





TR-MISR: Multiimage Super-Resolution Based on Feature Fusion With Transformers

Tai An , Xin Zhang , Chunlei Huo , *Member, IEEE*, Bin Xue, Lingfeng Wang , *Member, IEEE*, and Chunhong Pan, *Member, IEEE*

Abstract—Multiimage super-resolution (MISR), as one of the most promising directions in remote sensing, has become a needy technique in the satellite market. A sequence of images collected by satellites often has plenty of views and a long time span, so integrating multiple low-resolution views into a high-resolution image with details emerges as a challenging problem. However, most MISR methods based on deep learning cannot make full use of multiple images. Their fusion modules are incapable of adapting to an image sequence with weak temporal correlations well. To cope with these problems, we propose a novel end-to-end framework called TR-MISR. It consists of three parts: An encoder based on residual blocks, a transformer-based fusion module, and a decoder based on subpixel convolution. Specifically, by rearranging multiple feature maps into vectors, the fusion module can assign dynamic attention to the same area of different satellite images simultaneously. In addition, TR-MISR adopts an additional learnable embedding vector that fuses these vectors to restore the details to the greatest extent. TR-MISR has successfully applied the transformer to MISR tasks for the first time, notably reducing the difficulty of training the transformer by ignoring the spatial relations of image patches. Extensive experiments performed on the PROBA-V Kelvin dataset demonstrate the superiority of the proposed model that provides an effective method for transformers in other low-level vision tasks.

Index Terms—Deep learning, end-to-end networks, feature extraction and fusion, multiimage super-resolution (MISR), remote sensing, transformers.

I. INTRODUCTION

IMAGE super-resolution, as one of the critical technologies in computer vision, aims to convert low-resolution images into high-resolution images. High-resolution images bring rich high-frequency details and play an essential role in medical

Manuscript received October 30, 2021; revised December 2, 2021 and December 27, 2021; accepted January 12, 2022. Date of publication January 18, 2022; date of current version February 2, 2022. This work was supported in part by the National Key Research and Development Program of China under Grant 2018AAA0100400, in part by the National Natural Science Foundation of China under Grant 62071466 and Grant 61802407, and in part by the Guangxi Natural Science Foundation under Grant 2018GXNSFBA281086. (*Corresponding author: Chunlei Huo.*)

Tai An, Xin Zhang, and Bin Xue are with the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China, and also with the School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing 100049, China (e-mail: antai2018@ia.ac.cn; xin.zhang2018@nlpr.ia.ac.cn; xuebin2018@ia.ac.cn).

Chunlei Huo, Lingfeng Wang, and Chunhong Pan are with the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China (e-mail: clhuo@nlpr.ia.ac.cn; lfwang@nlpr.ia.ac.cn; chpan@nlpr.ia.ac.cn).

Digital Object Identifier 10.1109/JSTARS.2022.3143532

Data is available at online at <https://github.com/Suanmd/TR-MISR>

imaging [1], face recognition [2], [3], video [4], [5], remote sensing [6]–[10], and other fields. Super-resolution technologies based on the remote sensing data need to satisfy the actual needs, such as disaster monitoring, land cover mapping [11], [12], and vegetation growth [13]. Limited by the cost and transmission rate [14], satellites mostly cannot obtain a large number of high-resolution images. In that case, researchers have been actively looking for effective super-resolution methods, such as discrete wavelet transforms (DWTs) [15], [16], and sparse coding [17]. With the rapid development of deep learning, the convolutional neural network (CNN) also serves as the main tool in the super-resolution of satellite images [18]–[20].

As for image super-resolution, two subdirections are worth studying: Single image super-resolution (SISR) and multiimage super-resolution (MISR). MISR focuses on extracting the information of multiple low-resolution images collected from the same scene and then merging them into a high-resolution image. In recent years, the increasing practical needs have led to an expansion of the low-cost satellite market. However, low-cost satellites always suffer from low-resolution images. In this context, MISR has become a key technology enabling more satellites to serve users better with its low cost and high quality. At present, deep learning methods have shown great success on SISR [21], [22], but existing deep learning methods of MISR, especially the end-to-end networks, progress at a slow pace.

In remote sensing, MISR tasks involve two issues that need consideration. First, the time span of acquiring multiple images may be so long that environmental factors become uncontrollable. Sometimes even the order of the images is unknown [23]. Therefore, how to process multiple low-resolution images with weak temporal correlations becomes a significant challenge. Currently, most existing MISR methods [18], [24]–[30] fix the input sequence, which means they handle these multiple images as order frames by default, so their results are sensitive to the sequence order. Some methods [29], [30] randomly shuffle the input sequence and obtain a robust result by computing the mean image during inference. The second issue is the insufficient utilization of multiple images. Under the influence of the actual environment, different views of the same scene may have different clearness (i.e., the number of clear pixels). Some existing methods based on deep learning [26], [27], [29], [30] put forward a high clearness requirement for an image and discard unclear images, which results in a loss of useful information. An ideal solution to reduce information loss is introducing

the attention mechanism to improve the utilization of multiple images.

However, there is no widely used attention module yet in MISR. As for low-level vision tasks, the attention mechanism has two major categories: Channel attention [31] and spatial attention [32]. Because the input is a stack of multiple images, channel attention, which assigns weights to different images, is equivalent to frame attention [33]. Spatial attention is equivalent to the weights given to different areas in the same image. Moreover, MISR requires focusing on specific areas of images regardless of their position, which means that the existing attention modules [27], [30] need to be improved.

From another perspective of attention mechanism, nonlocal attention focuses on the entire entry status space compared to local attention [34]. A typical framework is transformer [35], which bases on self-attention and can address the main problems faced by MISR in remote sensing. On the one hand, compared with the recurrent networks that require recursive processing of sequence elements, self-attention is insensitive to the sequence order. Transformer assigns dynamic attention to all elements simultaneously, making itself more capable of capturing long-term features and adapting to multiple satellite images with weak temporal correlations. On the other hand, unlike CNN that calculates static weights, self-attention calculates weights dynamically and adapts to variable-length sequences, which improves the utilization of multiple images.

Two points [36], [37] are limiting the application of transformers in remote sensing. First, transformer only takes a sequence of one-dimensional vectors as input. The image or feature should be flattened into a sequence of one-dimensional vectors before being fed into a transformer. In this way, transformer requires a large number of data to learn the spatial relations between vectors [38]–[42]. Second, transformer is inseparable from the pretraining of massive data compared to CNN with inductive bias. However, it is not easy to carry out large-scale pretraining for remote sensing datasets with high specificity. Transformer will not perform as well as CNN in some low-level vision tasks without pretraining [42], [43]. Thus, a transformer-based MISR framework needs to reduce the difficulty of training the transformer in a proper way.

To address the problems of MISR, we propose a novel end-to-end network based on transformers, namely, TR-MISR. TR-MISR alleviates the two limitations above of the transformer. On the one hand, TR-MISR does not destroy the two-dimensional structure of images. On the other hand, it receives a sequence of feature vectors that are fused without positional encoding [35], [38]. TR-MISR works as follows: First, the encoder encodes a set of image features. Then the transformer-based fusion module pays attention to feature vectors of the same area from different images and adopts an additional learnable embedding vector to fuse these features, i.e., no matter what area of a high-resolution image is generated, the corresponding areas of all low-resolution images can be observed simultaneously. Finally, the decoder generates the corresponding area of the high-resolution image by decoding the learnable embedding vector. Our method not only accommodates an arbitrary number of image patches but also adapts to images with a long time span. TR-MISR has

reached the state of the art on the PROBA-V Kelvin dataset released by the European Space Agency [18] and provided an effective strategy for transformers that will be compatible with other specific low-level vision tasks in the future.

In short, TR-MISR has the following highlights.

- 1) Our proposed method introduces transformer to MISR tasks in remote sensing for the first time. TR-MISR does not require pretraining, because the fusion module reduces the training difficulty of the transformer by preventing it from additionally learning the spatial relations between different image patches. This advantage alleviates the problem of insufficient MISR data in remote sensing.
- 2) TR-MISR provides a new approach for addressing the issues faced by MISR. With our proposed feature rearrangement module, TR-MISR can simultaneously focus on all image patches and adapt to multiple images with weak temporal correlations. In this way, TR-MISR can accommodate image sequences of any length, which notably improves the utilization of multiple images.
- 3) The transformer-based fusion module significantly improves the robustness of the model to noise. Compared with the existing deep learning-based MISR methods, TR-MISR can receive more unclear input images without performance degradation. Experiments show that our proposed fusion module performs better while maintaining lower #FLOPs and #Params.

The organization of this article is as follows: Section II reviews the methods of previous work. Section III presents our method and introduces the framework of TR-MISR. Section IV gives the results of the experiment under different hyperparameters and makes comparisons with other methods. Finally, Section V concludes the article.

II. RELATED WORK

Super-resolution has two primary directions according to the input data: Image super-resolution (ISR) and video super-resolution (VSR). The main difference between VSR and ISR is that the former needs to process consecutive frames and use interframe information. MISR, which faces many challenging problems, needs to process multiple images rather than consecutive frames.

In this section, we describe the transformer as well as three directions of super-resolution.

A. Transformer

With the interdisciplinary development of deep learning, a novel framework named transformer [35] stands out in natural language processing (NLP) and computer vision (CV). Transformer is a sequence-to-sequence framework based on self-attention. Essentially, self-attention is one type of nonlocal attention that can compute mutual attention on all elements of a sequence at one time. Transformer has been successfully applied to image classification [38], [39], [44], object detection [40], [45], semantic segmentation [41], [46], and other high-level vision tasks, but it rarely has been applied to low-level vision tasks such as reference-based super-resolution [47], multitask

image processing [42], image colorization [43], and video super-resolution [48]. Section III-A describes the structure of the Transformer in detail.

B. Single Image Super-Resolution

SISR is the technology that utilizes a single low-resolution image to reconstruct a single high-resolution image. In most cases, the low-resolution image comes out of a degraded high-resolution image. The mainstream algorithms of SISR involve three types. The interpolation-based methods [49] are fast and straightforward but generate images with insufficient details. The reconstruction-based methods [50], which require complex prior knowledge to limit the solution space and generate a high computational cost. The learning-based methods [51], [52] focus on learning the relationship between low-resolution images and high-resolution images, which can reduce the computational overhead and generate detailed results. With the development of deep learning, SISR has also adopted some basic frameworks such as CNN, recurrent neural network (RNN), and generative adversarial network [5], [21], [53]–[55] to further improve the performance and generate better details.

In remote sensing, the frameworks based on deep learning still play an essential role, such as the super-resolution of multispectral satellite images [56], [57] and the Sentinel-2 satellite imagery [58]–[60].

C. Video Super-Resolution

VSR uses a sequence of low-resolution frames to generate a sequence of high-resolution frames. The pipeline of most VSR methods includes a registration module, a feature extraction module, a fusion module, and a reconstruction module [61]. There are two types of VSR methods: Methods based on motion estimation and motion compensation (MEMC) as well as learning-based methods. Both of them emphasize the utilization of interframe information. Compared with traditional methods, deep learning is more capable of information extraction and feature fusion. It has played an essential role in estimations such as optical flow estimation [62], [63], and kernel estimation [64].

In remote sensing, the super-resolution of video imagery relies on video satellites, such as the Jilin-1 satellite [65], [66] that can directly collect videos instead of static images. At present, the super-resolution research of remote sensing video imagery is still subject to public datasets.

D. Multiimage Super-Resolution

MISR reconstructs a high-resolution image from multiple low-resolution images. Beyond single image and video processing, effectively combining more data means more reliable results [67]–[69]. For instance, a general way to multispectral image super-resolution is pan-sharpening [70]–[72], which combines prior information and fuses multiple images. When the camera takes multiple images of the same scene from different shooting angles, views, and times, more details will be obtained, which is of great help to reconstruct high-resolution images. Usually, multiple images are captured within a few seconds by

burst shooting devices, such as mobile phones and cameras with a burst mode, or captured by satellites for several days or months. Images captured by burst shooting devices have a strong inter-frame correlation, but images captured by satellites have a weak correlation due to environmental factors, landscape changes, or missing annotations [23]. MISR tasks with a strong interframe correlation are also called multiframe super-resolution (MFSR), but most studies regard MISR as MFSR without considering the inter-frame correlations. In this section, this article does not distinguish these two concepts. In raw image scenes, the camera obtains multiple images through burst shooting, such as the jitter camera prototype [73] applied on cameras and the handheld image super-resolution [74] applied on mobile phones.

At present, most deep learning-based MISR methods employ the encoder–decoder network, which includes a registration module, an encoder, a fusion module, and a decoder, to finish specific tasks. The process has the following main steps: First, the network fixes the sequence of multiple low-resolution images through sampling and padding. After the encoder extracts image features from multiple low-resolution images, the fusion module fuses these features into a fused feature. Finally, the decoder decodes the fused feature and gets a high-resolution image. Fig. 1 demonstrates this process. In remote sensing, the stacking CNN layers are responsible for implementing the encoder and decoder, and some of the encoder designs include specific attention mechanism [27], [30]. Besides, the fusion module has multiple designs such as fixed network design [18], rule-based design [24], RNN-based design [25], and 3-D convolution-based design [26]–[30].

For convenience, here we denote the super-resolution scaling factor by r , the low-resolution images of one scene by $\{LR_i\}_{i=1}^k \in \mathbb{R}^{C \times W \times H}$, and the high-resolution image (i.e., the ground truth) by $HR \in \mathbb{R}^{C \times rW \times rH}$, where k , C , H , and W represents the number, the channel, the height, and the width of images, respectively. HighRes-Net [24] is an end-to-end framework. It encodes $\{LR_i\}_{i=1}^k$ and gets features $\{S_i\}_{i=1}^k$. The features are later merged in pairs by recursive design through $t = \lceil \log_2 k \rceil$ steps. Then the network decodes the fused feature S^t to obtain the high-resolution image. MISR-GRU [25] uses convGRU [75] to extract the hidden states $\{h_i\}_{i=1}^k$ at each time step and obtains the fused state h_u through global average pooling (GAP). DeepSUM [26] widely uses 3-D convolutions to integrate different channels, realizing the feature-wise registration and fusion. The fusion module of DeepSUM, which is also a recursive design, requires $k/2$ 3-D convolutional layers to fuse features of k registered images. DeepSUM++ [27] adds the graph convolution in the encoder and improves the performance with the help of introducing nonlocal attention.

Another approach learns from the mature structure of VSR. Inspired by 3DSRNet [76] applied to VSR, Francisco Dorr *et al.* [29] propose 3DWDSRNet, the core of which is to use the WDSR blocks to acquire the temporal correlations between frames. WDSR-MFSR [28] uses multiple WDSR residual blocks to strengthen the feature extraction further.

Francesco Salvetti *et al.* [30] realize the difference between MISR and MFSR and propose RAMS. RAMS tries to avoid the

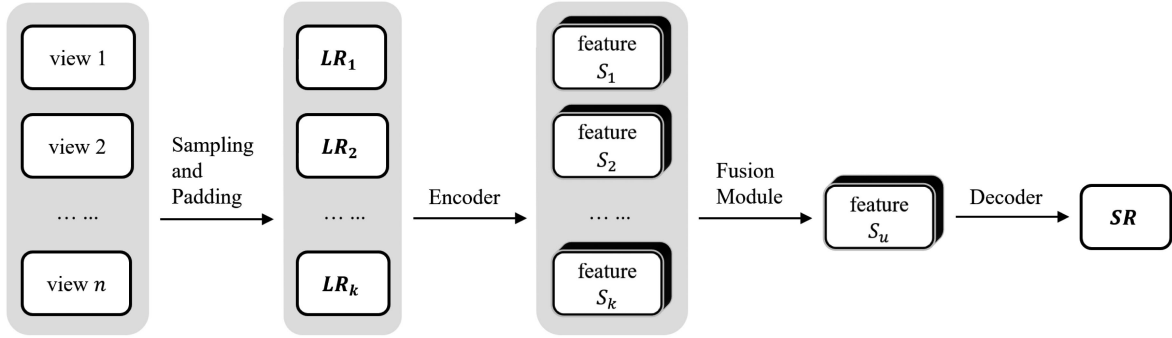


Fig. 1. Pipeline of MISR methods. The pipeline includes a pre-processing module for processing views, an encoder for extracting features, a fusion module for getting a fused feature, a decoder for reconstructing a high-resolution image, and a registration module for aligning images/features. Low-resolution images LR_1, \dots, LR_k obtained by preprocessing are encoded separately to get features S_1, \dots, S_k . The registration module can be inserted in any process of MISR, so we omit it in the figure.

effect of the sequence order on the results in various aspects. Specifically, it randomly shuffles the image sequence before images are fed into the network. In addition, it retains features to a great extent in the temporal and spatial dimensions by using the residual temporal attention blocks (RTABs) and the residual feature attention blocks (RFABs). In testing/verification, RAMS randomly shuffles the image sequence several times and averages these results to obtain a reliable high-resolution image, e.g., $RAMS_{+20}$.

In summary, most existing MISR methods mainly focus on image coding and use 3-D convolutions to fuse features gradually. Although the 3-D convolution can merge information across channels, it is sensitive to the sequence order and noise. Also, it still requires a fixed number of input images, which makes most methods less practical in MISR.

III. METHODOLOGY

In this section, we describe the structure of TR-MISR. First, we briefly introduce the transformer, especially the self-attention mechanism. Then we introduce three modules of TR-MISR in detail and finally present the loss function.

A. Structure of Transformer

Transformer [35] is a sequence-to-sequence framework based on self-attention. It consists of a set of encoders and decoders. The encoder of a transformer includes a self-attention layer and a feed-forward network. We assume that the input sequence of a self-attention layer is $X = [x_1, \dots, x_n] \in \mathbb{R}^{d \times n}$ and the output sequence is $Y = [y_1, \dots, y_n] \in \mathbb{R}^{d \times n}$. For each $x_i \in X$, the self-attention layer linearly maps x_i to the query vector $q_i = W_q x_i \in \mathbb{R}^{d \times 1}$, the key vector $k_i = W_k x_i \in \mathbb{R}^{d \times 1}$, and the value vector $v_i = W_v x_i \in \mathbb{R}^{d \times 1}$. Here d may not be equal to d' . For each q_i , the output vector y_i can be expressed by

$$y_i = \sum_{j=1}^n \left[\text{softmax} \left(\frac{q_i^T K}{\sqrt{d}} \right) \right]_j v_j \quad (1)$$

where $K = [k_1, \dots, k_n] \in \mathbb{R}^{d \times n}$ represents the key matrix, d represents the length of x_i and y_i . Each y_i is equivalent to the

weighted sum of $\{v_j\}_{j=1}^n$. The weights of $\{v_j\}_{j=1}^n$ depend on the correlations between $\{q_i\}_{i=1}^n$ and $\{k_i\}_{i=1}^n$. Note that, the attention weights are dynamically generated and not affected by the position of x_i .

The decoder of a transformer includes three parts. The self-attention layer is responsible for establishing the relationship between the current decoded value and the decoded part. The encoder-decoder attention layer establishes the relationship between the current decoded value and the encoded feature vectors. The feed-forward network is the same as that in the transformer encoder.

Moreover, by using multiheaded self-attention, transformer maps the query vector q_i and the key vector k_i to different subspaces $\{\mu_1, \dots, \mu_q\}$ of the high-dimensional space μ and calculate their similarity. The formation of multiple subspaces aims to pay attention to different aspects of features [35]. The transformer concatenates the results $\{Y^i\}_{i=1}^q$ produced by $\{\text{head}^i\}_{i=1}^q$ and multiplies it by the weight W^o to get the multiheaded output Z :

$$Z = \text{Concat}(Y^1, \dots, Y^q) \times W^o \quad (2)$$

where q denotes the number of multiheads. $Z = [z_1, \dots, z_n] \in \mathbb{R}^{d \times n}$ has the same size as the input X . We use $\text{MSA}()$ to represent multiheaded self-attention so that the self-attention of the transformer can be briefly represented by

$$[z_1, \dots, z_n] = \text{MSA}([x_1, \dots, x_n]). \quad (3)$$

The feed-forward network includes two fully connected layers that generally serve as the information integration layers. These layers mainly provide nonlinear transformations for the transformer and enhance its expression capability.

B. TR-MISR Framework

TR-MISR takes any number of low-resolution images as input and outputs a high-resolution image. It consists of an encoder, a fusion module, and a decoder. In the encoder, the network extracts the features through the residual blocks. Then, these features are assigned dynamic attention and fused by an additional learnable embedding vector in the fusion module.

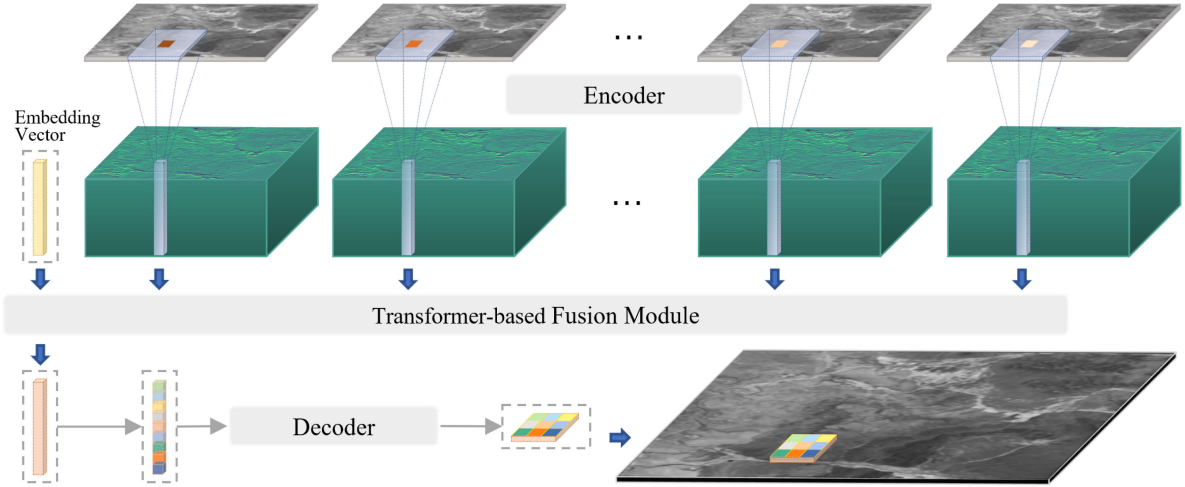


Fig. 2. Overview of TR-MISR. The encoder based on residual blocks encodes the same area of different low-resolution images. The fusion module based on transformer is responsible for a featurewise fusion by an additional learnable embedding vector. The decoder based on subpixel convolution decodes all fused features into a high-resolution image. The parameters of the encoder are shared.

Finally, the decoder relies on sub-pixel convolution to obtain a high-resolution image. Fig. 2 shows the overview of TR-MISR. Referring to the feature vectors extracted from the same area of low-resolution images, TR-MISR generates the corresponding area of the high-resolution image. In this way, nonlocal attention is introduced to different image patches, significantly improving the applicability of our model.

Next, we will introduce these modules in detail.

1) *The Encoder*: We represent the low-resolution images $\{LR_i\}_{i=1}^k$ by a tensor of size $B \times H \times W \times K \times C_{in}$, where B is the batch size. H , W , K , and C_{in} represent the image height, the image width, the number of images, and the channels per image, respectively. The encoder refers to HighRes-Net [24] based on residual blocks. MISR assumes that the information of multiple images is greater than that of any single image. However, the redundant information of multiple low-resolution images will cause trouble for models to extract different features. To mitigate this impact, we calculate a reference image LR_{ref} of an image sequence by the Median() function that highlights differences across multiple images [77]. The reference image LR_{ref} is used both as a shared representation and a condition for implicit registration between images $\{LR_i\}_{i=1}^k$. Each low-resolution image LR_i is concatenated with LR_{ref} and then fed into the encoder to get the feature map f_i . The formulas are as follows:

$$LR_{ref} = \text{Median}(LR_1, \dots, LR_k) \quad (4)$$

$$G^i = [LR_i, LR_{ref}], \quad i = 1, \dots, k \quad (5)$$

$$f_i = \text{Encoder}(G^i), \quad i = 1, \dots, k \quad (6)$$

where k represents the number of input images. The Median() function aims to calculate the median image LR_{ref} from $\{LR_i\}_{i=1}^k$. For convenience, we omit the batch size dimension. The feature map $f_i \in \mathbb{R}^{H \times W \times C_h}$ obtained from $LR_i \in \mathbb{R}^{H \times W \times C_{in}}$ has expanded the number of channels (i.e., $C_h >$

C_{in}). Here k can be set manually. For scenes where the number of original views n is less than k , $(k - n)$ padded images need to be generated; otherwise, images need sampling. We use a Boolean variable α_i to mark whether LR_i is a padded image or not.

2) *Fusion Module*: Our transformer-based fusion module keeps the spatial resolution of the input and output unchanged, while effectively capturing global context information. In self-attention, the query vector is a sequence that needs expressing by the following steps: First, the module calculates the similarity between the query vectors and key vectors in the same high-dimensional space. Then it computes a weighted sum of value vectors to express the sequence with attention.

Transformer can extract a set of query vectors $\{q_i\}_{i=1}^k$, key vectors $\{k_i\}_{i=1}^k$, and value vectors $\{v_i\}_{i=1}^k$ from each set of feature vectors $\{x_i\}_{i=1}^k$. These vectors $\{q_i\}_{i=1}^k$, $\{k_i\}_{i=1}^k$, and $\{v_i\}_{i=1}^k$ are calculated by (1) to obtain the output $\{y_i\}_{i=1}^k$ with mutual attention. Thus, the output y_0 of a manually added vector x_0 is the fused feature of $\{x_i\}_{i=1}^k$. In this context, we add a learnable embedding vector to each set of features that is randomly initialized. Through multiple transformer layers, the learnable embedding vector can pay attention to the same area of different low-resolution images.

Specifically, the fusion module divides f_i into $H \times W$ one-dimensional vectors $x_{(h,w)}^i \in \mathbb{R}^{d_0 \times 1}$ along the channel dimension. Each vector $x_{(h,w)}^i$ has $d_0 = C_h$ features encoded by the receptive field that takes the coordinate (h, w) of LR_i as center. As shown in Fig. 2, the module transforms $\{f_i\}_{i=1}^k \in \mathbb{R}^{H \times W \times K \times C_h}$ into $H \times W$ groups $\{x_{(h,w)}^i\}_{i=1}^k \in \mathbb{R}^{K \times C_h}$ and adds a learnable embedding vector $x_{(h,w)}^0 \in \mathbb{R}^{d_0 \times 1}$ at the beginning of each sequence $\{x_{(h,w)}^i\}_{i=1}^k$ to obtain $z_{(h,w)}^0$. Then $z_{(h,w)}^0 = [x_{(h,w)}^0, x_{(h,w)}^1, \dots, x_{(h,w)}^k]$ is fed into the transformer. Finally, the output $z_{(h,w)}^u$ that $x_{(h,w)}^0$ generates through the transformer layers fuses k vectors $\{x_{(h,w)}^i\}_{i=1}^k$. This kind of feature

fusion is a parallel computing process with the permutation invariance property.

For each feature vector $x_{(h,w)}^i$ extracted from the feature map f_i at each position (h, w) , the module transforms them like above and uses the transformer to fuse features. We retain the entire structure of the transformer encoder, including the multiheaded self-attention (MSA) layers, the feed-forward networks (FFN), layer normalization (LN), and skip connections. For simplicity, we omit all subscripts (h, w) . The whole fusion processes are represented by

$$z^0 = [x^0, x^1, \dots, x^k] \quad (7)$$

$$\alpha = [1, 1, \dots, 1] \text{ or } [1, 1, \dots, 0] \quad (8)$$

$$z_a^l = \text{MSA}(\text{LN}(z^{l-1}, \alpha)) + z^{l-1}, \quad l = 1, \dots, M \quad (9)$$

$$z^l = \text{FFN}(\text{LN}(z_a^l)) + z_a^l, \quad l = 1, \dots, M \quad (10)$$

$$z^u = z^M|_{\text{index}=0} \quad (11)$$

where M represents the number of the transformer layers. $z^u \in \mathbb{R}^{d \times 1}$ is the first element of z^M . After $H \times W$ groups of the fusion process, the module obtains a fused feature $f_u \in \mathbb{R}^{H \times W \times C_h}$ from $\{f_i\}_{i=1}^k \in \mathbb{R}^{H \times W \times K \times C_h}$. The original transformer requires a large amount of data to learn the spatial relations of image patches [38]. Moreover, the fusion module we proposed avoids the additional learning of spatial relations, significantly reducing the difficulty of training the transformer.

3) *Decoder*: In super-resolution tasks, researchers mainly use a decoder that increases the resolution of feature maps to obtain a high-resolution image. Suppose a deconvolution is used for upscaling directly. In that case, the convolution will face zero-filled images, which introduces unnecessary calculations or even produces the checkerboard artifacts [78].

Subpixel convolution [5], also known as pixel shuffle, is an efficient and parameter-free pixel rearrangement method. We use a subpixel convolution as the primary part of the decoder. It guides the feature map to generate subpixels of the high-resolution image. In the fusion module, the feature map is composed of $H \times W$ numbers of $z_{(h,w)}^u$. Specifically, $z_{(h,w)}^u$ serves as the common representation of low-resolution images at the same position centered on (h, w) . Later it is decoded to the $r \times r$ patch in the high-resolution image, as illustrated in Fig. 2. The patch takes $((r-1)h+1, (r-1)h+1)$ as the coordinate of the upper-left corner and takes (rh, rw) as the coordinate of the lower-right corner, where r is the super-resolution scaling factor.

The work of the decoder has two steps, involving a general convolution operation and a subpixel convolution. First, the convolution is to reduce the channels C_h of $f_u \in \mathbb{R}^{H \times W \times C_h}$ to $C_t = r^2 C_{in}$ and get a response map $TR \in \mathbb{R}^{H \times W \times C_t}$. Then, the subpixel convolution is for decoding TR to a high-resolution image $SR \in \mathbb{R}^{rH \times rW \times C_{in}}$. In other words, starting from one pixel at the same position in feature maps $\{f_i\}_{i=1}^k$, the decoder generates r^2 pixels in the corresponding position of the high-resolution image SR . We choose PReLU as the activation function. The specific process is

$$TR = \text{PReLU}(\text{Conv}(f_u)) \quad (12)$$

$$SR_{(h,w,c)} = TR_{\left(\lfloor \frac{h}{r} \rfloor, \lfloor \frac{w}{r} \rfloor, c \cdot r \cdot \text{mod}(w,r) + c \cdot \text{mod}(h,r)\right)} \quad (13)$$

where $\text{Conv}()$ denotes a 2-D convolution, $\text{mod}(\cdot, \cdot)$ returns the remainder of two numbers after division. Equation (13) is to map the response values of TR to the pixel values of the high-resolution image SR , and the operation is parameter-free.

The TR-MISR framework is shown in Fig. 3.

C. Loss Function

We use supervised learning to train TR-MISR in an end-to-end manner. TR-MISR takes multiple images $\{LR\}_{i=1}^k$ as input and outputs a single image SR . Because low-resolution images $\{LR\}_{i=1}^k$ and the ground truth HR are in different collection environments, SR generates some different problems from HR , such as horizontal/vertical offset, overall brightness deviation, and pixel occlusion or smear.

Assuming that the image offset is consistent across the entire image, TR-MISR requires cropping HR and SR to compensate for the image offset and return an appropriate loss. The maximum value of the offset required for registration is c . On the one hand, the model crops the four edges of SR by c pixels and obtains $SR_{(c,c)}$. On the other hand, it crops HR that takes (u, v) as the starting coordinate to the same size as $SR_{(c,c)}$ and obtains $HR_{(u,v)}$, where $(u, v) \in \{0, \dots, 2c\}$. Then the model returns the smallest loss of $SR_{(c,c)}$ and $HR_{(u,v)}$ for each SR - HR pair.

Considering the overall brightness deviation, we first correct the overall brightness of $SR_{(c,c)}$ and $HR_{(u,v)}$ by calculating the average deviation and then calculate the loss function. The mean absolute error l_1 and the mean square error l_2 are the most common pixel-level loss functions. In general, l_1 aims to get a more balanced error distribution, while l_2 focuses on penalizing error pixels, aiming to reduce mispredictions.

In summary, we can calculate the average brightness deviation $b_{(u,v)}$ between $SR_{(c,c)}$ and $HR_{(u,v)}$ by clear pixels, then choose either l_1 or l_2 loss as the loss function

$$b_{(u,v)} = \frac{SM_{(u,v)}}{\sum SM_{(u,v)}} (HR_{(u,v)} - SR_{(c,c)}) \quad (14)$$

$$l_1 = \min_{u,v} \frac{|(HR_{(u,v)} - SR_{(c,c)} - b_{(u,v)}) \cdot SM_{(u,v)}|}{\sum SM_{(u,v)}} \quad (15)$$

$$l_2 = \min_{u,v} \frac{[(HR_{(u,v)} - SR_{(c,c)} - b_{(u,v)}) \cdot SM_{(u,v)}]^2}{\sum SM_{(u,v)}} \quad (16)$$

where the binary image $SM_{(u,v)}$ denotes the pixel quality indicator of $HR_{(u,v)}$.

IV. EXPERIMENTS

In this section, we conduct experiments in detail on the existing MISR dataset. First, the dataset and evaluation metrics are described. Second, comparisons between our method and existing methods are introduced. Then the effect of the hyperparameters and the ablation study of fusion modules are presented. Finally, the attention from the output vector to the input sequence is visualized.

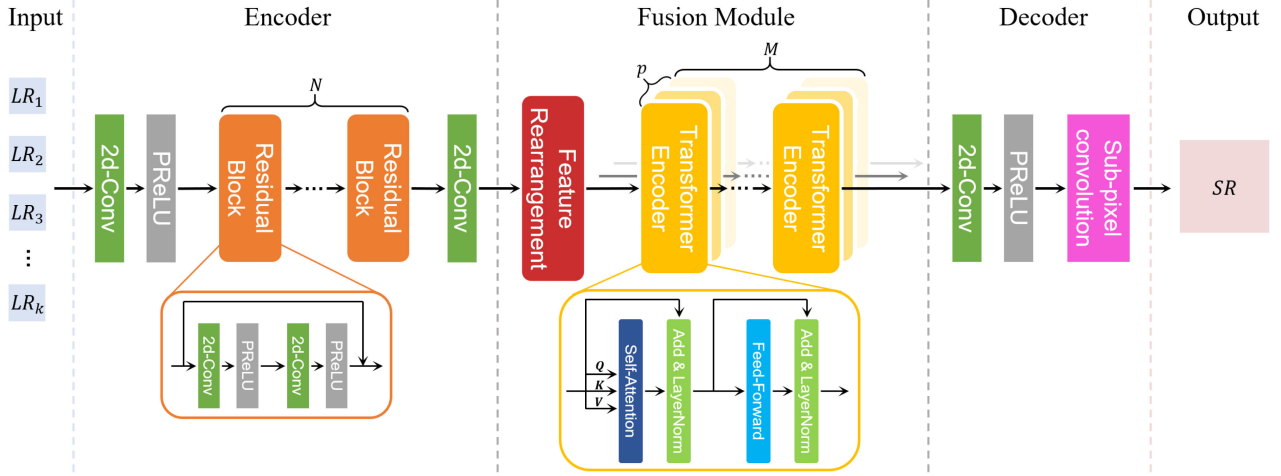


Fig. 3. TR-MISR framework. k is the number of images fed into the shared encoder. N is the number of residual blocks, which determines the $(2N + 2) \times (2N + 2)$ size of the receptive field. p represents the number of attention heads in the transformer, which is mainly used to pay attention to different features. M is the number of the Transformer layers, which can be flexibly adjusted based on the size of datasets. In the feature rearrangement, multiple feature maps are rearranged into pairs of feature vectors along the channel dimension. It ensures that each sequence fed into the transformer belongs to the same area of different images.

A. PROBA-V Kelvin Dataset

PROBA-V is an earth observation satellite designed to map land cover and vegetation growth. This dataset has been released by the Advanced Concept Team of the European Space Agency (ESA) [18]. Unlike most open-source datasets for super-resolution, all the views provided by PROBA-V are real records instead of algorithm processing. For the same scene, the PROBA-V satellite provides views of two resolutions that have different revisit frequencies. The data collection spans 30 days in total, which means that some scenes may have changed. More importantly, the order of the views is unknowable, so researchers need to regard the views as a sequence with weak time correlations.

The PROBA-V Kelvin dataset involves 74 manually selected locations, a total of 1450 scenes that cover the red spectrum band RED and the near-infrared spectrum band NIR, of which 290 scenes are for testing. Each scene includes multiple low-resolution images with 300-m resolution (128×128 grayscale images, called $\{LR_i\}_{i=1}^n$) and a high-resolution image with 100 m resolution (384×384 grayscale image, called HR). The super-resolution scaling factor r is 3. These images saved in 16-bit png format have a bit-depth of 14. In addition, most images have missing or noisy values due to the actual environment, so each image (LR_i or HR) owns a manually labeled quality map (QM_i or SM) indicating the concealed pixels and clear pixels. Every scene has 19 images on average, to a minimum of 9 and a maximum of 35. Specifically, all images in each scene are not registered, and their brightness is often inconsistent.

B. Evaluation Metric

The realism of pixel content is an essential indicator in the remote sensing environment. To judge the usability of the high-resolution image SR , we need to consider two points in terms of the core evaluation indicators: 1) SR should not reconstruct

occlusions; 2) the actual pixel value of SR should be as close as possible to HR . Given these two requirements, the Advanced Concept Team of ESA puts forward the cPSNR metric [18] based on the square error between pixels.

PSNR is a critical metric for satellite image quality assessment. As a modified version of PSNR, cPSNR takes the differences between SR and HR into account. Both sizes of SR and HR are 384×384 with the maximum registration offset $c = 3$, so the starting coordinate is denoted by $(u, v) \in \{0, \dots, 6\}$. First, we can get $SR_{(3,3)}^b$ according to

$$SR_{(3,3)}^b = SR_{(3,3)} + b_{(u,v)} \quad (17)$$

where $b_{(u,v)}$ has been defined in (14).

Then, the cPSNR metric is

$$\begin{aligned} & \text{cPSNR}(SR, HR) \\ &= \max_{(u,v) \in \{0, \dots, 6\}} 10 \lg \frac{(2^D - 1)^2 \cdot \sum SM_{(u,v)}}{\left[\left(HR_{(u,v)} - SR_{(3,3)}^b \right) \cdot SM_{(u,v)} \right]^2} \end{aligned} \quad (18)$$

where D denotes the bit-depth of SR and HR . $SM_{(u,v)}$ denotes the pixel quality indicator of $HR_{(u,v)}$.

Similarly, the cSSIM metric based on SSIM [79] can be represented by

$$\begin{aligned} & \text{cSSIM}(SR, HR) \\ &= \max_{(u,v) \in \{0, \dots, 6\}} \text{SSIM} \left(HR_{(u,v)} \cdot SM_{(u,v)}, SR_{(3,3)}^b \cdot SM_{(u,v)} \right) \end{aligned} \quad (19)$$

SSIM generally uses the Gaussian function to calculate the mean and the covariance of an image, and it is not the primary evaluation metric of the PROBA-V Kelvin dataset. In the PROBA-V challenge, the evaluation metric of MISR is the average score based on cPSNR [see (18)]. To be specific, each scene has a baseline method, which obtains the base SR through

a merger of some bicubic upscalings of low-resolution images. The score R of a single scene is

$$R(SR, HR) = \frac{\text{cPSNR}(\text{base}SR, HR)}{\text{cPSNR}(SR, HR)}. \quad (20)$$

Given the average score of all test scenes, the final score \bar{R} is represented by

$$\bar{R} = \text{Mean}(R). \quad (21)$$

Score \bar{R} is used as a ranking index. If the proposed method is better than the baseline overall, \bar{R} should be smaller than 1. The smaller the \bar{R} , the better the results we get.

C. Experimental Settings

1) *Data Preparation*: Before training TR-MISR, we perform data augmentation by cropping and sampling. The steps are as follows.

- a) We crop low-resolution views to the 64×64 LR_i and crop the ground-truth view to 192×192 HR . We set the cropping stride to 64, i.e., crop a scene into 9 subscenes.
- b) Without rotation or flipping for subscenes, we only discard the unclear subscene where the average ratio of clear pixels is less than 85% and preserve all views in each scene.

At the front of the encoder, $\{LR_i\}_{i=1}^n$ need to be processed in the following steps.

- a) For $n \geq k$, we select k images by sampling; for $n < k$, blank images are added by padding.
- b) We select the image with the largest number of clear pixels in the scene as the reference LR^c . In addition, we register $\{LR_i\}_{i=1}^n$ with LR^c based on subpixel image translation registration [80].
- c) According to (4), we calculate LR_{ref} as the implicit coregistration information. Then each image LR_i needs to be encoded together with LR_{ref} .

Indeed, the implicit coregistration information LR_{ref} is crucial to TR-MISR. We will discuss it later in Section IV-E.

2) *Parameter Settings*: The experiments are conducted by a server cluster with a 64-bit Linux operating system. The hardware includes Tesla V100 GPU (32 GB memory) and Intel(R) Xeon(R) Gold 6230 CPU @ 2.10 GHz. As is shown in Fig. 3, we set the number of input images $k = 24$. In the encoder, we set $N = 2$ and expand the number of channels from $C_{in} = 2$ to $C_h = 64$. The kernel size of the 2-D convolution is set to 3×3 . The stride and the padding are set to 1. We choose PReLU as the activation function. In the fusion module, we set the number of transformer multiheads $p = 8$ and layers $M = 6$. In the transformer feed-forward network, the mapping dimension of fully connected layers is set to 128. In the decoder, we use a 2-D convolution to reduce the channels from $C_h = 64$ to $C_r = 9$. The kernel size of the 2-D convolution is set to 1×1 and the stride is set to 1. We choose PReLU as the activation function, and a subpixel convolutional layer with upscale factor $r = 3$ is connected after the PReLU activation function.

In terms of network training, we set the initial learning rate of the encoder and decoder to $1e-3$, the learning rate of the fusion module to $5e-4$. Experiments show that l_2 converges much faster

than l_1 , so we choose l_2 as the loss function. We choose Adam to train our network and design a learning rate decay based on the evaluation metric \bar{R} of the validation set. If \bar{R} is not improved three times in a row, the learning rate of all modules will drop by 5%. We set the batch size to 4 and the training epoch to 400. It takes about 60 h to train one band on a Tesla V100 GPU. After this, we fine-tune the model to obtain a slight improvement, i.e., set the learning rate of the encoder and decoder to $5e-5$ and set the learning rate of the fusion module to $2.5e-5$. Then it takes about 20 epochs of training to obtain the best model.

There are three main reasons why the training time is longer than that of most methods: First, transformer has a global attention with fully connected layers. Fully connected layers do not have an advantage over convolutional layers in terms of time and memory. Second, the model hyperparameters are large by default. The model will achieve a faster speed if we slightly lower the hyperparameters, e.g., the model size and the number of images. Third, because TR-MISR is an end-to-end framework, some built-in operations like rearranging vectors also decrease the computational speed. TR-MISR does not require pretraining, and all weights in the network are initialized randomly. In addition, all initialized seeds keep fixed to ensure that the experimental results are reproducible.

D. Comparing Methods

Because there is no ground truth on the PROBA-V Kelvin testing set, we split the training set at a ratio of 7:3. Given the same training/validation set, we choose the representative methods of SISR, VSR, and MISR, and compare them in terms of the single-band data NIR/RED and the full-band data ALL. The comparison metrics include cPSNR and cSSIM. Table I and Figs. 4–6 present the results.

Here the methods and their experimental settings to be compared with TR-MISR are briefly introduced.

1) SISR Methods:

- a) *Bicubic*: It is the baseline method. The clearest image LR^c in each scene is selected and performed the bicubic interpolation.
- b) *RCAN*: [31] It proposes channel attention (CA) to process different channels. The experiment sets the residual groups and the residual channels to 5.

2) VSR Method:

- a) *VSR-DUF*: [83] It uses dynamic upsampling filters to generate corresponding filters for different inputs. The experiment sets the number of input video frames to 9 and selects a 16-layer framework.

3) MISR Methods:

- a) *IBP*: [82] It is one of the most classic algorithms for image super-resolution that improves the resolution of images through iterations. The experiment uses bicubic interpolation to obtain the initial solution and uses the phase correlation algorithm for registration.
- b) *BTv*: [81] It is an image enhancement method, which focuses on restoring image edges and denoising. It minimizes an l_1 norm plus the bilateral regularization term in each iteration.

TABLE I
PERFORMANCE OF DIFFERENT METHODS ON THE VALIDATION SET

Method	NIR		RED		ALL	
	cPSNR	cSSIM	cPSNR	cSSIM	cPSNR	cSSIM
Bicubic	45.44	0.9770	47.33	0.9840	46.40	0.9806
BTV [81]	45.93	0.9794	48.12	0.9861	47.04	0.9828
IBP [82]	45.96	0.9796	48.21	0.9865	47.10	0.9831
RCAN [31]	45.66	0.9798	48.22	0.9870	46.96	0.9835
VSR-DUF [83]	47.20	0.9850	49.59	0.9902	48.42	0.9876
HighRes-Net [24]	47.55	0.9855	49.75	0.9904	48.67	0.9880
3DWDSRNet [29]	47.58	0.9856	49.90	0.9908	48.76	0.9882
DeepSUM [26]	47.84	0.9858	50.00	0.9908	48.94	0.9883
MISR-GRU [25]	47.88	0.9861	50.11	0.9910	49.01	0.9886
DeepSUM++ [27]	47.93	0.9862	50.08	0.9912	49.02	0.9887
RAMS [30]	48.17	0.9869	50.13	0.9910	49.17	0.9890
RAMS ₊₂₀ [30]	48.27	0.9870	50.27	0.9912	49.29	0.9891
TR-MISR(ours)	48.54	0.9882	50.67	0.9921	49.62	0.9902

The bold entities represent the best results for different evaluation metrics.

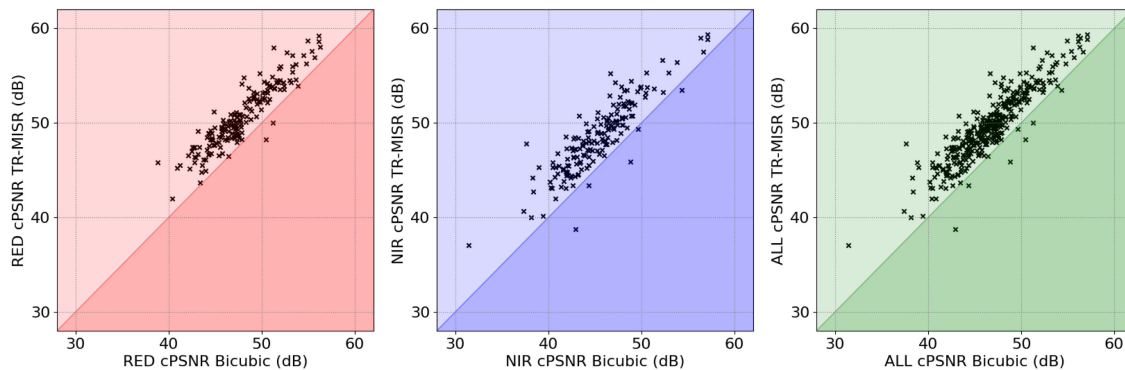


Fig. 4. cPSNR comparison between TR-MISR and bicubic interpolation on the validation set. Each data point represents a scene. A certain data point above the straight line $y = x$ indicates that TR-MISR can construct the corresponding scene better than the baseline.

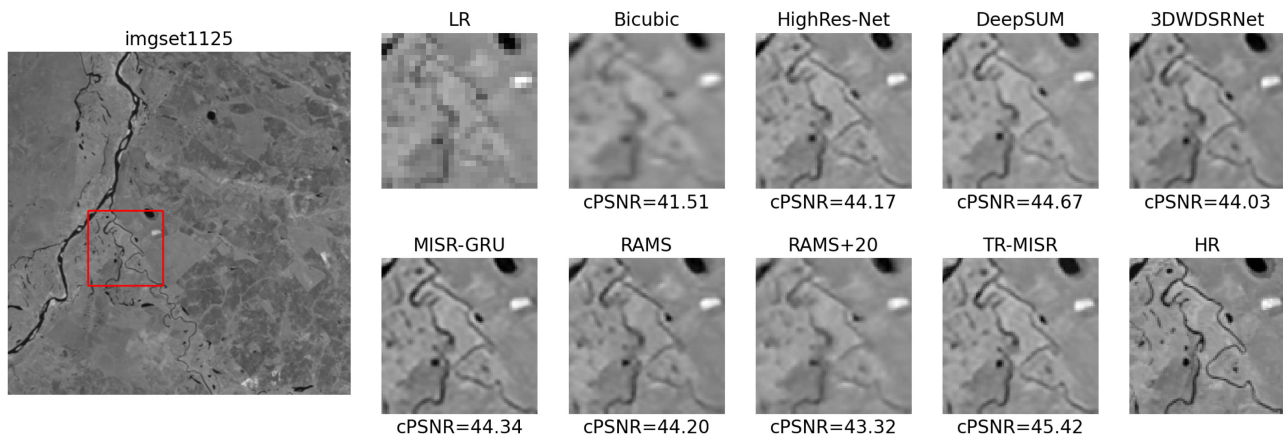


Fig. 5. Comparison between different MISR methods on the imgset1125 scene of the NIR band.

- c) *HighRes-net*: [24] As the runner-up of the PROBA-V challenge, it is an end-to-end framework for joint training of the encoder–decoder network and the registration network. The experiment adopts the default framework and sets the number of input images to 16.
- d) *MISR-GRU*: [25] It uses convGRU [75] to fuse different features and obtains the fused features by processing the

hidden states. The experiment sets the number of input images to 24.

- e) *DeepSUM*: [26] As the winner of the PROBA-V challenge, it is a deep framework focusing on exploring the spatio-temporal correlations between images. The experiment adopts 9 images. The authors have also released DeepSUM++ [27], which brings a slight

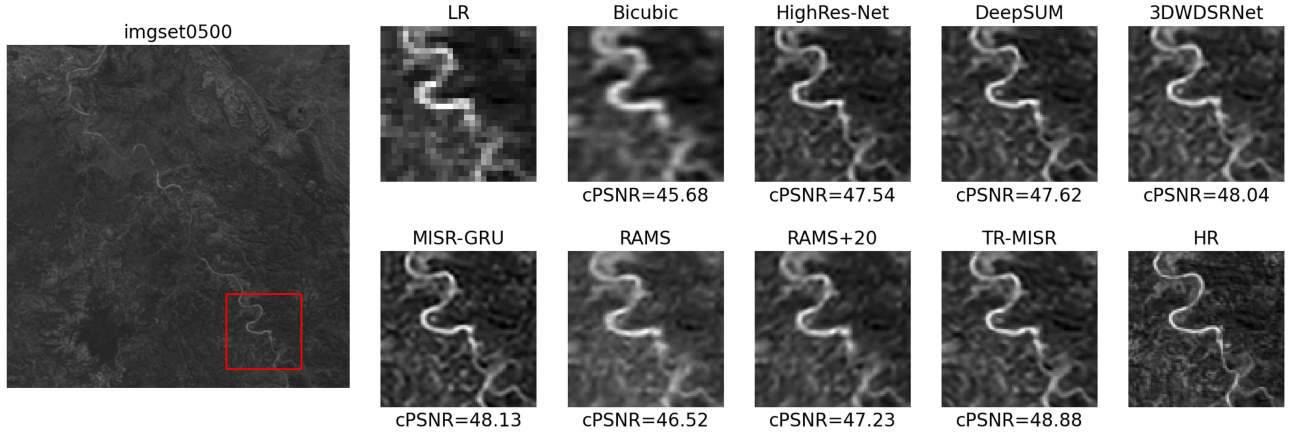


Fig. 6. Comparison between different MISR methods on the imgset0500 scene of the RED band.

TABLE II
cPSNR PERFORMANCE OF DIFFERENT ENCODER HYPERPARAMETERS ON THE VALIDATION SET

N	Transformer					small-Transformer				
	NIR	RED	ALL	#FLOPs(G)	#Params(M)	NIR	RED	ALL	#FLOPs(G)	#Params(M)
1-	48.17	50.40	49.31	31.09	0.310	47.87	50.14	49.03	21.02	0.211
1	48.27 $_{+0.10}$	50.51 $_{+0.11}$	49.41 $_{+0.10}$	31.15	0.311	47.95 $_{+0.08}$	50.24 $_{+0.10}$	49.11 $_{+0.08}$	21.08	0.212
2-	48.23	50.42	49.34	38.35	0.384	48.01	50.22	49.13	28.28	0.285
2	48.40 $_{+0.17}$	50.64 $_{+0.22}$	49.54 $_{+0.20}$	38.41	0.385	48.03 $_{+0.02}$	50.29 $_{+0.07}$	49.18 $_{+0.05}$	28.34	0.286
3-	48.20	50.40	49.32	45.61	0.458	47.96	50.19	49.10	35.54	0.359
3	48.36 $_{+0.16}$	50.63 $_{+0.23}$	49.52 $_{+0.20}$	45.67	0.458	48.13 $_{+0.17}$	50.36 $_{+0.17}$	49.26 $_{+0.16}$	35.60	0.359
4-	47.64	49.91	48.79	52.87	0.532	47.50	49.85	48.69	42.80	0.433
4	47.80 $_{+0.16}$	50.02 $_{+0.11}$	48.93 $_{+0.14}$	52.93	0.532	47.64 $_{+0.14}$	50.02 $_{+0.17}$	48.85 $_{+0.16}$	42.86	0.433

N represents the number of residual blocks in the encoder. '-' denotes without the calculation of implicit co-registration. #Params(M) indicates the model parameters. #FLOPs(G) indicates the computation amount of one scene during inference.

The bold entities represent the highest cPSNR results for different network hyperparameters.

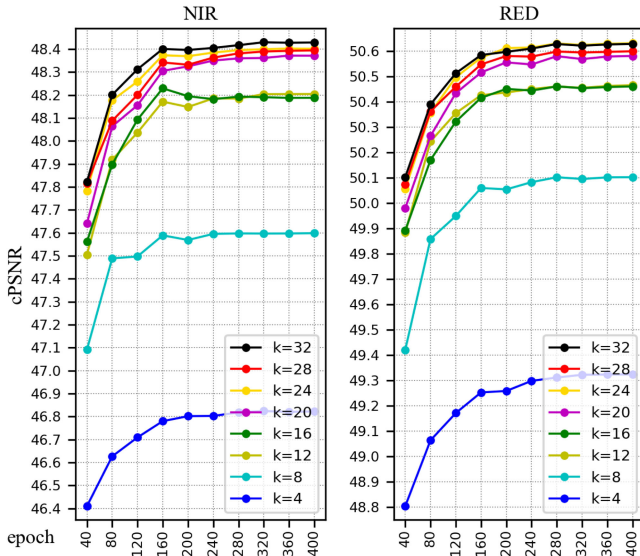


Fig. 7. Effect of different numbers of input images k on reconstruction outcomes. The vertical axis represents the cPSNR performance on the validation set. The horizontal axis represents the training epoch.

improvement by introducing the graph convolution into the encoder.

f) *3DWDSRNet*: [29]: It emphasizes the changing relations of the temporal dimension between acquired frames

TABLE III
cPSNR PERFORMANCE OF DIFFERENT TRANSFORMER HYPERPARAMETERS ON THE VALIDATION SET

p	M	NIR	RED	ALL	#FLOPs(G)
2	3	47.98	50.23	49.13	28.34
4	3	48.03	50.29	49.18	28.34
8	3	48.08	50.43	49.27	28.34
2	6	48.33	50.57	49.47	38.41
4	6	48.38	50.63	49.53	38.41
8	6	48.40	50.64	49.54	38.41

p denotes the number of transformer attention heads. M denotes the depth of the transformer. The definition of #FLOPs(G) is the same as that in Table II.

(images). The experiment sets the number of input images to 7.

g) *RAMS*: [30] Based on a large number of hand-designed attention blocks, it is the current state-of-the-art method of MISR on the PROBA-V Kelvin dataset. The experiment set the number of input images to 9. *RAMS* $_{+20}$ adopts the temporal self-ensemble, i.e., the input image sequence is shuffled 20 times randomly, and the mean image is taken under the premise of reducing the inference speed. We record the highest results of *RAMS* and *RAMS* $_{+20}$ in Table I.

Table I shows that TR-MISR has achieved the highest cPSNR and cSSIM scores on the NIR, RED, and ALL bands, which

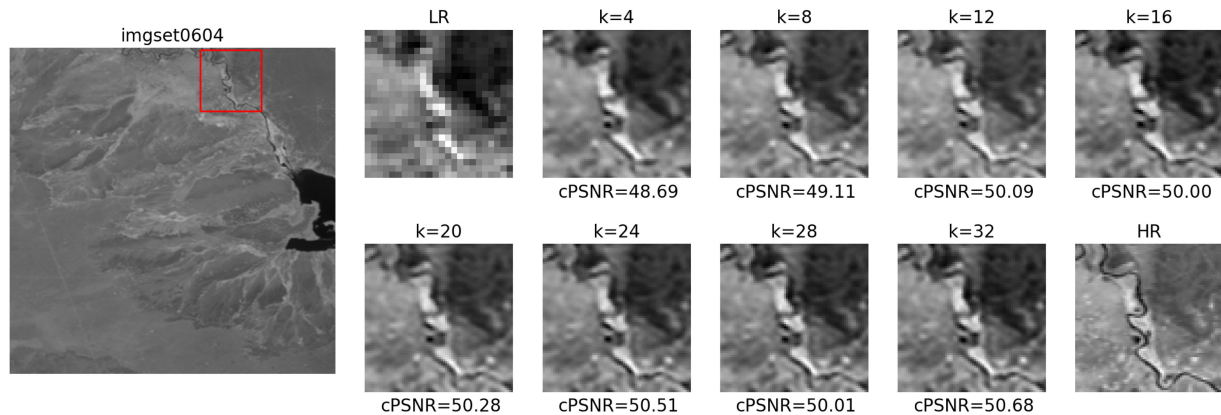


Fig. 8. Effect of different input images k on the imgset0604 scene of the NIR band. The reconstruction outcome achieves the best when $k = 32$.

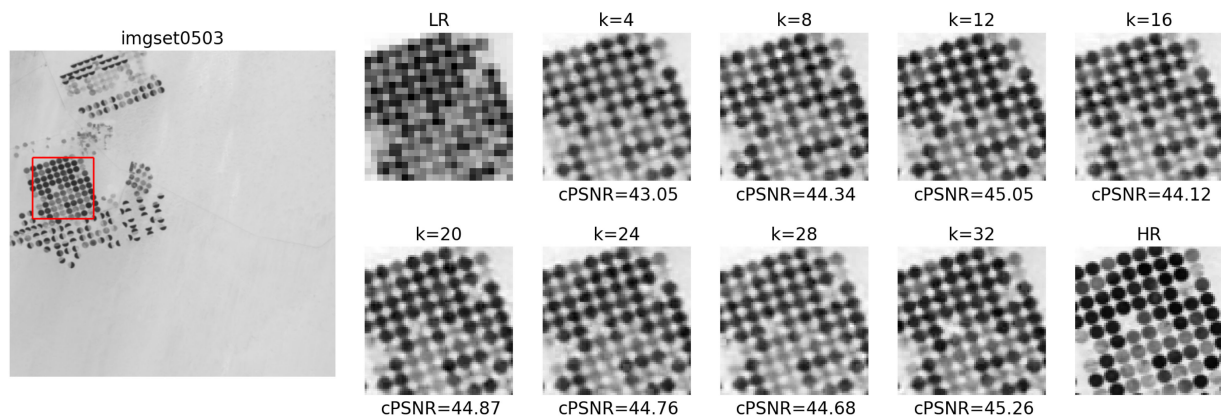


Fig. 9. Effect of different input images k on the imgset0503 scene of the RED band. The reconstruction outcome achieves the best when $k = 32$.

TABLE IV
CPSNR PERFORMANCE OF DIFFERENT NUMBERS OF INPUT
IMAGES k ON THE VALIDATION SET

k	NIR	RED	ALL	#FLOPs(G)
4	46.82	49.32	48.10	7.07
8	47.61	50.10	48.88	13.34
12	48.21	50.47	49.36	19.61
16	48.25	50.46	49.37	25.87
20	48.37	50.59	49.50	32.14
24	48.40	50.64	49.54	38.41
28	48.39	50.60	49.52	44.67
32	48.43	50.63	49.55	50.94

The definition of #FLOPs(G) is the same as that in Table II.

demonstrates the superiority of our method. Fig. 4 reveals that TR-MISR has an advantage over the baseline in 98.3% RED band and 97.1% NIR band.

E. Analysis of Network Settings

As shown in Fig. 3, TR-MISR supports predefining the model size to make a tradeoff between speed and accuracy. We fix the transformer settings and observe the performance of the encoder under different hyperparameters. There are two options for the encoder setting: The first option is the number of residual blocks

N , which determines the size of the receptive field. The second option is whether to use the reference image LR_{ref} as the implicit coregistration information of $\{LR_i\}_{i=1}^k$ during encoding.

We specify two fusion modules of different sizes: Transformer and small-transformer. The hyperparameter (p, M) of the transformer is $(8, 6)$ and the small-transformer is $(4, 3)$. We set the same random initialization and adopt 24 images as input. After using the same learning strategy mentioned in Section IV-C2, we get Table II. As reported in Table II, we also count floating-point operations #FLOPs (unit: Giga) and parameters #Params (unit: Million), representing the computation amount and the parameter amount of the framework, respectively.

In summary, we come to three points: First, the transformer with a large model capacity brings better results than small-transformer. Second, implicit image coregistration is necessary. Third, a larger receptive field is not necessarily better, i.e., the semantic information of high-level layers is not always effective in low-level tasks such as image super-resolution.

Next, under the condition of LR_{ref} and $N = 2$, we discuss the effect caused by different transformer hyperparameters (p, M) . Table III presents the results. The Transformer with more heads and deeper layers results in a better feature fusion capability but greater computational complexity.

TABLE V
CPSNR PERFORMANCE OF THE FUSION MODULES ON DIFFERENT VALIDATION SETS

val set	GAP			convGRU			convLSTM			recursion		
	NIR	RED	ALL	NIR	RED	ALL	NIR	RED	ALL	NIR	RED	ALL
std	47.24	49.58	48.43	47.48	49.74	48.63	47.45	49.84	48.67	47.03	49.37	48.22
w/o reg	46.96	49.25	48.13	47.20	49.45	48.34	47.20	49.56	48.41	46.15	49.16	48.03
+ 1	46.69	48.90	47.81	44.83	46.45	45.66	45.43	47.94	46.71	40.45	44.74	42.63
+ 2	46.26	48.52	47.41	44.04	46.01	45.04	44.22	46.81	45.54	38.61	43.48	41.09
+ 3	45.72	48.05	46.91	43.45	45.62	44.56	43.23	45.87	44.57	37.53	42.55	40.08

val set	3d conv			3d conv + attention			self-attention(ours)			Transformer(ours)		
	NIR	RED	ALL	NIR	RED	ALL	NIR	RED	ALL	NIR	RED	ALL
std	47.40	49.71	48.58	47.79	49.80	48.80	48.46	50.65	49.57	48.54	50.67	49.62
w/o reg	47.15	49.39	48.29	47.58	49.58	48.60	48.23	50.45	49.37	48.29	50.48	49.40
+ 1	39.45	41.93	40.71	47.10	49.17	48.15	48.15	50.36	49.28	48.34	50.50	49.44
+ 2	38.57	40.60	39.60	46.60	48.86	47.75	47.96	50.25	49.12	48.12	50.36	49.26
+ 3	37.57	39.37	38.51	45.86	48.19	47.04	47.68	50.09	48.91	47.72	50.13	48.95

“std” represents the standard validation set. “w/o reg” represents without image registration. “+ n” indicates the number of noise images mixed into each scene. The bold entities represent the highest cPSNR results for different validation sets.

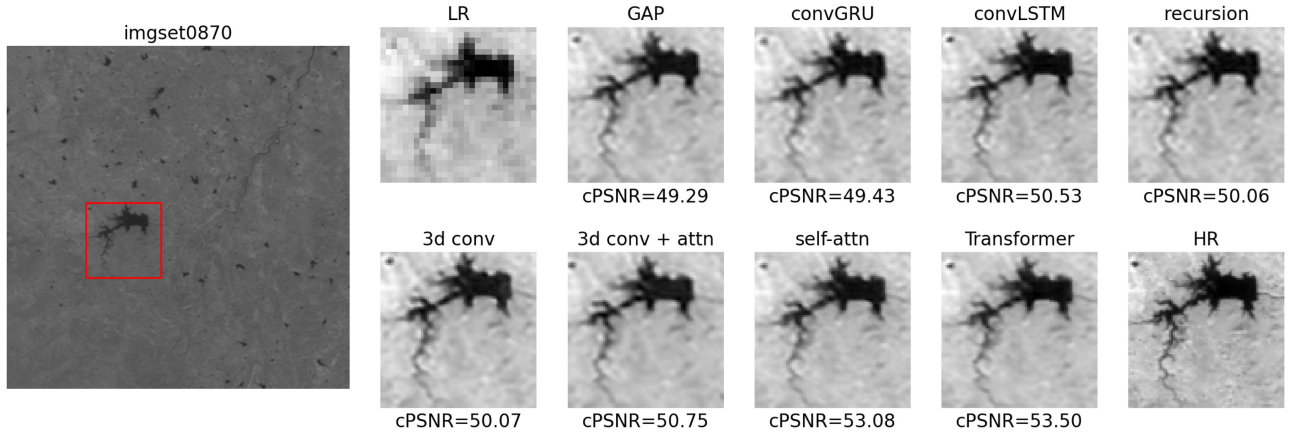


Fig. 10. Comparison between different fusion modules on the imgset0870 scene of the NIR band in the standard validation dataset.

TABLE VI
COMPARISON OF #FLOPs(G) AND #PARAMS(M) FOR THE FRAMEWORKS
BASED ON DIFFERENT FUSION MODULES

Fusion modules	#FLOPs(G)	#Params(M)
GAP	0.761	0.186
convGRU	2.575	0.629
convLSTM	3.179	0.777
recursion	2.225	0.555
3d conv	2.658	1.661
3d conv + attention	4.270	1.883
self-attention(ours)	1.177	0.285
Transformer(ours)	1.592	0.385

#FLOPs(G) denotes the average computation of a single image in one batch.

F. Analysis of Numbers of Input Images

The core assumption of MISR is that multiple images contain rich details. However, whether more images have a better reconstruction outcome has caused some discussion and controversy. In practice, researchers often select the clearest k images in each scene and feed them into the network, ignoring the remaining

unclear images. On the one hand, more images include more details while introducing more noise. Unclear images contain more noise and less information, which may have a bad effect on the generated images [26]. On the other hand, more input images will increase the pressure on the fusion module, because it is hard to extract long-distance relationships effectively. Influenced by the structures such as 3-D convolution, most MISR frameworks based on deep learning limit the number of input images to obtain a robust result. Specifically, the reconstruction outcome of HighRes-Net [24] achieves the best when the number of input images $k = 16$. The reconstruction outcome of DeepSUM [26] may worsen when the number of images $k > 9$, so DeepSUM sets k to 9. RAMS [30] also follows this rule. However, our TR-MISR allows any number of images to be input at one time, and more images do not cause performance degradation. As illustrated in Fig. 7, we discuss the cPSNR performance of TR-MISR under different numbers of input images k .

The results obtained by TR-MISR under the condition of $k = 24$ are second only to $k = 32$; see Figs. 7–9 and Table IV. In general, the more input images, the better the reconstruction outcome of TR-MISR. We hold that only by using more effective

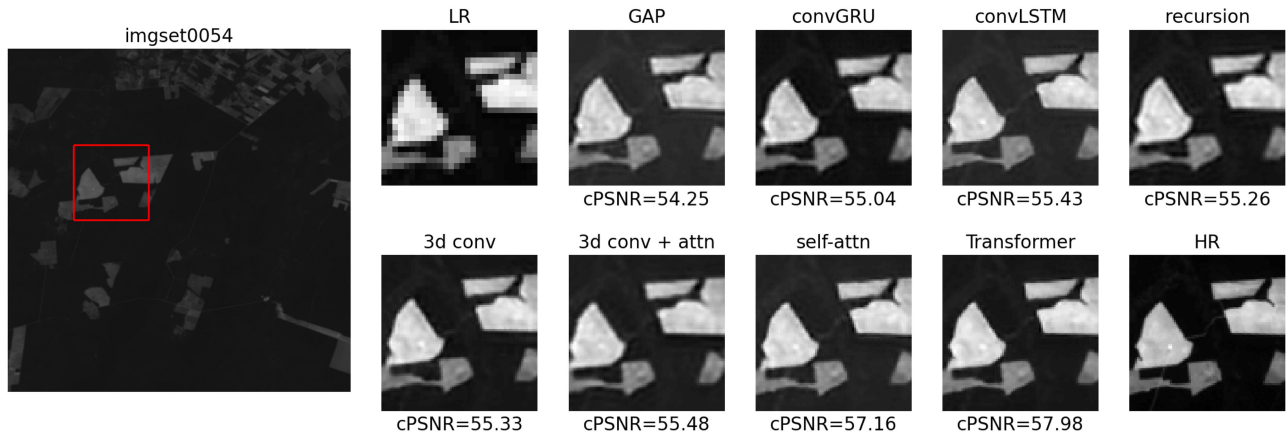


Fig. 11. Comparison between different fusion modules on the imgset0870 scene of the NIR band in the standard validation dataset.

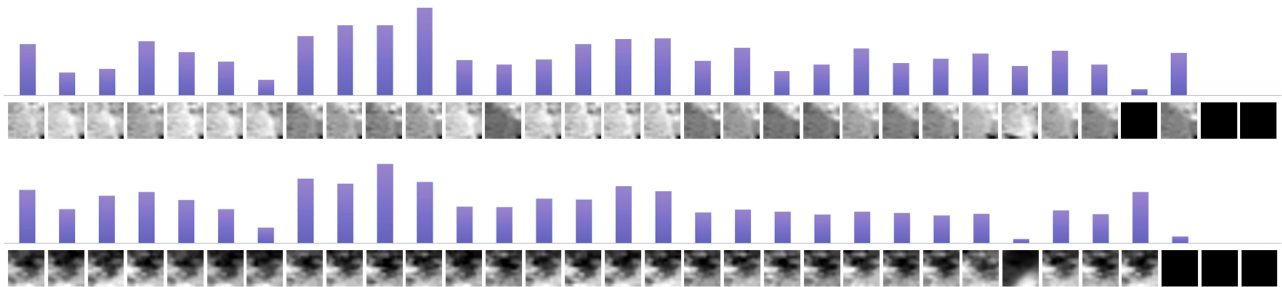


Fig. 12. Attention visualization of two areas on the imgset0252 scene. The histogram indicates the intensity of attention to different image patches during the fusion process.

image information can the upper limit of MISR be increased. As k becomes large, more unclear images are mixed into each scene, and the reconstruction outcome of TR-MISR tends to be better. There are two main reasons for this result: 1) The Transformer-based fusion module performs attention to multiple images dynamically, which makes itself more robust than 3-D convolution with fixed weights. 2) The high adaptation between the decoder and the fusion module guarantees the extracted information to be directly used to restore the details of high-resolution images, which minimizes the reconstruction deviation and avoids the overlap issue [78].

It is worth mentioning that when we set (k, N, p, M) to $(32, 2, 8, 6)$ and use the entire training set to train for 400 epochs on each band, TR-MISR reaches an \bar{R} score of 0.9300¹ on the testing set of the PROBA-V challenge and places at the top of the leaderboard.

G. Analysis of Fusion Modules

The superior performance and high adaptability to image sequences of TR-MISR are well illustrated in Sections IV-E and IV-F. Since image fusion is a crucial step in MISR, it is essential to compare the effect of different fusion modules on generated

images. In this section, we fix the encoder and decoder and compare the performance of the existing deep learning-based fusion modules on different validation sets. The results demonstrate the superiority of the transformer-based fusion module in MISR tasks. The fusion modules we implemented are as follows.

- 1) *GAP*: Global average pooling (GAP) is used to average the channels of feature maps.
- 2) *convGRU*: The convGRU [75] is employed as the fusion module, as in MISR-GRU [25].
- 3) *convLSTM*: The configuration of the convLSTM [84] is similar to the convGRU mentioned above.
- 4) *recursion*: The handcrafted iterative algorithm of HighRes-Net [24] is directly used.
- 5) *3-D conv*: A 3-layer 3-D convolution is adopted as the fusion module.
- 6) *3-D conv + attention*: Feature attention is introduced to the 3-D convolution by adding a FAB in front of each convolutional layer, which is inspired by RAMS [30].
- 7) *self-attention*: A fusion module based on multiheaded self-attention layers of the transformer is adopted.
- 8) *Transformer*: A transformer-based fusion module with multiheaded self-attention layers and feed-forward networks is used.

As for the evaluation, five different validation sets are used to verify the robustness of different fusion modules. “std” denotes

¹Online. [Available]: <https://kelvins.esa.int/proba-v-super-resolution/leaderboard/post-mortem-leaderboard/>

the standard validation set. “w/o reg” represents the validation set without registration [80]. “+1,” “+2,” and “+3” indicate that 1, 2, or 3 noisy images are mixed into each scene, respectively. These noisy images are just Gaussian noise that does not contain any information. The effect of noise should be considered, although images with a lot of noise have been removed in the PROBA-V dataset. Table V records the cPSNR results of the eight fusion modules on the five validation sets. Examples with details are given in Figs. 10 and 11 for the NIR and RED bands, respectively.

From Table V, we can summarize some general conclusions: 1) Image registration [80] improves the quality of image fusion; 2) the average pooling of feature channels can lead to a positive result. 3) The antinoise ability of convGRU [75], convLSTM [84], recursion, and 3-D convolution is low due to the lack of attention mechanism; 4) the transformer-based module has a higher model capacity and can achieve better results than the self-attention-based module. Figs. 10 and 11 reveal that the high-resolution images generated by the transformer-based fusion module are less noisy.

In addition, we show the complexity of the frameworks based on different fusion modules in Table VI. The statistic #FLOPs(G) counts the amount of calculation required for a single input image. #Params(M) represents the parameter amount of the framework. Note that, the #Params(M) of a GAP-based framework is 0.186 M, while the GAP operation is parameter-free.

As reported in Table VI, compared with other fusion modules, both #FLOPs(G) and #Params(M) of the transformer are in the lower range. The complexity of the transformer is not high, because we only use the encoder part of the transformer to design the fusion module.

H. Attention Visualization

In this subsection, we show the effect of self-attention in the fusion process. Because the encoder involves a total of 6 convolutional layers, the size of the receptive field is 13×13 . As is shown in Fig. 12, we take two sets of patches on the imgset0252 scene of the RED band as an example to visualize the attention. First, we extract the feature vectors of selected patches. Then the attention of the learnable embedding vector to the other encoded vectors is output based on attention maps. The attention here is the average value obtained under different transformer heads.

Fig. 12 shows that TR-MISR assigns different attention to different image patches and the unclear patches are assigned less attention. In remote sensing, pixel fidelity is more meaningful than visual perception. Some patches may have a good visual perception, but the attention to them is at a low level due to the influence of the noncontent pixels or few details. The transformer-based fusion module can effectively refer to the features of different patches, which is more conducive to the restoration of high-resolution images.

V. CONCLUSION

This article proposes TR-MISR, a novel end-to-end framework that addresses the problems of poor adaptability and

low data utilization in MISR tasks. The self-attention mechanism of transformers gives us much inspiration. Our proposed framework mainly includes three parts: An encoder, a fusion module, and a decoder. TR-MISR places great emphasis on the feature fusion capability of the fusion module. Specifically, the transformer-based fusion module assigns dynamic attention to different image patches. It has the permutation invariance property, which is especially suitable for the MISR datasets in remote sensing. Our experiments discuss the influence of the model size and the number of input images on the results and compare TR-MISR with other existing MISR methods. Then we analyze the superiority of the transformer-based fusion module and show the importance of self-attention.

TR-MISR has achieved the state of the art on the PROBA-V Kelvin dataset. More importantly, it has alleviated the limitations of transformers in low-level computer vision tasks. The original transformer destroys the two-dimensional structure of images or features and relies on massive data to surpass CNN that has inherent structural advantages. Our transformer avoids learning the spatial relations additionally, notably reducing the reliance on data. It starts with the same patches of different images, giving dynamic attention and integrating them through an additional learnable embedding vector. After decoding, our Transformer obtains the image patch that is corresponding to the high-resolution image. The ablation experiment of fusion modules shows that the Transformer, especially the self-attention mechanism, is of great help to the improvement of image fusion in MISR.

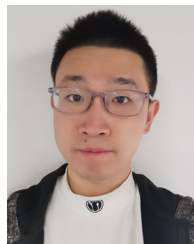
We intend to improve this work from the following aspects: 1) Research on a more effective encoder for registered/unregistered images; 2) removal of redundant attention and reduction of transformer complexity; and 3) further improvement of the results on more public datasets in the future.

REFERENCES

- [1] H. Greenspan, “Super-resolution in medical imaging,” *Comput. J.*, vol. 52, no. 1, pp. 43–63, 2009.
- [2] J. Jiang, J. Ma, C. Chen, X. Jiang, and Z. Wang, “Noise robust face image super-resolution through smooth sparse representation,” *IEEE Trans. Cybern.*, vol. 47, no. 11, pp. 3991–4002, Nov. 2017.
- [3] J. Jiang, C. Chen, J. Ma, Z. Wang, Z. Wang, and R. Hu, “SRLSP: A face image super-resolution algorithm using smooth regression with local structure prior,” *IEEE Trans. Multimedia*, vol. 19, no. 1, pp. 27–40, Jan. 2017.
- [4] A. Kappeler, S. Yoo, Q. Dai, and A. K. Katsaggelos, “Video super-resolution with convolutional neural networks,” *IEEE Trans. Comput. Imag.*, vol. 2, no. 2, pp. 109–122, Jun. 2016.
- [5] 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.*, 2016, pp. 1874–1883.
- [6] X. X. Zhu and R. Bamler, “Super-resolution power and robustness of compressive sensing for spectral estimation with application to spaceborne tomographic SAR,” *IEEE Trans. Geosci. Remote Sens.*, vol. 50, no. 1, pp. 247–258, Jan. 2011.
- [7] Z. Pan *et al.*, “Super-resolution based on compressive sensing and structural self-similarity for remote sensing images,” *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 9, pp. 4864–4876, Sep. 2013.
- [8] J. M. Haut, R. Fernandez-Beltran, M. E. Paoletti, J. Plaza, A. Plaza, and F. Pla, “A new deep generative network for unsupervised remote sensing single-image super-resolution,” *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 11, pp. 6792–6810, Nov. 2018.

- [9] P. Wang, L. Wang, H. Leung, and G. Zhang, "Super-resolution mapping based on spatial-spectral correlation for spectral imagery," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 3, pp. 2556–2568, Mar. 2021.
- [10] F. Li, L. Xin, Y. Guo, D. Gao, X. Kong, and X. Jia, "Super-resolution for GaoFen-4 remote sensing images," *IEEE Geosci. Remote Sens. Lett.*, vol. 15, no. 1, pp. 28–32, Jan. 2017.
- [11] X. Li *et al.*, "Spatial-temporal super-resolution land cover mapping with a local spatial-temporal dependence model," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 7, pp. 4951–4966, Jul. 2019.
- [12] 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.
- [13] H.-j. Xu, X.-p. Wang, and T.-b. Yang, "Trend shifts in satellite-derived vegetation growth in Central Eurasia, 1982–2013," *Sci. Total Environ.*, vol. 579, pp. 1658–1674, 2017.
- [14] R. Fernandez-Beltran, P. Latorre-Carmona, and F. Pla, "Single-frame super-resolution in remote sensing: A practical overview," *Int. J. Remote Sens.*, vol. 38, no. 1, pp. 314–354, Jan. 2017.
- [15] H. Demirel and G. Anbarjafari, "Satellite image resolution enhancement using complex wavelet transform," *IEEE Geosci. Remote Sens. Lett.*, vol. 7, no. 1, pp. 123–126, Jan. 2010.
- [16] H. Demirel and G. Anbarjafari, "Discrete wavelet transform-based satellite image resolution enhancement," *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 6, pp. 1997–2004, Jun. 2011.
- [17] S. Yang, M. Wang, Y. Chen, and Y. Sun, "Single-image super-resolution reconstruction via learned geometric dictionaries and clustered sparse coding," *IEEE Trans. Image Process.*, vol. 21, no. 9, pp. 4016–4028, Sep. 2012.
- [18] M. Märtens, D. Izzo, A. Krzic, and D. Cox, "Super-resolution of PROBA-V images using convolutional neural networks," *Astrodynamics*, vol. 3, no. 4, pp. 387–402, 2019.
- [19] X. Liu, C. Deng, J. Chanussot, D. Hong, and B. Zhao, "StfNet: A two-stream convolutional neural network for spatiotemporal image fusion," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 9, pp. 6552–6564, Sep. 2019.
- [20] S. Mei, R. Jiang, X. Li, and Q. Du, "Spatial and spectral joint super-resolution using convolutional neural network," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 7, pp. 4590–4603, Jul. 2020.
- [21] C. Ledig *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 4681–4690.
- [22] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, and Y. Fu, "Residual dense network for image super-resolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 2472–2481.
- [23] V. Mnih, N. Heess, A. Graves, and K. Kavukcuoglu, "Recurrent models of visual attention," 2014, *arXiv:1406.6247*.
- [24] M. Deudon *et al.*, "Highres-Net: Multi-frame super-resolution by recursive fusion," 2019, *arXiv:2002.06460*.
- [25] M. R. Arefin *et al.*, "Multi-image super-resolution for remote sensing using deep recurrent networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops*, 2020, pp. 206–207.
- [26] A. B. Molini, D. Valsesia, G. Fracastoro, and E. Magli, "DeepSum: Deep neural network for super-resolution of unregistered multitemporal images," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 5, pp. 3644–3656, Dec. 2019.
- [27] A. B. Molini, D. Valsesia, G. Fracastoro, and E. Magli, "DeepSum: Non-local deep neural network for super-resolution of unregistered multitemporal images," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2020, pp. 609–612.
- [28] M. Bajo, "Multi-frame super resolution of unregistered temporal images using WDSR nets," 2020. [Online]. Available: <https://doi.org/10.5281/zenodo.3733116>
- [29] F. Dorr, "Satellite image multi-frame super resolution using 3D wide-activation neural networks," *Remote Sens.*, vol. 12, no. 22, 2020, Art. no. 3812.
- [30] F. Salvetti, V. Mazzia, A. Khaliq, and M. Chiaberge, "Multi-image super resolution of remotely sensed images using residual attention deep neural networks," *Remote Sens.*, vol. 12, no. 14, 2020, Art. no. 2207.
- [31] 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.*, 2018, pp. 286–301.
- [32] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 3–19.
- [33] D. Meng, X. Peng, K. Wang, and Y. Qiao, "Frame attention networks for facial expression recognition in videos," in *Proc. IEEE Int. Conf. Image Process.*, 2019, pp. 3866–3870.
- [34] L. Weng, "Attention? attention!" lilianweng.github.io/lil-log, 2018. [Online]. Available: <http://lilianweng.github.io/lil-log/2018/06/24/attention-attention.html>
- [35] A. Vaswani *et al.*, "Attention is all you need," 2017, *arXiv:1706.03762*.
- [36] S. Khan, M. Naseer, M. Hayat, S. W. Zamir, F. S. Khan, and M. Shah, "Transformers in vision: A survey," 2021, *arXiv:2101.01169*.
- [37] K. Han *et al.*, "A survey on visual transformer," 2020, *arXiv:2012.12556*.
- [38] A. Dosovitskiy *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," 2020, *arXiv:2010.11929*.
- [39] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou, "Training data-efficient image transformers & distillation through attention," 2020, *arXiv:2012.12877*.
- [40] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 213–229.
- [41] S. Zheng *et al.*, "Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers," 2020, *arXiv:2012.15840*.
- [42] H. Chen *et al.*, "Pre-trained image processing transformer," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 12299–12310.
- [43] M. Kumar, D. Weissenborn, and N. Kalchbrenner, "Colorization transformer," 2021, *arXiv:2102.04432*.
- [44] D. Hong *et al.*, "SpectralFormer: Rethinking hyperspectral image classification with transformers," *IEEE Trans. Geosci. Remote Sens.*, 2021.
- [45] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable DETR: Deformable transformers for end-to-end object detection," 2020, *arXiv:2010.04159*.
- [46] Z. Xu, W. Zhang, T. Zhang, Z. Yang, and J. Li, "Efficient transformer for remote sensing image segmentation," *Remote Sens.*, vol. 13, no. 18, 2021, p. 3585.
- [47] F. Yang, H. Yang, J. Fu, H. Lu, and B. Guo, "Learning texture transformer network for image super-resolution," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 5791–5800.
- [48] J. Cao, Y. Li, K. Zhang, and L. Van Gool, "Video super-resolution transformer," 2021, *arXiv:2106.06847*.
- [49] R. Keys, "Cubic convolution interpolation for digital image processing," *IEEE Trans. Acoust. Speech Signal Process.*, vol. 29, no. 6, pp. 1153–1160, Dec. 1981.
- [50] S. Dai, M. Han, W. Xu, Y. Wu, Y. Gong, and A. K. Katsaggelos, "SoftCuts: A soft edge smoothness prior for color image super-resolution," *IEEE Trans. Image Process.*, vol. 18, no. 5, pp. 969–981, May 2009.
- [51] W. T. Freeman, T. R. Jones, and E. C. Pasztor, "Example-based super-resolution," *IEEE Comput. Graph. Appl.*, vol. 22, no. 2, pp. 56–65, Mar./Apr. 2002.
- [52] J. Yang, J. Wright, T. S. Huang, and Y. Ma, "Image super-resolution via sparse representation," *IEEE Trans. Image Process.*, vol. 19, no. 11, pp. 2861–2873, Nov. 2010.
- [53] 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.
- [54] Y. Tai, J. Yang, and X. Liu, "Image super-resolution via deep recursive residual network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 3147–3155.
- [55] J. Kim, J. K. Lee, and K. M. Lee, "Deeply-recursive convolutional network for image super-resolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 1637–1645.
- [56] L. Liebel and M. Körner, "Single-image super resolution for multispectral remote sensing data using convolutional neural networks," *ISPRS-Int. Arch. Photogrammetry, Remote Sens. Spatial Inf. Sci.*, vol. 41, pp. 883–890, 2016.
- [57] C. Tuna, G. Unal, and E. Sertel, "Single-frame super resolution of remote-sensing images by convolutional neural networks," *Int. J. Remote Sens.*, vol. 39, no. 8, pp. 2463–2479, 2018.
- [58] C. Lanaras, J. Bioucas-Dias, S. Galliani, E. Baltsavias, and K. Schindler, "Super-resolution of sentinel-2 images: Learning a globally applicable deep neural network," *Int. J. Photogrammetry Remote Sens.*, vol. 146, pp. 305–319, 2018.
- [59] D. Pouliot, R. Latifovic, J. Pasher, and J. Duffe, "Landsat super-resolution enhancement using convolution neural networks and sentinel-2 for training," *Remote Sens.*, vol. 10, no. 3, 2018, p. 394.
- [60] L. S. Romero, J. Marcello, and V. Vilaplana, "Super-resolution of sentinel-2 imagery using generative adversarial networks," *Remote Sens.*, vol. 12, no. 15, 2020, p. 2424.
- [61] H. Liu, Z. Ruan, P. Zhao, F. Shang, L. Yang, and Y. Liu, "Video super resolution based on deep learning: A comprehensive survey," 2020, *arXiv:2007.12928*.

- [62] A. Ranjan and M. J. Black, "Optical flow estimation using a spatial pyramid network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 4161–4170.
- [63] J. Xu, R. Ranftl, and V. Koltun, "Accurate optical flow via direct cost volume processing," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 1289–1297.
- [64] S. Niklaus, L. Mai, and F. Liu, "Video frame interpolation via adaptive convolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 670–679.
- [65] 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, 2018, p. 1194.
- [66] 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.
- [67] L. Gao, D. Hong, J. Yao, B. Zhang, P. Gamba, and J. Chanussot, "Spectral superresolution of multispectral imagery with joint sparse and low-rank learning," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 3, pp. 2269–2280, Mar. 2020.
- [68] X. Cao, X. Fu, C. Xu, and D. Meng, "Deep spatial-spectral global reasoning network for hyperspectral image denoising," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, 2021, Art. no. 5504714.
- [69] R. Dian, S. Li, and X. Kang, "Regularizing hyperspectral and multispectral image fusion by CNN denoiser," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 3, pp. 1124–1135, Mar. 2020.
- [70] L. Alparone, L. Wald, J. Chanussot, C. Thomas, P. Gamba, and L. M. Bruce, "Comparison of pansharpening algorithms: Outcome of the 2006 GRS-S data-fusion contest," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, no. 10, pp. 3012–3021, Oct. 2007.
- [71] R. Dian, S. Li, A. Guo, and L. Fang, "Deep hyperspectral image sharpening," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 11, pp. 5345–5355, Nov. 2018.
- [72] A. Guo, R. Dian, and S. Li, "Unsupervised blur kernel learning for pansharpening," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, 2020, pp. 633–636.
- [73] M. Ben-Ezra, A. Zomet, and S. K. Nayar, "Video super-resolution using controlled subpixel detector shifts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 6, pp. 977–987, Jun. 2005.
- [74] B. Wronski *et al.*, "Handheld multi-frame super-resolution," *ACM Trans. Graph.*, vol. 38, no. 4, pp. 1–18, 2019.
- [75] N. Ballas, L. Yao, C. Pal, and A. Courville, "Delving deeper into convolutional networks for learning video representations," 2015, *arXiv:1511.06432*.
- [76] S. Y. Kim, J. Lim, T. Na, and M. Kim, "3DSRnet: Video super-resolution using 3-D convolutional neural networks," 2018, *arXiv:1812.09079*.
- [77] E. H. Sanchez, M. Serrurier, and M. Ortner, "Learning disentangled representations of satellite image time series," in *Proc. Joint Eur. Conf. Mach. Learn. Knowl. Discov. Databases*, 2019, pp. 306–321.
- [78] A. Odena, V. Dumoulin, and C. Olah, "Deconvolution and checkerboard artifacts," *Distill*, vol. 1, no. 10, p. e3, 2016.
- [79] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.
- [80] M. Guizar-Sicairos, S. T. Thurman, and J. R. Fienup, "Efficient subpixel image registration algorithms," *Opt. Lett.*, vol. 33, no. 2, pp. 156–158, 2008.
- [81] S. Farsiu, M. D. Robinson, M. Elad, and P. Milanfar, "Fast and robust multiframe super resolution," *IEEE Trans. Image Process.*, vol. 13, no. 10, pp. 1327–1344, Oct. 2004.
- [82] M. Irani and S. Peleg, "Improving resolution by image registration," *CVGIP: Graphical Models Image Process.*, vol. 53, no. 3, pp. 231–239, 1991.
- [83] 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 Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 3224–3232.
- [84] S. Xingjian, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-C. Woo, "Convolutional LSTM network: A machine learning approach for precipitation nowcasting," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 802–810.



Tai An received the B.S. degree in information engineering from Xi'an Jiaotong University, Xi'an, China, in 2018. He is currently working toward the Ph.D. degree in pattern recognition and intelligent system with the Institute of Automation, Chinese Academy of Sciences and University of Chinese Academy of Sciences, Beijing, China.

His research interests include remote sensing, pattern recognition, computer vision, especially on super-resolution.



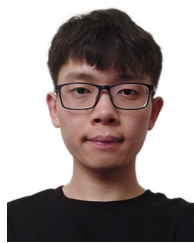
Xin Zhang received the B.S. degree in information and computing science from Beijing University of Technology, Beijing, China, in 2014, the M.S. degree in applied mathematics from Beijing University of Technology, Beijing, China, in 2018. He is currently working toward the Ph.D. degree in pattern recognition and intelligent system with the Institute of Automation, Chinese Academy of Sciences and University of Chinese Academy of Sciences, Beijing, China.

His research interests are object detection, computer vision, pattern recognition, and remote sensing.



Chunlei Huo (Member, IEEE) received the B.S. degree in applied mathematics from Hebei Normal University, Shijiazhuang, China, in 1999, the M.S. degree in applied mathematics from Xidian University, Xi'an, China, in 2002, and the Ph.D. degree in pattern recognition and intelligent system from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2009.

He is currently a Professor with the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences. His research interests include remote sensing image processing, computer vision, pattern recognition.



Bin Xue received the B.S. degree in automation from Northwestern Polytechnical University, Xi'an, China, in 2018. He is currently working toward the Ph.D. degree in pattern recognition and intelligent system with the Institute of Automation, Chinese Academy of Sciences and University of Chinese Academy of Sciences, Beijing, China. His research interests are computer vision and pattern recognition.



Lingfeng Wang (Member, IEEE) received the B.S. degree in computer science from Wuhan University, Wuhan, China, in 2007. He received the Ph.D. degree in pattern recognition and intelligent system from Institute of Automation, Chinese Academy of Sciences, in 2013.

He is currently an Associate Professor with the National Laboratory of Pattern Recognition of Institute of Automation, Chinese Academy of Sciences. His research interests include computer vision and image processing.



Chunhong Pan (Member, IEEE) received the B.S. degree in automatic control from Tsinghua University, Beijing, China, in 1987, the M.S. degree from the Shanghai Institute of Optics and Fine Mechanics, Chinese Academy of Sciences, Shanghai, China, in 1990, and the Ph.D. degree in pattern recognition and intelligent system from the Institute of Automation, Chinese Academy of Sciences, Beijing, in 2000.

He is currently a Professor with the National Laboratory of Pattern Recognition of Institute of Automation, Chinese Academy of Sciences. His research interests include computer vision, image processing, computer graphics, and remote sensing.