^{LR}\right)$;
end
Return $F_1, \cdots, F_N = \left\{F_{t-N}^{LR}, F_t^{LR}, F_{t+N}^{LR}\right\}$;
MSD Alignment Module:
foreach $F_{t+i}^{LR} \in F_n$ **do**
  $\Theta_{t+i} = f_{MSRB}\left(Conv\left[F_t^{LR}, F_{t+i}^{LR}\right]\right)$;
  $F_{t+i}^{a} = f_{DConv}\left(F_{t+i}^{LR}, \Theta_{t+i}\right)$;
end
Return $F_1^a, \cdots, F_N^a = \left\{F_{t-N}^a, F_t^a, F_{t+N}^a\right\}$;
foreach F_n^a **do**
  $M_n = ConvBlock1\left(\left[F_{t-n}^a, F_t^a, F_{t+n}^a\right]\right)$;
end
Return $M_1, \cdots, M_N$;
$S_0 = ConvBlock2\left(I_t^{LR}\right)$;
Projection Module:
for $n \in [1 : N]$ **do**
  $\left(S_n^{LR}, T_n\right) = f_{projection}\left(S_{n-1}^{LR}, M_n\right)$;
end
Return $T_1, \cdots, T_N, S_N^{LR}$;
$S_N^{HR} = Net_{SISR}\left(S_N^{LR}\right)$;
TA:
for $n \in [1 : N + 1]$ **do**
  $D_{T_n} = \alpha\left(S_N^{HR}\right) \otimes \beta(T_n)$;
  $\tilde{T}_n = T_n \odot d_{T_n}$;
end
$D_{S_N} = \alpha\left(S_N^{HR}\right) \otimes \beta\left(S_N^{HR}\right)$;
$\tilde{S}_N = S_N^{HR} \odot d_{S_N}$;
Return $\tilde{T} = \left\{\tilde{T}_1, \cdots, \tilde{T}_N\right\}$ and $\tilde{S}_N$;
$I_t^{SR} = ConvBlock3\left(\left[\tilde{S}_N, \tilde{T}\right]\right)$.

where $\odot$ denotes the element-wise multiplication. The same method can be used to obtain $\tilde{T}_2$ and $\tilde{S}_2$.

The modulated features are fused through a 3×3 convolution to obtain the final HR reference frame I_t^{SR}; note that this convolution layer has no activation function

$$I_t^{\mathrm{SR}} = \mathrm{Conv}\left(\left[\tilde{S}_N, \tilde{T}\right]\right). \tag{16}$$

IV. Experiment and Discussion

A. Dataset Setting

We have ten scenes of video data from Jilin-1 satellites, which have a resolution of $1m$, a frame rate of 25 frames per second, and a video duration of 20–30 s. The video name is shown in Table II. Due to the large area of reflection and cloud occlusion in the videos of San Diego-USA and Adana 01-Turkey, the image quality is poor. Therefore, these two videos do not participate in the construction of the training set but only serve as the test set.

TABLE II
JILIN-1 SATELLITE VIDEO DATA THAT WE USED TO CONSTRUCT THE DATASET

Names of the Video
20170424- Elevation Angle (17.0168) -San Francisco - United States
20170520- Elevation Angle (2.1256) - Derna - Libya
20170520- Elevation Angle (-6.6864) - Valencia - Spain
20170522- Elevation Angle (15.6646) - San Diego - United States
20170525- Elevation Angle (7.5114) - Tunis
20170525- Elevation Angle (18.0424) - Adana 01 - Turkey
20170525- Elevation Angle (18.0424) - Adana 02 - Turkey
20170602- Elevation Angle (5.0133) - Minneapolis 01 - United States
20170602- Elevation Angle (5.0133) - Minneapolis 02 - United States
20170604- Elevation Angle (4.8243) - Muhalag - Bahrain

The original frame size of the video is 4096×2160 except 3840×2160 in San Francisco-USA. We crop the video into scenes with a size of 640×640, and the overlap rate between each scene is 25%. For each video, we only take the first 100 frames. In the end, we are able to get 189 video clips as our training set. Five scenes are randomly cropped from San Diego-USA and Adana 01-Turkey, respectively, as test sets, and finally, ten test video clips (000–009) are obtained.

B. Implementation Details

We only focus on $\times 4$ SR in this article and use the *imresize* function in MATLAB to downsample the frames through bicubic interpolation to get the LR frames. The network takes five consecutive frames as input. The batch size of each epoch is 8, and in each batch, the LR patch is cropped to 32×32 from LR images with size 160×160. We also apply data augmentation by rotation and flipping. We use the way in [61] to initialize our network and choose $\mathcal{L}_1 = \|I_t^{\mathrm{SR}} - I_t^{\mathrm{HR}}\|_1$ as our loss function to measure the difference in pixel level between the predicted HR reference frame I_t^{SR} and the ground-truth I_t^{HR}. As for optimization, we use the Adam optimizer with the momentum $\beta_1 = 0.9$ and $\beta_2 = 0.999$. The initial learning rate is set to $8e-5$ and decay to $1/10$ of the previous one when the epochs reach half of the total 50. It took 35 h to train our model on a single NVIDIA RTX 2080Ti GPU. The deep learning environment is CUDA10.0 with Pytorch1.2.

C. Comparison With SOTAs

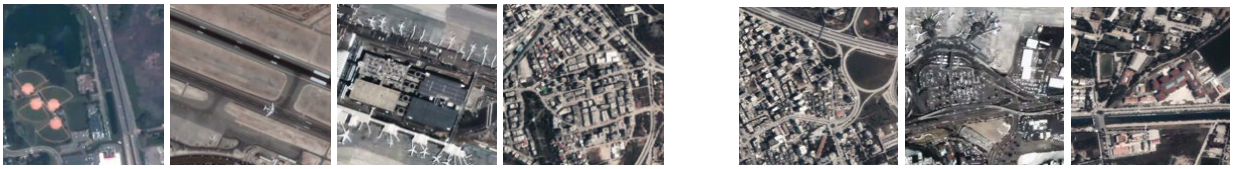
We compared the method with several SOTA VSR methods, including TDAN [32], DUF [31], RBPN [29], EDVR-L [33], and SOF-VSR [45]. The quantitative metrics used in the simulation experiment are peak signal-to-noise ratio (PSNR), SSIM [62], root mean square error (RMSE), correlation coefficient (CC), and naturalness image quality evaluator (NIQE) [63]. Note that NIQE is a no-reference indicator, which can be adopted to assess the image quality of real-world SR reconstruction without HR reference images. The rest metrics give a comprehensive comparison of different methods by requiring HR references. Since DUF does not provide training codes, we use the official pretrained model provided by the author for testing. We carefully retrain the rest of the models using our training set mentioned before, and the specific training settings are consistent with those in the corresponding official articles. We calculate the average metrics of all the video frames for each test video clip as the final result. Note that SOF-VSR cannot handle the first and last frames of a video, so only the average of 98 frames is calculated. Besides, DUF has serious edge defects, so in order to ensure the fairness of comparison, we cut off eight pixels [27], [29] on each edge of the frame when calculating PSNR and SSIM on the luminance channel (Y).

1) Quantitative Evaluation: We adopt PSNR, SSIM, RMSE, CC, and NIQE [63] as the evaluation metrics. The average PSNR/SSIM calculated on the ten test clips are shown in Table III and the other three metrics are shown in Table IV. It can be seen that our method has achieved the best performance on all test sets. Since DUF cannot be retrained, it achieved the worst results. The PSNR result of our model on the 000 test clip is 0.32 dB higher than SOF-VSR and 0.48 dB higher than EDVR. In the final average result of all test sets, our model leads second place by 0.19 dB and leads RBPN by 0.31 dB.

Both RBPN and SOF-VSR adopt optical flow to realize motion estimation. The performance of RBPN is lower than that of SOF-VSR, mainly because optical flow cannot capture the motion information that only occupies a small number of pixels in remote-sensing images, which leads to inaccurate alignment. Since SOF-VSR first super-resolves the optical flow, it can provide more accurate optical flow results, so as to achieve accurate alignment. However, our method is still superior to SOF-VSR, which fully demonstrates that the proposed MSD alignment module can provide more accurate alignment in remote-sensing images compared with the optical flow-based method.

It is noted that TDAN and EDVR also use deformable convolution for alignment, but the performance of TDAN is lower than that of RBPN, which indicates that directly applying conventional deformable convolution in remote-sensing images is not suitable either. In addition, the EDVR using the pyramid structure alignment module PCD achieves results comparable to RBPN, which shows that increasing the receptive field of the deformable convolution helps improve the alignment performance. However, the pyramid structure introduces too many parameters and also loses the edge information that is originally scarce in the remote-sensing image, which makes the performance improvement limited. With the designed MSRB, our MSD alignment module can not only learn multi-scale contextual information which is essential to preserve the texture details, but can also effectively capture small motion information in remote-sensing images. While it solves the problem of alignment difficulty in remote-sensing images, the best results are achieved.

2) Qualitative Results: In the part of qualitative results, we mainly focus on the reconstruction of objects with various scales, such as aircraft and vehicles running on the road, as well as some stationary scenes, such as buildings. Fig. 7 shows the aircraft in the scene of test clip 001. We partially enlarge the details for better observation. Also, we place



(a) (b)

Fig. 6. Dataset we built based on Jilin-1 satellite video. (a) Some samples of the training set. (b) Some samples of the test set.

TABLE III

QUANTITATIVE RESULTS ON 10 TEST VIDEO CLIPS (000–009). TAKE THE AVERAGE RESULT OF ALL FRAMES FOR EACH VIDEO. **RED** AND BLUE INDICATES THE BEST AND THE SECOND BEST PSNR/SSIM PERFORMANCE, RESPECTIVELY

Test-Clip	**Bicubic**	**DUF** [32]	**TDAN** [33]	**RBPN** [30]	**EDVR-L** [34]	**SOF-VSR** [46]	**Ours**
#Param.	-	6.8M	1.97M	12.8M	20.6M	1.1M	14.1M
000	31.05/0.9097	34.43/0.9521	35.37/0.9580	35.73/0.9595	35.88/0.9605	35.89/0.9610	**36.21/0.9633**
001	29.69/0.8795	32.27/0.9328	32.96/0.9390	33.46/0.9440	33.32/0.9441	33.53/0.9453	**33.56/0.9458**
002	31.97/0.9242	35.54/0.9583	36.38/0.9631	36.56/0.9637	36.89/**0.9657**	36.75/0..9643	**36.94**/0.9654
003	32.28/0.9251	35.33/0.9586	36.18/0.9629	36.62/0.9647	36.84/0.9660	36.86/0.9656	**37.00/0.9675**
004	30.44/0.8965	33.30/0.9396	34.07/0.9455	34.14/0.9457	34.24/0.9466	34.41/0.9475	**34.65/0.9509**
005	28.57/0.8630	30.91/0.9181	31.70/0.9252	32.10/0.9307	32.02/0.9304	32.26/0.9328	**32.45/0.9352**
006	30.62/0.9009	33.24/0.9404	34.09/0.9478	34.65/0.9534	34.39/0.9517	34.70/0.9537	**34.85/0.9554**
007	33.72/0.9391	37.14/0.9668	38.40/0.9725	39.26/0.9767	39.12/0.9760	39.15/0.9760	**39.44/0.9773**
008	32.47/0.9246	35.30/0.9558	36.38/0.9624	36.90/0.9660	36.84/0.9659	36.88/0.9660	**37.14/0.9675**
009	30.83/0.8989	33.41/0..9389	34.20/0.9450	34.61/0.9487	34.47/0.9479	34.74/0.9498	**34.84/0.9515**
Average	31.16/0.9062	34.09/0.9461	34.97/0.9521	35.40/0.9553	35.40/0.9555	35.52/0.9561	**35.71/0.9580**

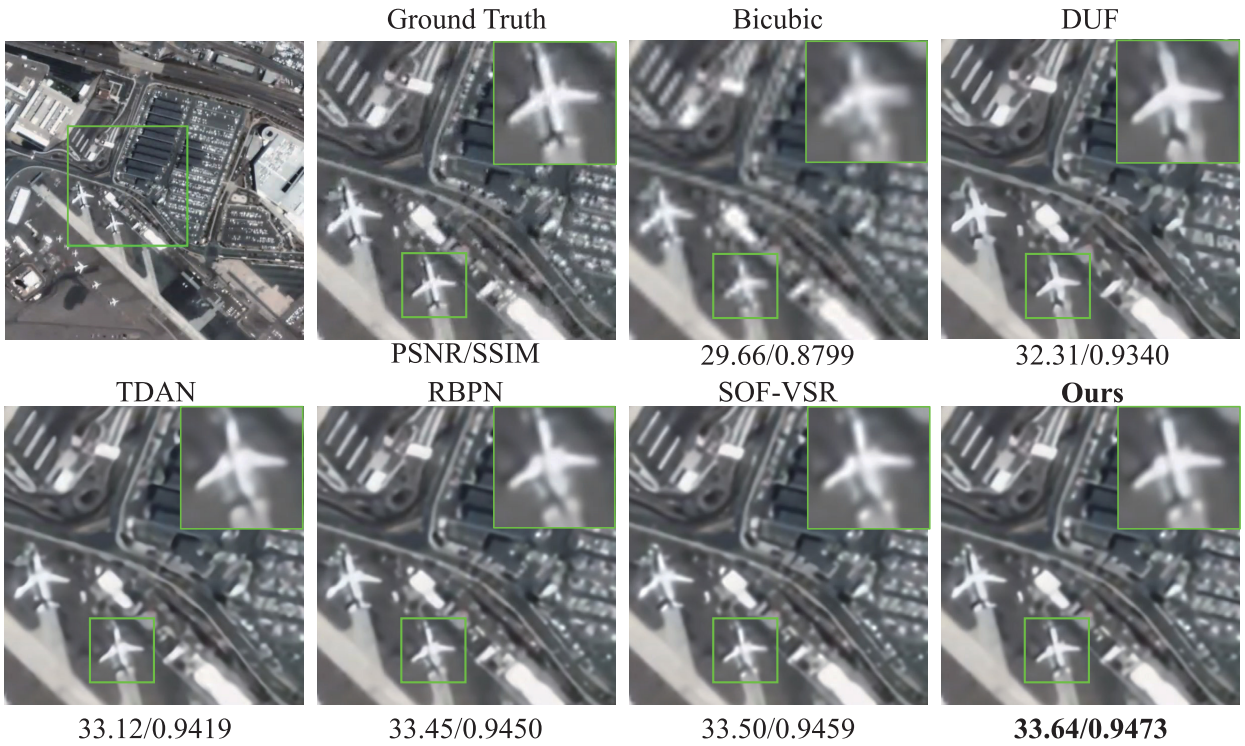


Fig. 7. Qualitative results on 001 for ×4 scaling factor. We cropped out the area marked by green box and zoomed-in view on the details for better viewing. The best performing PSNR and SSIM are shown in bold.

the corresponding method at the top of the image, and the quantitative results are displayed at the bottom. The method with the highest quantitative index is shown in bold. This is shown in Fig. 7 that all of these methods have a significant improvement over bicubic interpolation. The wings recovered by TDAN and RBPN have obvious contortions. In particular, in the fuselage near the tail of the aircraft, our method has recovered the most complete fuselage. The edge of the

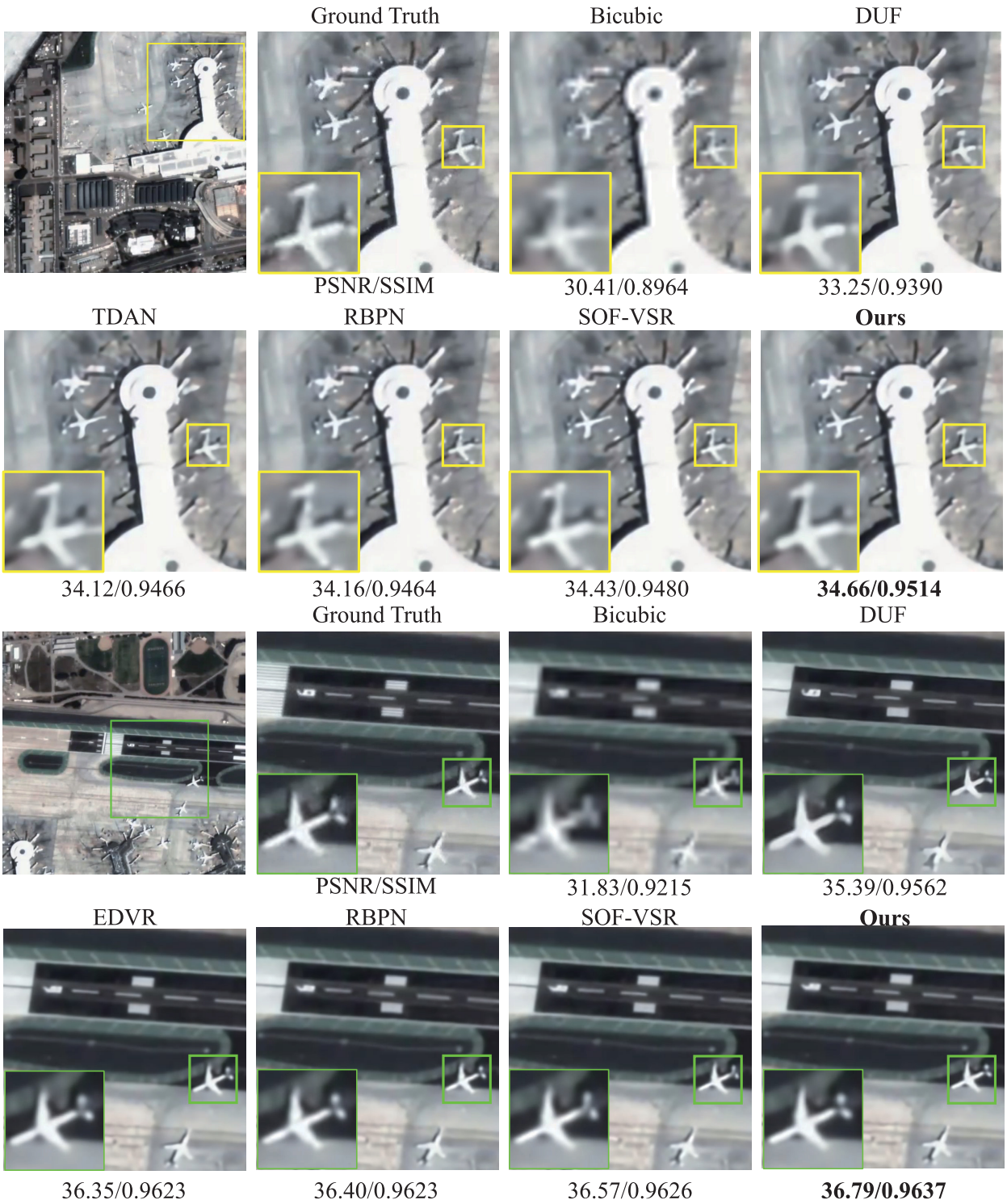


Fig. 8. Qualitative results for $\times 4$ scaling factor. The scene at the top belongs to test clip 002 and the scene below belongs to test clip 004.

landmark under the aircraft should be straight, but neither RBPN nor SOF-VSR is able to recover the edge information properly. Our method obtains the sharpest edge information, which is closest to the ground truth.

In Fig. 8, similar results can be obtained. In the yellow box marked scene, DUF does not even recover the correct shape of the aircraft. Noting the edges of the aircraft's wings, our method produces less artifacts and distortions and has more detailed information. In the green box marked scene, the wing end recovered by RBPN is bent. The result of our recovery alleviates this situation and gets a sharper texture edge. SOF-VSR failed to handle the tail of the aircraft well and mixed it with the fuselage.

In Fig. 9, we show the road scene from test clip 006 in Fig. 9(a) and zoomed-in view on a single moving object on the road. Even in scenes with dense buildings, our method can recover small-scale moving objects well. The result recovered by EDVR has a serious mix of foreground and background.

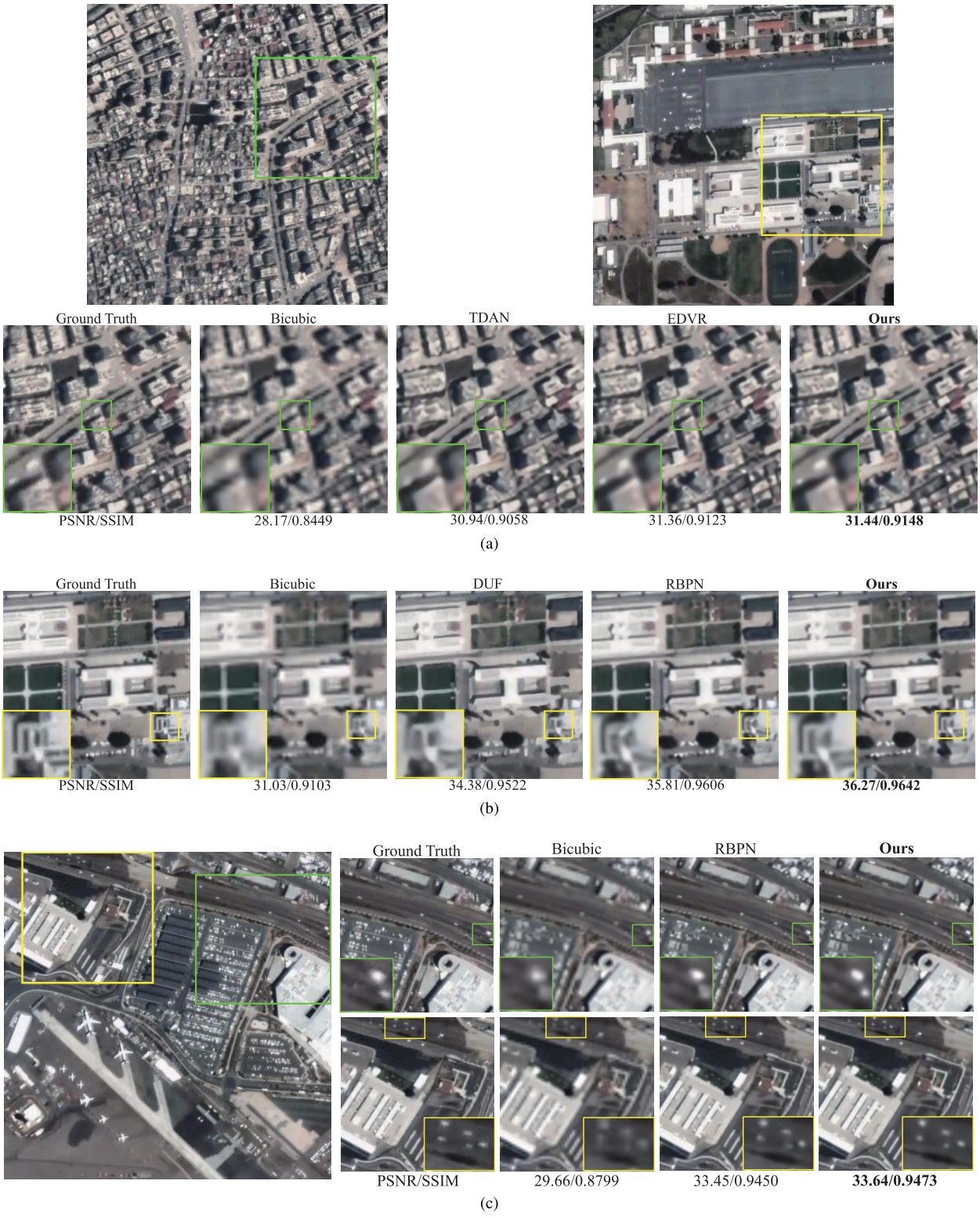


Fig. 9. Qualitative results for $\times 4$ scaling factor. (a) Zoomed-in view on a single moving object in test clip 006. (b) Zoomed-in view buildings in test clip 000. (c) Zoomed-in view multiple moving objects in test clip 001.

This demonstrates the high performance of our method when dealing with multiscale moving objects. Fig. 9(b) shows the buildings in test clip 000. The building has many strips of edge information, which cannot be recovered using interpolation methods. Due to the density of buildings in the scene, RBPN cannot correctly recover the shape of the building, resulting in distortion. In contrast, our method has yielded a clearer visual effect. In Fig. 9(c), two scenes marked by yellow and green boxes are intercepted for display. The dense vehicle area on the road is partially enlarged. It can be seen in the scene of

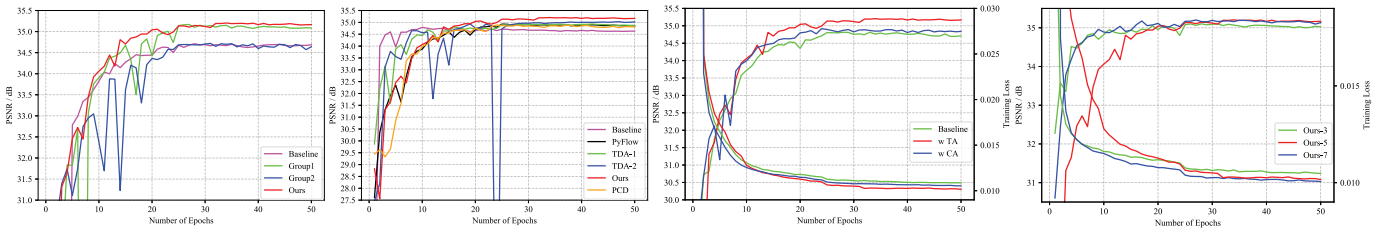


Fig. 10. Training processes for (a) models with different grouping strategy, (b) models with different alignment module, (c) models with different attention module, and (d) models with different number of input frames.

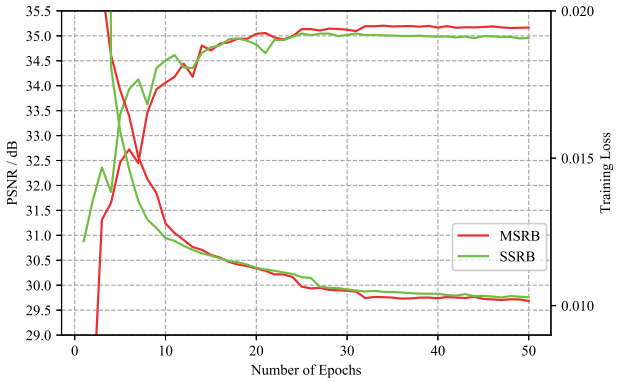


Fig. 11. Function of MSRB and SSRB.

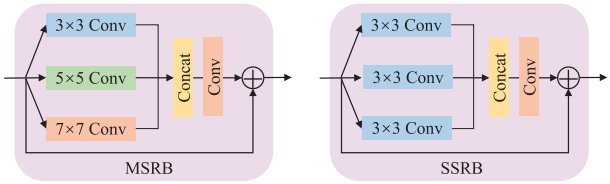


Fig. 12. Function of MSRB and SSRB.

the green box, RBPN fails to recover the small-scale vehicle, while the large-scale vehicles also have poor recovery results. The foreground of small moving objects and the background are not well distinguished. Our method recovers more precise edge information on both small- and large-scale vehicles, and the foreground and background do not mixed. In the scene of the yellow box, the situation is more challenging with a more significant density and smaller scale of moving vehicles. We can see that RBPN only recovers the four vehicles with a slightly larger scale, while the smallest car fails to recover again. Our method recovers the total number of vehicles and results in a more refined result, which further demonstrates the effectiveness of our MSD alignment module. Even for dense small moving objects, our method can cope well and achieve the best visual effect.

D. Ablation Studies

1) Temporal Grouping Strategy: To verify the effectiveness of our temporal grouping strategy, we set up different grouping strategies, leaving the rest of the network unchanged. First, a baseline is set, that is, as the conventional VSR method does not use any grouping strategy but directly aligns all the four supporting frames to the reference frame by our MSD alignment module. Since there is no grouping, it only needs to be projected once. The next three experiments adopt the idea of grouping, but the strategy of grouping was different. Group1 regroups the supporting frames before the reference frame into one group and the frames after the reference frame into another group, that is, {(1, 3, 2), (4, 3, 5)}. Group2 selects one supporting frame before and after the reference frame, but does not group them according to the temporal distance, that is, {(1, 3, 4), (2, 3, 5)}. Finally, we use our temporal grouping strategy (named T-Grouping) to rearrange the input frames, that is, {(1, 3, 5), (2, 3, 4)}. The results on test clip 002 are shown in Table V.

TABLE IV
AVERAGE RESULTS OF RMSE, CC, AND NIQE ON 10 TEST CLIPS

Algorithm	**RMSE** ↑	**CC** ↑	**NIQE** ↓
Bicubic	7.15	0.9888	19.12
TDAN	4.65	0.9951	17.60
RBPN	4.44	0.9955	17.68
EDVR-L	4.44	0.9955	17.98
SOF-VSR	4.37	0.9957	17.72
Ours	**4.28**	**0.9958**	**17.46**

TABLE V
COMPARE DIFFERENT GROUPING STRATEGIES ON TEST CLIP 002. T-GROUPING MEANS TEMPORAL GROUPING STRATEGY

Model	Baseline	{(1,3,2),(4,3,5)}	{(2,3,5),(1,3,4)}	{(1,3,5),(2,3,4)}
Grouping	✗	✓	✓	✓
T-Grouping	✗	✗	✗	✓
PSNR(dB)	34.712	35.168	34.719	**35.265**

TABLE VI
COMPARE DIFFERENT ALIGNMENT MODULES. PSNR IS CALCULATED ON TEST CLIP 002

Alignment	Baseline	PyFlow [31]	TDA-1	TDA-2 [33]	PCD [34]	MSD
Paramater	0.66 M	-	0.67 M	0.87 M	1.53M	0.69 M
PSNR(dB)	34.791	34.926	34.931	35.031	34.889	**35.265**

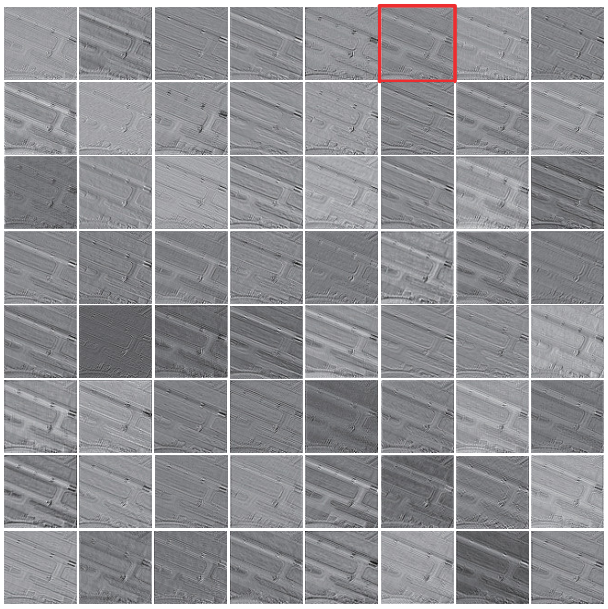


Fig. 13. Visualized features of the target frame before MSD alignment.

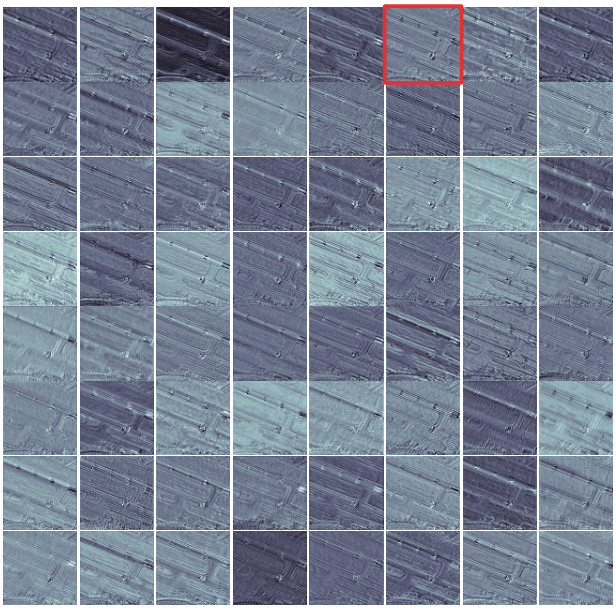


Fig. 14. Visualized features of the target frame after MSD alignment.

It can be seen that the baseline achieves the worst results without grouping. Besides, the performance of Group2 is almost as bad as not grouping. That is because the motion information contained in the supporting frames before and after the reference frame is too different, and it is difficult for the network to align them to the reference frame simultaneously. Although Group1 is not regrouped according to temporal distance, our network can still learn the complementary information provided by the supporting frames in different groups. However, using our temporal grouping strategy, the result obtained is nearly 0.1 dB higher than the second place, indicating that grouping according to temporal distance is more conducive to the network to learn valuable temporal information in different groups.

2) MSD Alignment Module:

a) Multiscale versus single-scale: The task of this experiment is to show that the multiscale structure can generate more accurate sampling parameters than the single-scale structure, thereby obtaining more accurate alignment results. As shown in Fig. 12, we additionally design a single-scale residual block (SSRB) by replacing the convolution of 5×5 and 7×7 in our MSRB with the convolution of 3×3. The results show that the performance of MSRB (35.265 dB) is better than that of SSRB (35.048 dB). The training process is shown in Fig. 11.

b) Compare with SOTA alignment modules: Next, we prove that our MSD alignment module performs better than other SOTA alignment methods. The ablation experiment for the alignment module also keeps the rest parts unchanged, and only the setting of the alignment module is changed. We also set a baseline, that is, the alignment is not performed. To make a fair comparison, we chose to replace the alignment module with two convolution blocks with a similar number of parameters. Second, we chose several SOTA alignment methods, including the optical flow-based method named PyFlow and deformable convolution-based methods named TDA and PCD. Specifically, we use PyFlow [30] to estimate

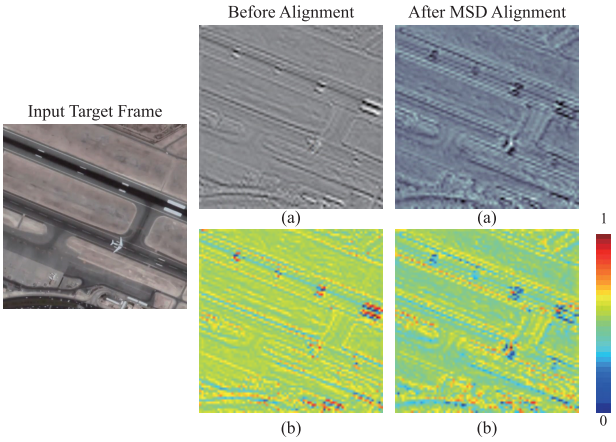


Fig. 15. Select representative feature. (a) The sixth feature map. (b) Their corresponding activation intensity map.

the optical flow between frames to complete motion estimation and then use the optical flow to warp supporting frames to realize motion compensation. TDA is the alignment module used in TDAN. It uses four 3×3 convolution layers as the generator of sampling parameters. After each convolution layer, it uses DConv to carry out a rough alignment. To be fair in comparison, we remove the last convolution layer of TDA, which is denoted as TDA-1, making it close to the number of parameters of our MSD alignment module. Note that the complete TDA module, denoted as TDA-2, is also involved in the experiment. PCD (Pyramid, Cascading and Deformable Convolution) is an alignment module used in EDVR that introduces a pyramid structure to increase the receptive field. The experimental results are shown in Table VI.

It can be seen that if the alignment between frames is not carried out, the network cannot learn enough temporal information from the supporting frames, resulting in poor performance. PyFlow has nearly the same performance as TDA-1, which illustrates that the proper alignment of supporting frames and reference frames helps the network to

TABLE VII

COMPARE MODELS WITH DIFFERENT ATTENTION MODULES. PSNR IS CALCULATED ON TEST CLIP 002

Model	Baseline	Model-1	Ours
Channel Attention	✗	✓	✗
Temporal Attention	✗	✗	✓
PSNR(dB)	34.815	34.918	**35.265**

TABLE VIII

COMPARE THE NUMBER OF INPUT FRAMES USED IN OUR MODEL. PSNR/SSIM ARE CALCULATED ON ALL TEN TEST CLIPS

$N_{frame} = 2N + 1$	PSNR(dB)/SSIM
3	35.58 / 0.9567
5	**35.71 / 0.9580**
7	35.68 / 0.9576

TABLE IX

FLOPS ARE CALCULATED ON AN LR IMAGE OF SIZE 64×64. THE TEST TIME IS THE TOTAL TIME TAKEN TO COMPLETE THE TESTS FOR TEN TEST CLIPS DIVIDED BY THE TOTAL NUMBER OF FRAMES. PSNR IS CALCULATED ON ALL TEN TEST CLIPS

Model	FLOPs(T)	Test time(s / per frame)	PSNR(dB)
DUF	0.12	0.167	34.09
TDAN	-	0.163	34.97
RBPN	1.77	0.214	35.4
SOF-VSR	0.06	0.062	35.52
EDVR-L	0.18	0.071	35.54
Ours-3	0.45	0.067	35.58
Ours-5	0.77	0.113	**35.71**
Ours-7	1.07	0.165	35.68

learn complementary information. Note that after removing a convolutional layer, TDA-1 has a 0.1 dB drop compared to TDA-2. The PCD module only achieves better performance than the baseline, because the use of strided convolution may lose the texture information of the objects [34]. Our MSD alignment module still achieves 0.23 dB higher than TDA-2 and 0.376 dB higher than PCD while the number of parameters is less. With the same number of parameters, our model is 0.33 dB higher than TDA-1, which further proves that our MSD alignment module can achieve more accurate alignment on remote sensing images, providing richer information for subsequent fusion.

In addition, we visualize the feature maps of the reference frame before and after the MSD alignment module in Figs. 13 and 14. Select the representative 6th feature map as shown in Fig. 15, the aligned features are clearer and cleaner in detail, which demonstrates the validity of our MSD alignment module.

3) TA Module: The predicted HR reference frame should have a high degree of consistency in the spatial structure with the LR reference frame. Therefore, the spatial features of the reference frame should play a leading role in the network. Temporal features learned from different groups have different contributions to reconstruct the reference frame. Based on the above two points, we adopt a TA module to enable the network to adaptively learn the most favorable information to recover the reference frame. Suppose our TA module is removed, and

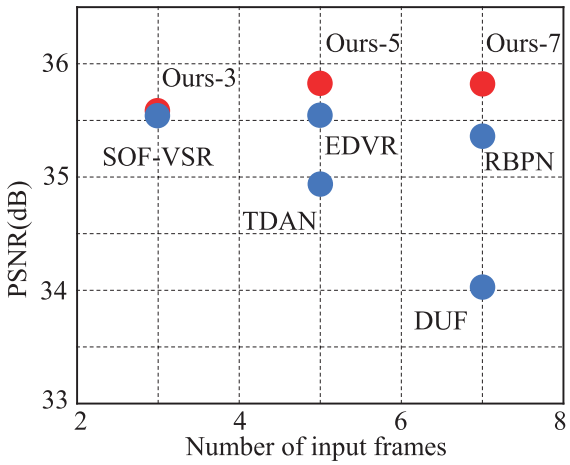


Fig. 16. Compare the number of input frames used by different methods. The PSNR was calculated on all ten test clips.

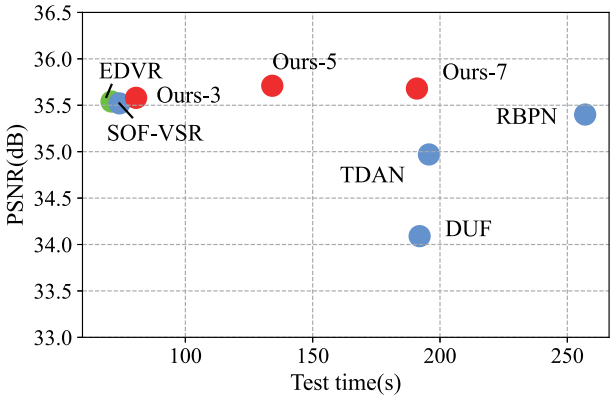


Fig. 17. Performance versus testing time on all ten test clips. PSNR was calculated on all ten test clips. The test time here is the total time.

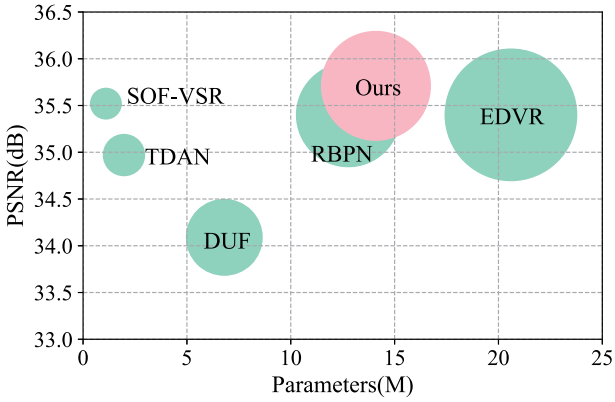


Fig. 18. Comparison of the number of parameters in different methods.

$S_N^{\mathrm{HR}}, T_1, T_2$ are simply fused through a convolution layer like RBPN. In that case, we can see that the performance is reduced by 0.45 dB compared with the addition of the TA module. Besides, we add a widely used Squeeze-and-Excitation building block (channel attention) as comparison. The result shows that our TA exceeds the channel attention by 0.347 dB. Through our TA module, the network can eliminate the unimportant information in the temporal features and improve the quality of SR.

4) Input Frame Number: We explored the performance of the model in the case of three, five, and seven frames as input,

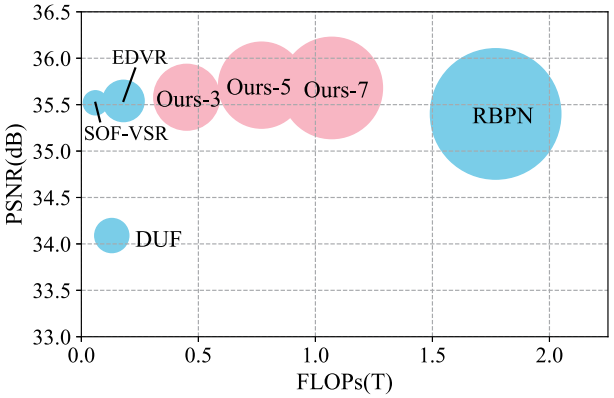


Fig. 19. FLOPs of different models, calculated on an LR image of size 64 × 64. PSNR was calculated on all ten test clips.

which are denoted as Ours-3,Ours-5, and Ours-7, respectively. The average quantitative results on all test sets are shown in Table VIII. Experiments show that Ours-5 achieves the best results. Actually, as shown in Fig. 16, Ours-3 has already achieved performance that surpasses all the SOTA methods and used fewer frames than the seven frames of RBPN and the seven frames of DUF, TDAN, and EDVR. Fewer frames are used to achieve better results, demonstrating the high efficiency of our network.

E. Model Efficiency Analysis

We compare the parameters of all methods in Fig. 18. The test time on all test sets is also taken into account. In addition, FLOPs [64] is introduced as a more intuitive manifestation of model complexity. The quantitative results of test time and FLOPs are shown in Table IX. The visualization results are shown in Figs. 17 and 19, respectively.

As shown in Fig. 17, EDVR takes the shortest time and RBPN takes the longest time. The test time of our model gradually increases with the increase of the number of input frames. Ours-3 achieves superior results when its efficiency is comparable to that of SOF-VSR. Ours-7 has the same efficiency as TDAN and DUF, but it achieves higher performance. Our final model Ours-5 has made a good tradeoff between performance and running speed. In Fig. 18, compared to EDVR, our model reduces the amount of parameters by 31.5%, but the performance is improved by 0.31 dB. The same conclusion can be obtained from Fig. 19. Compared to RBPN, Ours-3 improves performance by 0.14 dB with a 75% reduction in complexity and achieves the best performance of all methods. A good balance is reached between model complexity and performance.

F. Further Discussion

1) Generative Adversarial Training for Our Method: The task of training our network with GAN is to show that our method can work in other training strategies. In this part, we use adversarial training strategies to train our methods, as shown in Fig. 20. Specifically, we make the network framework shown in Fig. 1 as a generator. The generator receives a sequence of LR frames and generates a fake image SR.

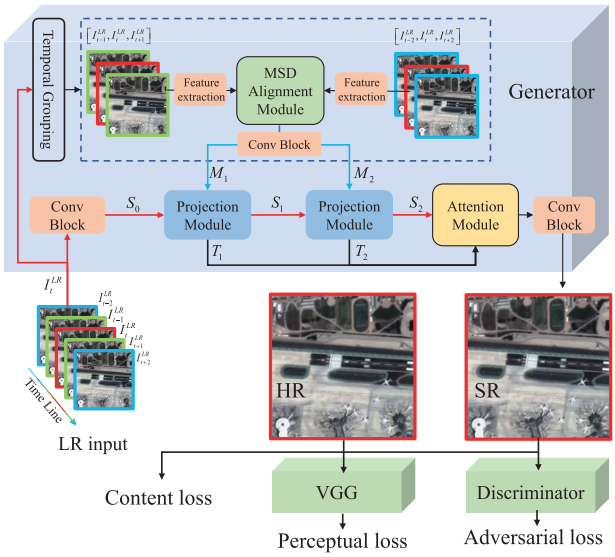


Fig. 20. Adversarial training framework for our method.

TABLE X

QUANTITATIVE RESULTS ON 10 TEST CLIPS. OURS-GAN STANDS FOR TRAINING OUR MODEL WITH GAN FRAMEWORK

Test Clips	Bicubic	Ours-GAN	Ours
000	31.05/0.9097	34.50/0.9448	**36.21/0.9633**
001	29.69/0.8795	32.44/0.9265	**33.56/0.9458**
002	31.97/0.9242	35.14/0.9481	**36.94/0.9654**
003	32.28/0.9251	35.32/0.9493	**37.00/0.9675**
004	30.44/0.8965	33.22/0.9297	**34.65/0.9509**
005	28.57/0.8630	31.22/0.9117	**32.45/0.9352**
006	30.62/0.9009	33.43/0.9380	**34.85/0.9554**
007	33.72/0.9391	37.45/0.9632	**39.44/0.9773**
008	32.47/0.9246	35.51/0.9518	**37.14/0.9675**
009	30.83/0.8989	33.57/0.9335	**34.84/0.9515**
Average	31.16/0.9062	34.18/0.9397	**35.71/0.9580**

Then we use the discriminator in ESRGAN [65] to distinguish between ground-truth images and fake images. Following the settings in ESRGAN, our discriminator can be expressed as

$$D_{\mathrm{Ra}}\left(I_t^{\mathrm{HR}}, I_t^{\mathrm{SR}}\right) = \sigma\left(C\left(I_t^{\mathrm{HR}}\right) - \frac{1}{N}\sum_{i=1}^{N} C\left(I_t^{\mathrm{HR}}\right)\right) \quad (17)$$

where I_t^{HR} represents the ground-truth HR image, and I_t^{SR} represents the fake HR image obtained by the generator. $C(\cdot)$ is the standard discriminator used in SRGAN [6], $\sigma(\cdot)$ represents the sigmoid activation function, and N is mini-batch.

The adversarial loss of the discriminator $\mathcal{L}_D^{\mathrm{Ra}}$ and generator $\mathcal{L}_G^{\mathrm{Ra}}$ can be expressed as

$$\mathcal{L}_D^{\mathrm{Ra}} = -\frac{1}{N}\sum_{i=1}^{N}\left[\log\left(D_{\mathrm{Ra}}\left(I_t^{\mathrm{HR}}, \mathrm{I}\right)\right) + \log\left(1 - D_{\mathrm{Ra}}\left(I_t^{\mathrm{SR}}, I_t^{\mathrm{HR}}\right)\right)\right] \quad (18)$$

$$\mathcal{L}_G^{\mathrm{Ra}} = -\frac{1}{N}\sum_{i=1}^{N}\left[\log\left(1 - D_{\mathrm{Ra}}\left(I_t^{\mathrm{HR}}, \mathrm{I}\right)\right) + \log\left(D_{\mathrm{Ra}}\left(I_t^{\mathrm{SR}}, I_t^{\mathrm{HR}}\right)\right)\right] \quad (19)$$

where I represents the LR frames input, $I_t^{\mathrm{SR}} = G(\mathrm{I})$, and $G(\cdot)$ represents the generator.

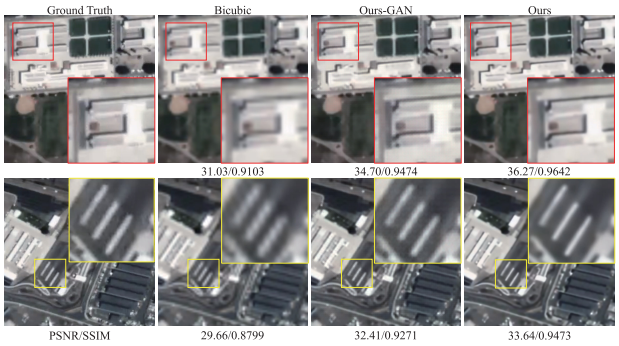


Fig. 21. Qualitative results. Ours-GAN can produce more realistic results.

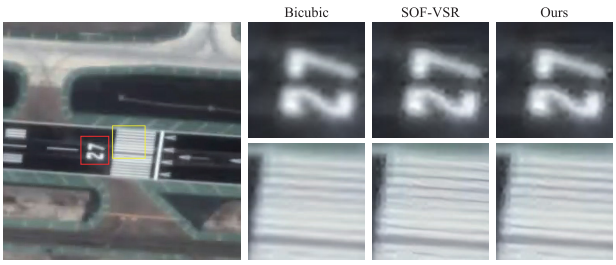


Fig. 22. Qualitative results of real-world experiments.

Then, the total loss of the generator is

$$\mathcal{L}_G = \mathcal{L}_{\text{percep}} + \lambda \mathcal{L}_G^{\text{Ra}} + \eta \mathcal{L}_1. \tag{20}$$

Among them, $\mathcal{L}_{\text{percep}}$ is the perceptual loss, which can be expressed as

$$\mathcal{L}_{\text{percep}} = \frac{1}{N} \sum_{i=1}^{N} \left\| \phi\left(I_t^{\text{SR}}\right) - \phi\left(I_t^{\text{HR}}\right) \right\|_1 \tag{21}$$

where $\phi(\cdot)$ is the 4th convolution before the 5th maxpooling layer in pretrained VGG19 [66], and $\mathcal{L}_1 = \|I_t^{\text{SR}} - I_t^{\text{HR}}\|_1$ is the content loss. We followed the settings in [65] and set $\lambda = 5 \times 10^{-3}$ and $\eta = 1 \times 10^{-2}$.

The results after training with GAN are shown in Table X. The average PSNR and SSIM have dropped by -1.53 dB and -0.0183, respectively. The qualitative results are shown in Fig. 21. Although the perception-driven training strategy loses a certain amount of PSNR/SSIM, the generated image is not as over-smooth as the PSNR-driven result and is closer to the ground truth.

2) Real-World Experiment: The task of adding experiments in real world is to show that our method has good generalization ability. In real-world experiments, we do not downsample the video and super-resolve it directly. However, the degradation in real-world is more challenging. The model trained by using bicubic to simulate the degradation cannot achieve as good results as in the simulation experiment. The qualitative results are shown in Fig. 22. Pay attention to the markings on the ground, our method can produce clearer details.

V. Conclusion

This article proposes a deep learning network for satellite VSR using MSD convolution alignment and temporal grouping projection. First, a simple but effective temporal grouping strategy regroups the continuously input frames into different groups according to the temporal distance from the reference frame. Subsequently, we use our proposed MSD convolution alignment module to align the frames in each group to the reference frame to realize implicit motion estimation and motion compensation. Finally, in order to ensure that the HR results maintain a high degree of consistency with the LR reference frame in terms of spatial features, we adopt a TA module. The projection results of each group and the spatial features of the reference frame are sent into the TA module to adaptively learn the most valuable information for restoring the reference frame. Our experiment on Jilin-1 proves that our method is superior to SOTA methods and achieves the best tradeoff between performance and model efficiency.

In the future work, more works should be done to simplify the network while ensuring high performance. The projection module brings out a large number of parameters. Although the FLOPs of our model are significantly reduced and the running speed has been greatly improved, the number of model parameters is still large. In addition, the performance in real-world experiment is severely degraded, and we will try our best to explore the SR problem in real-world.

References

[1] J. Shao, B. Du, C. Wu, M. Gong, and T. Liu, "HRSiam: High-resolution siamese network, towards space-borne satellite video tracking," *IEEE Trans. Image Process.*, vol. 30, pp. 3056–3068, Feb. 2021.

[2] J. Wang, Y. Zhong, Z. Zheng, A. Ma, and L. Zhang, "RSNet: The search for remote sensing deep neural networks in recognition tasks," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 3, pp. 2520–2534, Mar. 2021.

[3] L. Yue, H. Shen, J. Li, Q. Yuanc, H. Zhang, and L. Zhang, "Image super-resolution: The techniques, applications, and future," *Signal Process.*, vol. 128, pp. 389–408, Nov. 2016.

[4] P. Yi, Z. Wang, K. Jiang, J. Jiang, T. Lu, and J. Ma, "A progressive fusion generative adversarial network for realistic and consistent video super-resolution," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, Dec. 3, 2020, doi: 10.1109/TPAMI.2020.3042298.

[5] Z.-S. Liu, L.-W. Wang, C.-T. Li, W.-C. Siu, and Y.-L. Chan, "Image super-resolution via attention based back projection networks," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshop (ICCVW)*, Oct. 2019, pp. 3517–3525.

[6] C. Ledig *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 105–114.

[7] T. Dai, J. Cai, Y. Zhang, S.-T. Xia, and L. Zhang, "Second-order attention network for single image super-resolution," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 11057–11066.

[8] S. Zhang, Q. Yuan, J. Li, J. Sun, and X. Zhang, "Scene-adaptive remote sensing image super-resolution using a multiscale attention network," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 7, pp. 4764–4779, Jul. 2020.

[9] X. Hu, H. Mu, X. Zhang, Z. Wang, T. Tan, and J. Sun, "Meta-SR: A magnification-arbitrary network for super-resolution," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 1575–1584.

[10] W. Sun and Z. Chen, "Learned image downscaling for upscaling using content adaptive resampler," *IEEE Trans. Image Process.*, vol. 29, pp. 4027–4040, Feb. 2020, doi: 10.1109/TIP.2020.2970248.

[11] M. Deudon *et al.*, "HighRes-Net: Recursive fusion for multi-frame super-resolution of satellite imagery," 2020, *arXiv:2002.06460*. [Online]. Available: http://arxiv.org/abs/2002.06460

[12] E. Faramarzi, D. Rajan, and M. P. Christensen, "Unified blind method for multi-image super-resolution and single/multi-image blur deconvolution," *IEEE Trans. Image Process.*, vol. 22, no. 6, pp. 2101–2114, Jun. 2013.

[13] Z. Wang *et al.*, "Multi-memory convolutional neural network for video super-resolution," *IEEE Trans. Image Process.*, vol. 28, no. 5, pp. 2530–2544, May 2018.

[14] T. Isobe *et al.*, "Video super-resolution with temporal group attention," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 8008–8017.

[15] M. S. M. Sajjadi, R. Vemulapalli, and M. Brown, "Frame-recurrent video super-resolution," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6626–6634.

[16] W. Shi *et al.*, "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1874–1883.

[17] S. D. Babacan, R. Molina, and A. K. Katsaggelos, "Variational Bayesian super resolution," *IEEE Trans. Image Process.*, vol. 20, no. 4, pp. 984–999, Apr. 2011.

[18] R. Zeyde, M. Elad, and M. Protter, "On single image scale-up using sparse-representations," in *Proc. Int. Conf. Curves Surf.* Berlin, Germany: Springer, 2010, pp. 711–730.

[19] M. Bevilacqua, A. Roumy, C. Guillemot, and M.-L.-A. Morel, "Low-complexity single-image super-resolution based on nonnegative neighbor embedding," in *Proc. Brit. Mach. Vis. Conf.*, 2012.

[20] Q. Yuan *et al.*, "Deep learning in environmental remote sensing: Achievements and challenges," *Remote Sens. Environ.*, vol. 241, pp. 1–24, 2020.

[21] Q. Zhang, Q. Yuan, J. Li, F. Sun, and L. Zhang, "Deep spatio-spectral Bayesian posterior for hyperspectral image non-i.i.d. noise removal," *ISPRS J. Photogramm. Remote Sens.*, vol. 164, pp. 125–137, Jun. 2020.

[22] Q. Zhang, Q. Yuan, J. Li, X. Liu, H. Shen, and L. Zhang, "Hybrid noise removal in hyperspectral imagery with a spatial–spectral gradient network," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 10, pp. 7317–7329, Oct. 2019.

[23] D. Liu, J. Li, and Q. Yuan, "A spectral grouping and attention-driven residual dense network for hyperspectral image super-resolution," *IEEE Trans. Geosci. Remote Sens.*, early access, Jan. 20, 2021, doi: 10.1109/TGRS.2021.3049875.

[24] Y. Xu, Z. Wu, J. Chanussot, and Z. Wei, "Hyperspectral images super-resolution via learning high-order coupled tensor ring representation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 11, pp. 4747–4760, Nov. 2020.

[25] Z. Wu, W. Zhu, J. Chanussot, Y. Xu, and S. Osher, "Hyperspectral anomaly detection via global and local joint modeling of background," *IEEE Trans. Signal Process.*, vol. 67, no. 14, pp. 3858–3869, Jul. 2019.

[26] Z. Wu *et al.*, "Scheduling-guided automatic processing of massive hyperspectral image classification on cloud computing architectures," *IEEE Trans. Cybern.*, vol. 51, no. 7, pp. 3588–3601, Jul. 2021.

[27] X. Tao, H. Gao, R. Liao, J. Wang, and J. Jia, "Detail-revealing deep video super-resolution," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 4472–4480.

[28] J. Caballero *et al.*, "Real-time video super-resolution with spatio-temporal networks and motion compensation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4778–4787.

[29] M. Haris, G. Shakhnarovich, and N. Ukita, "Recurrent back-projection network for video super-resolution," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3897–3906.

[30] T. Xue, B. Chen, J. Wu, D. Wei, and W. T. Freeman, "Video enhancement with task-oriented flow," *Int. J. Comput. Vis.*, vol. 127, no. 8, pp. 1106–1125, 2019.

[31] Y. Jo, S. W. Oh, J. Kang, and S. J. Kim, "Deep video super-resolution network using dynamic upsampling filters without explicit motion compensation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 3224–3232.

[32] Y. Tian, Y. Zhang, Y. Fu, and C. Xu, "TDAN: Temporally-deformable alignment network for video super-resolution," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 3360–3369.

[33] X. Wang, K. C. K. Chan, K. Yu, C. Dong, and C. C. Loy, "EDVR: Video restoration with enhanced deformable convolutional networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2019, pp. 1–10.

[34] H. Song, W. Xu, D. Liu, B. Liu, Q. Liu, and D. N. Metaxas, "Multi-stage feature fusion network for video super-resolution," *IEEE Trans. Image Process.*, vol. 30, pp. 2923–2934, Feb. 2021, doi: 10.1109/TIP.2021.3056868.

[35] C. Dong, C. C. Loy, K. He, and X. Tang, "Image super-resolution using deep convolutional networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 2, pp. 295–307, Feb. 2015.

[36] J. Kim, J. K. Lee, and K. M. Lee, "Accurate image super-resolution using very deep convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1646–1654.

[37] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[38] B. Lim, S. Son, H. Kim, S. Nah, and K. M. Lee, "Enhanced deep residual networks for single image super-resolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jul. 2017, pp. 136–144.

[39] J. Yu *et al.*, "Wide activation for efficient and accurate image super-resolution," 2018, *arXiv:1808.08718*. [Online]. Available: http://arxiv.org/abs/1808.08718

[40] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, and Y. Fu, "Residual dense network for image super-resolution," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 2472–2481.

[41] K. Jiang, Z. Wang, P. Yi, and J. Jiang, "Hierarchical dense recursive network for image super-resolution," *Pattern Recognit.*, vol. 107, Nov. 2020, Art. no. 107475.

[42] M. Haris, G. Shakhnarovich, and N. Ukita, "Deep back-projection networks for super-resolution," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1664–1673.

[43] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu, "Image super-resolution using very deep residual channel attention networks," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 286–301.

[44] W. Yang, X. Zhang, Y. Tian, W. Wang, and J. Xue, "Deep learning for single image super-resolution: A brief review," *IEEE Trans. Multimedia*, vol. 21, no. 12, pp. 3106–3121, Dec. 2019.

[45] L. Wang, Y. Guo, L. Liu, Z. Lin, X. Deng, and W. An, "Deep video super-resolution using hr optical flow estimation," *IEEE Trans. Image Process.*, vol. 29, pp. 4323–4336, 2020, doi: 10.1109/TIP.2020.2967596.

[46] J. Dai *et al.*, "Deformable convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 764–773.

[47] J. Lin, Y. Huang, and L. Wang, "FDAN: Flow-guided deformable alignment network for video super-resolution," 2021, *arXiv:2105.05640*. [Online]. Available: http://arxiv.org/abs/2105.05640

[48] X. Zhu, Z. Li, J. Lou, and Q. Shen, "Video super-resolution based on a spatio-temporal matching network," *Pattern Recognit.*, vol. 110, Feb. 2021, Art. no. 107619.

[49] K. C. Chan, X. Wang, K. Yu, C. Dong, and C. C. Loy, "BasicVSR: The search for essential components in video super-resolution and beyond," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2021, pp. 4947–4956.

[50] K. C. K. Chan, S. Zhou, X. Xu, and C. C. Loy, "BasicVSR++: Improving video super-resolution with enhanced propagation and alignment," 2021, *arXiv:2104.13371*. [Online]. Available: http://arxiv.org/abs/2104.13371

[51] P. Yi *et al.*, "Omniscient video super-resolution," 2021, *arXiv:2103.15683*. [Online]. Available: http://arxiv.org/abs/2103.15683

[52] Y. Luo, L. Zhou, S. Wang, and Z. Wang, "Video satellite imagery super resolution via convolutional neural networks," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 12, pp. 2398–2402, Dec. 2017.

[53] A. Xiao, Z. Wang, L. Wang, and Y. Ren, "Super-resolution for 'Jilin-1' satellite video imagery via a convolutional network," *Sensors*, vol. 18, no. 4, p. 1194, Apr. 2018.

[54] K. Jiang, Z. Wang, P. Yi, and J. Jiang, "A progressively enhanced network for video satellite imagery superresolution," *IEEE Signal Process. Lett.*, vol. 25, no. 11, pp. 1630–1634, Nov. 2018.

[55] K. Jiang, Z. Wang, P. Yi, G. Wang, T. Lu, and J. Jiang, "Edge-enhanced GAN for remote sensing image superresolution," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 8, pp. 5799–5812, Aug. 2019.

[56] S. Lei and Z. Shi, "Hybrid-scale self-similarity exploitation for remote sensing image super-resolution," *IEEE Trans. Geosci. Remote Sens.*, early access, Apr. 12, 2021, doi: 10.1109/TGRS.2021.3069889.

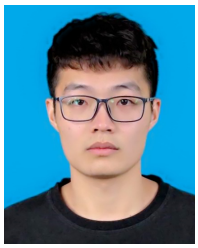
[57] H. Liu, Y. Gu, T. Wang, and S. Li, "Satellite video super-resolution based on adaptively spatiotemporal neighbors and nonlocal similarity regularization," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 12, pp. 8372–8383, Dec. 2020.

[58] Z. He and D. He, "A unified network for arbitrary scale super-resolution of video satellite images," *IEEE Trans. Geosci. Remote Sens.*, early access, Dec. 2, 2020, doi: 10.1109/TGRS.2020.3038653.

[59] D.-L. Chen, L. Zhang, and H. Huang, "Robust extraction and super-resolution of low-resolution flying airplane from satellite video," *IEEE Trans. Geosci. Remote Sens.*, early access, Mar. 18, 2021, doi: 10.1109/TGRS.2021.3064064.

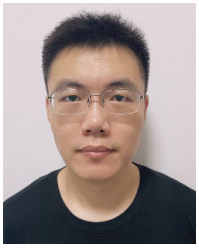
[60] X. Zhu, H. Hu, S. Lin, and J. Dai, "Deformable ConvNets v2: More deformable, better results," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 9308–9316.

[61] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1026–1034.
[62] A. Hore and D. Ziou, "Image quality metrics: PSNR vs. SSIM," in *Proc. 20th Int. Conf. Pattern Recognit.*, Aug. 2010, pp. 2366–2369.
[63] A. Mittal, R. Soundararajan, and A. C. Bovik, "Making a 'completely blind' image quality analyzer," *IEEE Signal Process. Lett.*, vol. 20, no. 3, pp. 209–212, Mar. 2013.
[64] P. Molchanov, S. Tyree, T. Karras, T. Aila, and J. Kautz, "Pruning convolutional neural networks for resource efficient inference," 2016, *arXiv:1611.06440*. [Online]. Available: http://arxiv.org/abs/1611.06440
[65] X. Wang *et al.*, "ESRGAN: Enhanced super-resolution generative adversarial networks," in *Proc. Eur. Conf. Comput. Vis. (ECCV) Workshops*, 2018, pp. 63–79.
[66] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*. [Online]. Available: http://arxiv.org/abs/1409.1556



Yi Xiao received the B.S. degree from the School of Mathematics and Physics, China University of Geosciences, Wuhan, China, in 2020. He is pursuing the M.S. degree with the School of Geodesy and Geomatics, Wuhan University, Wuhan.

His major research interests are remote-sensing image super-resolution and computer vision.



Xin Su received the B.S. degree in electronic engineering from Wuhan University, Wuhan, China, in 2008, and the Ph.D. degree in image and signal processing from Télécom ParisTech, Paris, France, in 2015.

He was a Post-Doctoral Researcher with the Team SIROCCO, Institut National de Recherche en Informatique et en Automatique, Rennes, France. He is an Assistant Professor with the School of Remote Sensing and Information Engineering, Wuhan University. His research interests include multitemporal remote sensing image processing, multiview image processing, and 3-D video communication.

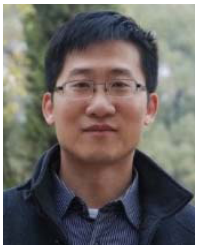
Qiangqiang Yuan (Member, IEEE) received the B.S. degree in surveying and mapping engineering and the Ph.D. degree in photogrammetry and remote sensing from Wuhan University, Wuhan, China, in 2006 and 2012, respectively.

In 2012, he joined the School of Geodesy and Geomatics, Wuhan University, where he is a Professor. He has published more than 90 research articles, including more than 70 peer-reviewed articles in international journals, such as *Remote Sensing of Environment, ISPRS Journal of Photogrammetry and Remote Sensing*, IEEE TRANSACTION ON IMAGE PROCESSING, and IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING. His research interests include image reconstruction, remote sensing image processing and application, and data fusion.

Dr. Yuan was a recipient of the Youth Talent Support Program of China in 2019, the Top-Ten Academic Star of Wuhan University in 2011, and the recognition of Best Reviewers of the IEEE GEOSCIENCE AND REMOTE SENSING LETTERS in 2019. In 2014, he received the Hong Kong Scholar Award from the Society of Hong Kong Scholars and the China National Postdoctoral Council. He is an Associate Editor of five international journals and has frequently served as a Referee for more than 40 international journals for remote sensing and image processing.

Denghong Liu received the B.S. degree in geodesy and geomatics engineering from Wuhan University, Wuhan, China, in 2019, where he is pursuing the M.S. degree with the School of Geodesy and Geomatics.

His research interests include hyperspectral image processing, deep learning, and computer vision.



Huanfeng Shen (Senior Member, IEEE) received the B.S. degree in surveying and mapping engineering and the Ph.D. degree in photogrammetry and remote sensing from Wuhan University, Wuhan, China, in 2002 and 2007, respectively.

In 2007, he joined the School of Resource and Environmental Sciences (SRES), Wuhan University, where he is a Luojia Distinguished Professor and the Associate Dean. He was the PI of two projects supported by the National Key Research and Development Program of China and six projects supported by the National Natural Science Foundation of China. He has authored over 100 research articles in peer-reviewed international journals. His research interests include remote sensing image processing, multisource data fusion, and intelligent environmental sensing.

Dr. Shen is a Council Member of the China Association of Remote Sensing Application, an Education Committee Member of the Chinese Society for Geodesy Photogrammetry and Cartography, and a Theory Committee Member of the Chinese Society for Geospatial Information Society. He is also a member of the Editorial Board of *Journal of Applied Remote Sensing and Geographical Information Science*.

Liangpei Zhang (Fellow, IEEE) received the B.S. degree in physics from Hunan Normal University, Changsha, China, in 1982, the M.S. degree in optics from the Xi'an Institute of Optics and Precision Mechanics, Chinese Academy of Sciences, Xi'an, China, in 1988, and the Ph.D. degree in photogrammetry and remote sensing from Wuhan University, Wuhan, China, in 1998.

He is a "Chang-Jiang Scholar" Chair Professor appointed by the Ministry of Education of China at the State Key Laboratory of Information Engineering in Surveying, Mapping, and Remote Sensing (LIESMARS), Wuhan University. He was a Principal Scientist for the China State Key Basic Research Project from 2011 to 2016 appointed by the Ministry of National Science and Technology of China to lead the Remote Sensing Program in China. He has published more than 700 research articles and five books. He is the Institute for Scientific Information (ISI) Highly Cited Author. He holds 30 patents. His research interests include hyperspectral remote sensing, high-resolution remote sensing, image processing, and artificial intelligence.

Dr. Zhang is a fellow of the Institution of Engineering and Technology (IET). He was a recipient of the 2010 Best Paper Boeing Award, the 2013 Best Paper ERDAS Award from the American Society of Photogrammetry and Remote Sensing (ASPRS), and the 2016 Best Paper Theoretical Innovation Award from the International Society for Optics and Photonics (SPIE). His research teams won the top three prizes of the IEEE GRSS 2014 Data Fusion Contest. His students have been selected as the Winners or a Finalists of the IEEE International Geoscience and Remote Sensing Symposium (IGARSS) Student Paper Contest in recent years. He is also the Founding Chair of the IEEE Geoscience and Remote Sensing Society (GRSS) Wuhan Chapter. He also serves as an Associate Editor or an Editor for more than ten international journals. He is also serving as an Associate Editor for the IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING.