Space-time super-resolution for satellite video: A joint framework based on multi-scale spatial-temporal transformer

Yi Xiao, Qiangqiang Yuan, Jiang He, Qiang Zhang, Jing Sun, Xin Su, Jialian Wu, Liangpei Zhang

Abstract

Satellite video is an emerging type of earth observation tool, which has attracted increasing attention because of its application in dynamic analysis. However, most studies only focus on improving the spatial resolution of satellite video imagery. In contrast, few works are committed to enhancing the temporal resolution, and the joint spatial-temporal improvement is even less. The joint spatial-temporal enhancement can not only produce high-resolution imagery for subsequent applications, but also provide the potentials of clear motion dynamics for extreme events observation. In this paper, we propose a joint framework to enhance the spatial and temporal resolution of satellite video simultaneously. Firstly, to alleviate the problem of scale variation and scarce motion in satellite video, we design a feature interpolation module that deeply couples optical flow and multi-scale deformable convolution to predict unknown frames. Deformable convolution can adaptively learn the multi-scale motion information and profoundly complement optical flow information. Secondly, a multi-scale spatial-temporal transformer is proposed to aggregate the contextual information in long-time series video frames effectively. Since multi-scale patches are embedded in multiple heads for spatial-temporal self-attention calculation, we can comprehensively exploit multi-scale details in all frames. Extensive experiments on the Jilin-1 satellite video demonstrate that our model is superior to the existing methods. The source code is available at https://github.com/XY-boy.

1. Introduction

Nowadays, several satellites with video sensors are launched into space, which dramatically improves the temporal resolution of regional observations and provides a novel data application scenario for dynamic event monitoring. In fields like vehicle tracking (Feng et al., 2021) and flood disaster monitoring (de Alwis Pitts and So, 2017), the instantaneous and continuity of video data are incomparably superior to traditional static images. However, on the one hand, the spatial resolution of satellite video is not comparable to traditional satellite imagery. In remote imaging procedure, the spatial resolution will suffer from degradation like atmospheric scattering and extreme weather (He et al., 2021; Lanaras et al., 2018). These factors have caused a bottleneck in the improvement of spatial resolution. On the other hand, the temporal resolution of satellite video is hindered by the limited bandwidth of the sensor. Within these limitations, high quality satellite video data is urgently needed in remote sensing field (He et al., 2022; Zhang et al., 2020, 2021a). A high spatial resolution remote sensing image with rich detailed information is of great significance for subsequent application (Chen et al., 2021), such as semantic segmentation (Li et al., 2022; Ma et al., 2021; Peng et al., 2021), environmental parameters estimation (Wang et al., 2021, 2022; Zhang et al., 2021b), and land cover classification (Abid et al., 2021; Amato et al., 2021). A slow-motion video with high temporal resolution provides clear motion dynamics, which is beneficial for us to analyze the evolution of extreme and transient events (Vandal and Nemani, 2021). Therefore, it is worthwhile to improve the spatial and temporal resolution of satellite video for its broad application prospects.

Space-time video super-resolution (STVSR) aims to improve the spatial and temporal resolution of the video simultaneously (Kang et al.,...
explicit spatial image super-resolution (SISR) and video super-resolution (VSR). It is still in its infancy. At present, related work for video super resolution can be divided into three categories:

(1) **Space super-resolution:** S-SR is mainly divided into single image super-resolution (SISR) and video super-resolution (VSR). SISR recovers a high-resolution (HR) image from its corresponding low-resolution (LR) image (Shen et al., 2020). The pioneering SISR method based on deep learning is proposed by Dong et al. (2015), which uses convolutional neural network to fit the nonlinear mapping between LR space and HR space. Subsequently, Lim et al. (2017) proposed an advanced deep super-resolution network (EDSR) to introduce both residual learning and sub-pixel convolution. Zhang et al. (2018) employed the channel attention mechanism into SISR and proposed a deep residual learning and sub-pixel convolution. Zhang et al. (2018) employed deformable convolution (DConv) to achieve motion estimation and pixel alignment. Currently, the research on satellite video imagery. Until now, the study on STVSR for satellite video is still in its infancy.

(2) **Time super-resolution:** T-SR requires predicting frames that do not exist between two original frames that already exist. The mainstream methods can be broadly divided into two kinds: optical flow-based and kernel-based. The optical flow-based method (Bao et al., 2019a,b; Jiang et al., 2018; Niklaus and Liu, 2018; Xu et al., 2019) usually combines the optical flow maps linearly to estimate the latent optical flows which need to be synthesized, and finally the original frames are warped by the predicted optical flows to estimate the intermediate target frame. Recently, Sim et al. (2021) proposed a network that adopts a recursive multi-scale shared structure to learn bidirectional optical flow between two input frames and bidirectional optical flow between target and input frames. Kernel-based methods use adaptive convolution kernel to realize motion estimation and pixel synthesis. To handle complex motion, Lee et al. (2020) designed a 2D deformable spatial-adaptive scheme to break the limits of the fixed grid shape of regular convolution kernel.

(3) **Space-time super-resolution:** ST-SR was first proposed by Shechtman et al. (2005). The purpose is to recover an HR and high frame rate (HFR) video from an LR and low-frame-rate (LFR) video. Different from separate T-SR and S-SR, ST-SR requires super-resolving both spatial and temporal dimensions simultaneously. ST-SR is a highly ill-posed problem since the requirement to predict non-existent frame pixels and HR frame pixels. Previous traditional methods (Mudenagudi et al., 2010; Shahar et al., 2011; Takeda et al., 2020) use hand-crafted priors to constrain solution space, which is complicated to constrain. Benefit from deep learning, the data-driven ST-SR method has shown far superior performance to traditional methods. Haris et al. (2020) followed a three-step ST-SR strategy to reconstruct all LR frames and HR frames. Xiang et al. (2020) employed DConv to achieve the alignment between the original input frames, then aggregated two aligned features to estimate the target features, and finally captured the contextual information between the frames through the Bidirectional Deformable Long Short-Term Memory (BD-LSTM). Shi et al. (2021) proposed an unconstrained STVSR framework that realizes frame interpolation at any temporal location through the adjustable optical flow.

These one-stage methods rely on either optical flow or DConv to synthesize missing frames, which is not sophisticated enough for more challenging satellite video. Besides, these computationally complex design (LSTM or dense structure) in satellite video may introduce interference information while being inefficient (Vaswani et al., 2017). The effective spatial–temporal information fusion is critical to ST-SR (Chen et al., 2022; Li et al., 2020, 2021), which demands us to devise valid frameworks to aggregate spatial–temporal information. Hence, we propose a lightweight and end-to-end framework based on multi-scale spatial–temporal transformer. An optical flow and multi-scale deformable convolutional deeply coupled module is designed to realize the prediction of non-existent frames. To our best knowledge, this paper is the first to study the joint enhancement of the spatial and temporal resolution for satellite video.

2. Methodology

2.1. Problem formulation

Firstly, we give the formula definition of STVSR. Given a LR and LFR video frame sequence \( I_{LR} = \{ F_{2i-1}^{LR} \}_{i=1}^{N-1} \), the goal of STVSR is to generate their corresponding HR frames \( I_{HR} = \{ F_{2i-1}^{HR} \}_{i=1}^{N-1} \) as well as N missing HR frames \( \{ F_{2i}^{HR} \}_{i=1}^{N} \), and finally obtain a HR and HFR video sequence \( I_{HR} = \{ F_{2i-1}^{HR} \}_{i=1}^{N+1} \). As shown in Fig. 1, assuming that the network takes two existing original LR frames \( F_{LR}^{1} \) and \( F_{LR}^{2} \) as input, we need to predict the HR frame \( I_{HR}^{1} \) at the intermediate moment \( t \) while obtaining the HR frame \( I_{HR}^{t-1} \) and \( I_{HR}^{t+1} \) corresponding to the original LR frames.

Our framework is divided into the following parts: feature extraction, Flow-DConv deep-Coupled (FDC) feature interpolation, Multi-scale Spatial-Temporal Transformer (MSTT) and reconstruction module. First, the optical flow map \( F_{opt}^{1} \) and \( F_{opt}^{2} \) containing forward and backward motion information can be obtained through an optical flow estimation approach. Then, two LR features and two optical flow maps enter the FDC feature interpolation module to synthesize the missing intermediate frame features \( F_{syn}^{1} \). After that, we further explores the contextual information in all frames through our MSTT module, and finally we reconstruct the super-resolved results \( F_{HR}^{1}, I_{HR}^{1} \) and \( I_{HR}^{2} \). The details of each part will be explained below.

2.2. Feature extraction

For each LR frame \( F_{LR}^{2i-1} \in \mathbb{R}^{h' \times w' \times c} \), where \( h' \) and \( w' \) are the height and width of LR frames and \( c = 3 \) is the number of RGB channels, we utilize five residual blocks without batch normalization (BN) for feature extraction. Each residual block follows the structure of “Conv + ReLU + Conv”. Finally, we get an LR features \( F_{LR}^{2i-1} \in \mathbb{R}^{h' \times w' \times b_4} \). In this paper, we have \( N = 3 \), so after feature extraction we will obtain 4 LR features \( \{ F_{LR}^{1}, F_{LR}^{2}, F_{LR}^{3}, F_{LR}^{4} \} \).
2.3. Flow-DConv deep-Coupled feature interpolation

We believe that the motion information in satellite video learned by optical flow is insufficient since it naturally has a disadvantage in capturing the large motion (Xiang et al., 2020). Although deformable convolution has the advantage of adaptive learning, it is difficult to capture moving objects with various scales (Xiao et al., 2021). Our FDC feature interpolation (FI) module aims to make optical flow and deformable convolution mutually constrain each other by deep coupling manner, and introduce the adaptive learnability of deformable convolution to complement the information that optical flow ignores, so the latent motion information can be learned adaptively. The structure of FDC-FI is shown in Fig. 2.

(1) Optical flow estimation: Let the forward optical flow $f_{1\rightarrow3}$ warp LR feature $F_{LR3}$, the feature $F_1'$ estimated by the forward motion information can be obtained. Similarly, we get the feature $F_3'$ derived from the backward optical flow $f_{3\rightarrow1}$. This process can be formulated as:

$$F_1' = \text{warp}(F_{LR3}, f_{1\rightarrow3}),$$

$$F_3' = \text{warp}(F_{LR1}, f_{3\rightarrow1}).$$

(2) Deformable convolution estimation: To accurately exploit the motion information of moving objects with variable scales in satellite video, we adopt the multi-scale residual block (MSRB) proposed in our previous work Xiao et al. (2021) to learn sampling parameters $\Theta$ for deformable grids. This means the sampling parameters can be expressed as:

$$\Theta_1 = \text{MSRB}(\text{concat}(F_{LR3}, F_{LR1})).$$

$$\Theta_3 = \text{MSRB}(\text{concat}(F_{LR1}, F_{LR3})).$$

Where $\Theta_1$ represents the sampling parameter that encodes the forward motion information, and $\Theta_3$ is the sampling parameter that encodes the backward motion information. Now we can perform deformable convolution (DConv) (Zhu et al., 2019) under the guidance of sampling parameters to get the feature $F_3'$ estimated by the forward motion information and the feature $F_1'$ estimated by the backward motion information:

$$F_1' = \text{DConv}(F_{LR1}, \Theta_1),$$

$$F_3' = \text{DConv}(F_{LR3}, \Theta_3).$$

The deformable convolution means that the convolution sampling position has an additional offset, and the sampling grid is no longer a regular square grid. After deformable convolution, the value of position $p_0$ can be defined as:

$$F_i'(p_0) = \sum_{k=1}^{K} \omega_k F_{LRi}(p_0 + p_k + \Delta p_k).$$

Fig. 1. Network structure diagram. Input $N+1$ LR frames, our network can get $2N+1$ HR frames in an end-to-end manner. For the convenience of explanation, we set $N = 1$ here and $T = 2N+1 = 3$ is the length of all frames.

Fig. 2. The diagram of Flow-DConv Deep-Coupled Feature Interpolation module. The red features are derived from the forward motion information, and the green features are derived from the backward motion information; The channel separation operation divides the features of each frame into four heads; The lower right corner is a schematic diagram of feature unfold and fold.
Multi-scale Spatial-Temporal Transformer (MSTT) to fully explore the learned by convolution structure, our multi-scale information is mined through multi-scale receptive fields. Different from multi-scale information processing (NLP), but owing to its superior temporal representation capability, we devise a multi-scale information representation for remote sensing imagery. For this reason, we embed features into the multi-heads of the self-attention module to simultaneously represent multi-scale information using a single-scale transformer, the scale gap in remote sensing imagery makes it difficult to represent the multi-scale information in the satellite video. To be more specific:

(1) Encoder: It is necessary to embed LR features into the embedding space for attention calculation. To save memory overhead, we first use 3 × 3 convolution with a stride of 2 to achieve feature down-sampling, and increase the number of feature channels to 256. Denote the feature before encoding as $x = \{F_{j}^{S}, F_{j}^{R}, \ldots, F_{j}^{E}\}$, where $T$ is the number of frames and $F_{j}^{R} \in \mathbb{R}^{h \times w \times 64}$ for $j \in [1, T]$. The encoded features $e_{i} \in \text{Encoder}(F_{j}^{R})$ is taken from the results of optical flow and DConv respectively to achieve the complementation. Finally, we blend the coupling features and the final synthesis feature $F_{j}^{S}$.

$$F_{j}^{S} = \alpha \cdot f_{1}\left(F_{j}^{S}, F_{j}^{R}\right) + \beta \cdot f_{2}\left(F_{j}^{S}, F_{j}^{E}\right). \quad (8)$$

where $\alpha$ denotes the concatenation operation, $\beta$ represents the multi-heads of the self-attention module.

(2) Channel Split: The encoded features are matched to different heads after channel chunk operation:

$$\{e_{i1}, e_{i2}, e_{i3}, e_{i4}\} = \text{chunk}(e_{i}). \quad (9)$$

For this reason, we embed features into the multi-heads of the self-attention module to simultaneously represent multi-scale information using a single-scale transformer, the scale gap in remote sensing imagery makes it difficult to represent the multi-scale information in the satellite video. To be more specific:

(1) Encoder: It is necessary to embed LR features into the embedding space for attention calculation. To save memory overhead, we first use 3 × 3 convolution with a stride of 2 to achieve feature down-sampling, and increase the number of feature channels to 256. Denote the feature before encoding as $x = \{F_{j}^{S}, F_{j}^{R}, \ldots, F_{j}^{E}\}$, where $T$ is the number of frames and $F_{j}^{R} \in \mathbb{R}^{h \times w \times 64}$ for $j \in [1, T]$. The encoded features $e_{i} \in \text{Encoder}(F_{j}^{R})$ is taken from the results of optical flow and DConv respectively to achieve the complementation. Finally, we blend the coupling features and the final synthesis feature $F_{j}^{S}$.

$$F_{j}^{S} = \alpha \cdot f_{1}\left(F_{j}^{S}, F_{j}^{R}\right) + \beta \cdot f_{2}\left(F_{j}^{S}, F_{j}^{E}\right). \quad (8)$$

where $\alpha$ denotes the concatenation operation, $\beta$ represents the multi-heads of the self-attention module.

(2) Channel Split: The encoded features are matched to different heads after channel chunk operation:

$$\{e_{i1}, e_{i2}, e_{i3}, e_{i4}\} = \text{chunk}(e_{i}). \quad (9)$$

For this reason, we embed features into the multi-heads of the self-attention module to simultaneously represent multi-scale information using a single-scale transformer, the scale gap in remote sensing imagery makes it difficult to represent the multi-scale information in the satellite video. To be more specific:

(1) Encoder: It is necessary to embed LR features into the embedding space for attention calculation. To save memory overhead, we first use 3 × 3 convolution with a stride of 2 to achieve feature down-sampling, and increase the number of feature channels to 256. Denote the feature before encoding as $x = \{F_{j}^{S}, F_{j}^{R}, \ldots, F_{j}^{E}\}$, where $T$ is the number of frames and $F_{j}^{R} \in \mathbb{R}^{h \times w \times 64}$ for $j \in [1, T]$. The encoded features $e_{i} \in \text{Encoder}(F_{j}^{R})$ is taken from the results of optical flow and DConv respectively to achieve the complementation. Finally, we blend the coupling features and the final synthesis feature $F_{j}^{S}$.

$$F_{j}^{S} = \alpha \cdot f_{1}\left(F_{j}^{S}, F_{j}^{R}\right) + \beta \cdot f_{2}\left(F_{j}^{S}, F_{j}^{E}\right). \quad (8)$$

where $\alpha$ denotes the concatenation operation, $\beta$ represents the multi-heads of the self-attention module.

(2) Channel Split: The encoded features are matched to different heads after channel chunk operation:

$$\{e_{i1}, e_{i2}, e_{i3}, e_{i4}\} = \text{chunk}(e_{i}). \quad (9)$$

For this reason, we embed features into the multi-heads of the self-attention module to simultaneously represent multi-scale information using a single-scale transformer, the scale gap in remote sensing imagery makes it difficult to represent the multi-scale information in the satellite video. To be more specific:

(1) Encoder: It is necessary to embed LR features into the embedding space for attention calculation. To save memory overhead, we first use 3 × 3 convolution with a stride of 2 to achieve feature down-sampling, and increase the number of feature channels to 256. Denote the feature before encoding as $x = \{F_{j}^{S}, F_{j}^{R}, \ldots, F_{j}^{E}\}$, where $T$ is the number of frames and $F_{j}^{R} \in \mathbb{R}^{h \times w \times 64}$ for $j \in [1, T]$. The encoded features $e_{i} \in \text{Encoder}(F_{j}^{R})$ is taken from the results of optical flow and DConv respectively to achieve the complementation. Finally, we blend the coupling features and the final synthesis feature $F_{j}^{S}$.

$$F_{j}^{S} = \alpha \cdot f_{1}\left(F_{j}^{S}, F_{j}^{R}\right) + \beta \cdot f_{2}\left(F_{j}^{S}, F_{j}^{E}\right). \quad (8)$$

where $\alpha$ denotes the concatenation operation, $\beta$ represents the multi-heads of the self-attention module.

(2) Channel Split: The encoded features are matched to different heads after channel chunk operation:

$$\{e_{i1}, e_{i2}, e_{i3}, e_{i4}\} = \text{chunk}(e_{i}). \quad (9)$$

For this reason, we embed features into the multi-heads of the self-attention module to simultaneously represent multi-scale information using a single-scale transformer, the scale gap in remote sensing imagery makes it difficult to represent the multi-scale information in the satellite video. To be more specific:

(1) Encoder: It is necessary to embed LR features into the embedding space for attention calculation. To save memory overhead, we first use 3 × 3 convolution with a stride of 2 to achieve feature down-sampling, and increase the number of feature channels to 256. Denote the feature before encoding as $x = \{F_{j}^{S}, F_{j}^{R}, \ldots, F_{j}^{E}\}$, where $T$ is the number of frames and $F_{j}^{R} \in \mathbb{R}^{h \times w \times 64}$ for $j \in [1, T]$. The encoded features $e_{i} \in \text{Encoder}(F_{j}^{R})$ is taken from the results of optical flow and DConv respectively to achieve the complementation. Finally, we blend the coupling features and the final synthesis feature $F_{j}^{S}$.

$$F_{j}^{S} = \alpha \cdot f_{1}\left(F_{j}^{S}, F_{j}^{R}\right) + \beta \cdot f_{2}\left(F_{j}^{S}, F_{j}^{E}\right). \quad (8)$$

where $\alpha$ denotes the concatenation operation, $\beta$ represents the multi-heads of the self-attention module.

(2) Channel Split: The encoded features are matched to different heads after channel chunk operation:

$$\{e_{i1}, e_{i2}, e_{i3}, e_{i4}\} = \text{chunk}(e_{i}). \quad (9)$$

For this reason, we embed features into the multi-heads of the self-attention module to simultaneously represent multi-scale information using a single-scale transformer, the scale gap in remote sensing imagery makes it difficult to represent the multi-scale information in the satellite video. To be more specific:

(1) Encoder: It is necessary to embed LR features into the embedding space for attention calculation. To save memory overhead, we first use 3 × 3 convolution with a stride of 2 to achieve feature down-sampling, and increase the number of feature channels to 256. Denote the feature before encoding as $x = \{F_{j}^{S}, F_{j}^{R}, \ldots, F_{j}^{E}\}$, where $T$ is the number of frames and $F_{j}^{R} \in \mathbb{R}^{h \times w \times 64}$ for $j \in [1, T]$. The encoded features $e_{i} \in \text{Encoder}(F_{j}^{R})$ is taken from the results of optical flow and DConv respectively to achieve the complementation. Finally, we blend the coupling features and the final synthesis feature $F_{j}^{S}$.

$$F_{j}^{S} = \alpha \cdot f_{1}\left(F_{j}^{S}, F_{j}^{R}\right) + \beta \cdot f_{2}\left(F_{j}^{S}, F_{j}^{E}\right). \quad (8)$$

where $\alpha$ denotes the concatenation operation, $\beta$ represents the multi-heads of the self-attention module.
Muhalag (Bahrain). Their spatial resolution is 1 m, the frame rate is 25/fps and the duration of each video is 30 s. The spatial pixel size of each video is 4096\times2160 (except for 3840\times2160 in San Francisco (United States)). San Diego (United States) and Minneapolis-01 (United States) scenes are selected to build our test set and the other eight scenes are used to construct the training set. Finally, we get 2646 short video clips as our training data. To construct LR and LFR input video, we eliminate even frames in each short video clip, leaving only odd frames, and use the \texttt{imresize} function in MATLAB to downsample the frame to size 160\times160. Five subareas in San Diego (United States) and Minneapolis-01 (United States) are cropped respectively to make up our ten test clips. Some training and test samples are shown in Fig. 3 (a) and 3(b), 3(c) shows 5 test clips extracted from San Diego (United States).

3.2. Training details

In this paper, we only focus on 4\times S-SR and 2\times T-SR. To ensure the fairness of comparison, we follow the setting in Haris et al. (2020) and take 4 odd-indexed frames as input, and predict 7-frame HR and HFR sequences.

(1) Loss function: To optimize the network, we adopt a Carbonnier penalty function \cite{Lai et al., 2017} as our loss function which can be expressed as:

\[ L = \sqrt{\| f_{HR}^t - f_{SR}^t \|^2 + \epsilon^2}, \]

where \( f_{HR}^t \) is the ground truth frame, \( f_{SR}^t \) is the predicted SR frame and \( \epsilon = 10^{-3} \) is the empirical parameter. Under the guidance of the loss function, our network can be trained end-to-end in a supervised manner to make the predicted SR frames as close as possible to the ground truth frames.

(2) Training Strategy: We use Adam as our optimizer, where \( \beta_1 = 0.9 \) and \( \beta_2 = 0.999 \). The min-batch is set to 2, and we randomly crop the 160\times160 LR frames into 80\times80 patches as input. Also, we use image rotating and flipping to augment our data. The learning rate is initialized to \( 1 \times 10^{-4} \), and it is reduced by a factor of 10 when reaching half of the total 50 epochs. It took about 40 h on a single NVIDIA RTX 2080Ti GPU to train our model.

3.3. Comparison with state-of-the-art methods

To verify the effectiveness of our method in breaking the spatial and temporal resolution barriers for satellite video, we have compared our method with several SOTA (state-of-the-art) methods in terms of quantitative and qualitative performance. First, some two-stage methods composed of T-SR and S-SR are compared. Specifically, a powerful method named XVFI \cite{Sim et al., 2021} is chosen to achieve T-SR. Next, bicubic interpolation is selected as the baseline of S-SR. Then, two widely used SISR methods include: EDSR \cite{Lim et al., 2017} and RCAN \cite{Zhang et al., 2018}, and two VSR approaches RBPN \cite{Haris et al., 2019} and EDVR \cite{Wang et al., 2019} are used to generate HR frames. Since the idea of back projection has been shown to be effective in satellite VSR \cite{Xiao et al., 2021}, we choose RBPN as a comparison method for VSR. EDVR is a SOTA VSR method, which represents the leading performance. Finally, since no one-stage ST-SR method has been proposed in the field of remote sensing, two one-stage joint ST-SR methods named STARNet \cite{Haris et al., 2020} and ZoomingSloM \cite{Xiang et al., 2020} are also used for comparison. ZoomingSloM is lightweight and advanced. STARNet is a more elaborate and larger model.

(1) Evaluation Metrics: Peak Signal-to-Noise Ratio (PSNR) and SSIM \cite{Hore and Ziou, 2010} are two widely used image quality evaluation indicators in the presence of reference images. In addition, we also calculated the Root Mean Square Error (RMSE) and the Correlation Coefficient (CC). Besides, we introduced a reference-free image quality evaluation index NIQE \cite{Mittal et al., 2012}. The lower the NIQE, the more natural the image is in human vision.

(2) Quantitative Evaluation: The average PSNR and SSIM results of each method on the 10 test videos are shown in Table. 1. Our model
Fig. 4. × 4 SR results on test clip 000 (top) and 001 (bottom). We zoom the details for better comparison.

Table 1
The PSNR/SSIM results on 10 test video clips. Red and blue indicates the best and the second best performance, respectively.

<table>
<thead>
<tr>
<th>Method</th>
<th>Baseline</th>
<th>XVFI + SISR</th>
<th>XVFI + VSR</th>
<th>XVFI + EDVR</th>
<th>XVFI + STARNet</th>
<th>ZoomingSloM</th>
<th>Ours</th>
<th>Ground Truth</th>
</tr>
</thead>
<tbody>
<tr>
<td>FI + Bicubic</td>
<td>FI + EDSR</td>
<td>FI + RCAN</td>
<td>FI + RBPN</td>
<td>FI + EDVR</td>
<td>STARNet 5.5</td>
<td>5.5</td>
<td>5.5</td>
<td>5.5</td>
</tr>
<tr>
<td>000</td>
<td>31.02/0.9089</td>
<td>35.09/0.9541</td>
<td>35.58/0.9590</td>
<td>35.67/0.9585</td>
<td>35.77/0.9598</td>
<td>35.75/0.9572</td>
<td>35.68/0.9595</td>
<td>35.79/0.9603</td>
</tr>
<tr>
<td>001</td>
<td>29.68/0.8794</td>
<td>32.82/0.9259</td>
<td>33.18/0.9424</td>
<td>33.42/0.9421</td>
<td>33.25/0.9433</td>
<td>33.25/0.9410</td>
<td>33.24/0.9411</td>
<td>33.26/0.9426</td>
</tr>
<tr>
<td>002</td>
<td>31.89/0.9220</td>
<td>35.56/0.9562</td>
<td>36.32/0.9627</td>
<td>36.43/0.9623</td>
<td><strong>36.80/0.9649</strong></td>
<td>36.19/0.9600</td>
<td>36.32/0.9616</td>
<td>36.49/0.9626</td>
</tr>
<tr>
<td>003</td>
<td>30.43/0.8961</td>
<td>33.45/0.9373</td>
<td>33.94/0.9432</td>
<td>33.96/0.9436</td>
<td>34.14/0.9448</td>
<td>34.05/0.9429</td>
<td>34.08/0.9438</td>
<td><strong>34.20/0.9449</strong></td>
</tr>
<tr>
<td>004</td>
<td>28.57/0.8628</td>
<td>31.57/0.9216</td>
<td>31.94/0.9290</td>
<td>32.04/0.9300</td>
<td>32.01/0.9299</td>
<td>31.93/0.9271</td>
<td>32.10/0.9295</td>
<td><strong>32.18/0.9308</strong></td>
</tr>
<tr>
<td>005</td>
<td>28.11/0.8448</td>
<td>30.71/0.9017</td>
<td>30.98/0.9078</td>
<td>31.12/0.9104</td>
<td>30.95/0.9073</td>
<td>31.19/0.9104</td>
<td>31.34/0.9132</td>
<td><strong>31.35/0.9133</strong></td>
</tr>
<tr>
<td>006</td>
<td>30.57/0.9000</td>
<td>33.93/0.9460</td>
<td>34.28/0.9564</td>
<td>34.48/0.9520</td>
<td>34.36/0.9511</td>
<td>34.41/0.9506</td>
<td>34.50/0.9519</td>
<td><strong>34.63/0.9530</strong></td>
</tr>
<tr>
<td>007</td>
<td>32.19/0.8714</td>
<td>31.59/0.9196</td>
<td>31.82/0.9236</td>
<td>31.96/0.9261</td>
<td>31.82/0.9237</td>
<td>32.16/0.9274</td>
<td>32.26/0.9301</td>
<td><strong>32.26/0.9294</strong></td>
</tr>
<tr>
<td>008</td>
<td>32.43/0.9243</td>
<td>36.20/0.9606</td>
<td>36.65/0.9648</td>
<td>36.69/0.9649</td>
<td>36.81/0.9654</td>
<td>36.56/0.9632</td>
<td>36.71/0.9650</td>
<td><strong>36.85/0.9654</strong></td>
</tr>
<tr>
<td>009</td>
<td>30.79/0.8984</td>
<td>34.01/0.9417</td>
<td>34.33/0.9464</td>
<td>34.27/0.9462</td>
<td>34.39/0.9469</td>
<td>34.40/0.9459</td>
<td>34.51/0.9466</td>
<td><strong>34.64/0.9490</strong></td>
</tr>
<tr>
<td>Avg.</td>
<td>30.27/0.8903</td>
<td>33.49/0.9375</td>
<td>33.96/0.9429</td>
<td>33.97/0.9436</td>
<td>34.03/0.9437</td>
<td>33.97/0.9426</td>
<td>34.07/0.9442</td>
<td><strong>34.18/0.9451</strong></td>
</tr>
</tbody>
</table>
reveals peak performance on all test sets. All of the deep-learning-based methods have significantly improved compared to the baseline. Among the two-stage methods, FI + SISR is slightly inferior to FI + VSR because using SISR in multi-temporal frames cannot utilize the redundant information between frames to enhance the spatial resolution. In the one-stage method, STARNet can achieve comparable effect with XVFI + RBPN, but it is a little bit worse than XVFI + EDVR. ZoomingSloM is slightly better than XVFI + EDVR, which proves that the joint enhancement not only has end-to-end advantage, but also reaches higher performance than existing SOTA two-stage method.

Comprehensively evaluating model performance and efficiency, our method is ahead of the SOTA one-stage method ZoomingSloM by 0.11 dB and leads STARNet by 0.21 dB, highlighting the superiority of our framework. What’s more noteworthy is that all two-stage methods have huge parameters due to the combination of two independent tasks T-SR + S-SR. Although STARNet can jointly achieve ST-SR and obtain satisfying results, the amount of parameters has reached nearly 20 times that of our method (111.6 M v.s. 5.7 M), which is evidently inefficient. The key reason is that STARNet introduces a UNet structure with a mass of parameters to enhance optical flow information for accurate missing motion information estimation. In our framework, even though the initial optical flow is rough, no additional parameters will be introduced. On the one hand, the optical flow can provide motion information estimation; on the other hand, it can be used as a constraint in the convergence of DConv to mitigate the learning pressure and make DConv focus on learning the information that the optical flow cannot pay attention to. Compared with the best two-stage method XVFI + EDVR, our model is nearly only 1/5 of its size (26.1 M v.s. 5.7 M) and has the advantage of joint ST-SR to exploit the inherent relationship between T-SR and S-SR to restrict each other and promote. The parameters of our method are even nearly equal to XVFI (5.5 M), but we can still accomplish both S-SR and T-SR simultaneously. Thanks to the multiscale self-attention mechanism in the transformer, we can effectively mine the scale variable context information in long time series frames without introducing too many parameters.

In Table 2, we further divided all frames into predicted frames that originally did not exist (even-indexed frames) and originally existing input frames (odd-indexed frames). In terms of even frames, the one-stage method STARNet and ZoomingSloM can achieve similar results to FI + VSR, but the performance on odd frames is lower than FI + VSR, which indicates that the one-stage method suffers from weakness of input frames (odd-indexed frames). In terms of even frames, the one-stage method STARNet demonstrates that our FDC feature interpolation can accurately predict our model not only achieved 0.16 dB ahead of ZoomingSloM when originally did not exist (even-indexed frames) and originally existing S-SR. Although STARNet can jointly achieve ST-SR and obtain satisfactory results, the amount of parameters has reached nearly 20 times that of our method (111.6 M v.s. 5.7 M), which is evidently inefficient. The key reason is that STARNet introduces a UNet structure with a mass of parameters to enhance optical flow information for accurate missing motion information estimation. In our framework, even though the initial optical flow is rough, no additional parameters will be introduced. On the one hand, the optical flow can provide motion information estimation; on the other hand, it can be used as a constraint in the convergence of DConv to mitigate the learning pressure and make DConv focus on learning the information that the optical flow cannot pay attention to. Compared with the best two-stage method XVFI + EDVR, our model is nearly only 1/5 of its size (26.1 M v.s. 5.7 M) and has the advantage of joint ST-SR to exploit the inherent relationship between T-SR and S-SR to restrict each other and promote. The parameters of our method are even nearly equal to XVFI (5.5 M), but we can still accomplish both S-SR and T-SR simultaneously. Thanks to the multi-scale self-attention mechanism in the transformer, we can effectively mine the scale variable context information in long time series frames without introducing too many parameters.

In Table 2, we further divided all frames into predicted frames that originally did not exist (even-indexed frames) and originally existing input frames (odd-indexed frames). In terms of even frames, the one-stage method STARNet and ZoomingSloM can achieve similar results to FI + VSR, but the performance on odd frames is lower than FI + VSR, which indicates that the one-stage method suffers from weakness of excavating enough redundant information from frame sequence. And our model not only achieved 0.16 dB ahead of ZoomingSloM when predicting unknown frames and even led the best two-stage method XVFI + EDVR by 0.17 dB on the original frame. This result not only demonstrates that our FDC feature interpolation can accurately predict missing frames, but also shows that our MSTT is able to better extract redundant information from all frames.

Table 2
<table>
<thead>
<tr>
<th>Method</th>
<th>Even frame</th>
<th>Odd frame</th>
<th>Overall</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>PSNR (dB)</td>
<td>SSIM</td>
<td>PSNR (dB)</td>
</tr>
<tr>
<td>XVFI + Bicubic</td>
<td>30.24</td>
<td>0.8902</td>
<td>30.30</td>
</tr>
<tr>
<td>XVFI + EDSR</td>
<td>33.49</td>
<td>0.9374</td>
<td>33.50</td>
</tr>
<tr>
<td>XVFI + RCAN</td>
<td>33.89</td>
<td>0.9429</td>
<td>33.91</td>
</tr>
<tr>
<td>XVFI + EDVR</td>
<td>33.99</td>
<td>0.9432</td>
<td>34.07</td>
</tr>
<tr>
<td>XVFI + RBPN</td>
<td>33.94</td>
<td>0.9432</td>
<td>34.01</td>
</tr>
<tr>
<td>STARNet</td>
<td>33.98</td>
<td>0.9436</td>
<td>34.02</td>
</tr>
<tr>
<td>ZoomingSloM</td>
<td>34.06</td>
<td>0.9439</td>
<td>34.08</td>
</tr>
<tr>
<td>Ours</td>
<td>34.11</td>
<td>0.9444</td>
<td>34.24</td>
</tr>
</tbody>
</table>

In Table 3, our method achieves higher fidelity, the image quality is also pleasing, which illustrates our MSTT can introduce less useless interference information and only exploit valuable context information.

(3) Qualitative Evaluation: The qualitative evaluation focuses on the texture and details of the results from a visual perspective. We have magnified the local information to better observe the details in Fig. 4. Both XVFI + EDSR and XVFI + EDVR have severe distortion, while XVFI + RCAN and XVFI + RBPN have a certain degree of blur, and the edge of buildings are not clear enough. Our method generates the most negligible blur and more texture information. The same situation can be found in test scene 001. In the XVFI + RBPN results, the wing of the aircraft was deformed. In XVFI + EDSR, XVFI + RCAN, and STARNet, the tail of the aircraft also had significant artifacts. As shown in Fig. 5, on a moving aircraft with a smaller scale in 007, only our method recovered a clear wing. Moreover, the tail of the aircraft in XVFI + EDVR is seriously distorted and mixed with the background, which was difficult to distinguish. We calculated the residual between the ground truth and the predicted frame and normalized it to [0, 1] as shown in error maps below, our results are closer to the ground truth with minimal errors, thus more detailed information is recovered. Our method can achieve good generalization in aircraft with various scales, which further proves the effectiveness of the multi-scale design in MSTT.

4. Discussions

4.1. Discussion on the coupling of optical flow and DConv

Here we need to discuss the bottleneck in using only optical flow or deformable convolution to synthesize missing frames, and these two methods can be complementary. Three experiments were designed for this purpose. Firstly, we only use the optical flow method to estimate the intermediate missing frame. Specifically, after optical flow warp operation, two warped feature maps are blended directly. We denote this model as Flow-only. Similarly, when we only adopt DConv to synthesize intermediate frames, we blend the results of deformable convolution, and this model is denoted as D-only. To ensure a fair comparison, we use four 1 x 1 convolutions to achieve the blending operation, the same number of convolutions used in our FDC feature interpolation module. The third model is our FDC method, which simultaneously introduces these two methods for deep coupling complementary. The experimental results are shown in Table. 4 and training process can be seen in Fig. 6(a). Flow-only and DConv-only can achieve similar performance. However, when they are combined, PSNR can increase by 0.1 dB and 0.8 dB, respectively, which fully demonstrates that the deep coupling idea can achieve complementary advantages for better results.

4.2. Discussion on the different strategy of coupling

Here we discuss different manners to realize coupling and prove the effectiveness of our deep coupling strategy. A direct approach is to mix the optical flow estimation result with the DConv estimation result (naive coupling). In this case, the optical flow and DConv estimation processes are essentially separate, just a shallow coupling of different results. It can be denoted as:

\[ F^A_{t} = \alpha \ast F_{t_i}^{14} \left( [F^1_{t_i}, F^2_{t_i}] \right) + \beta \ast F_{t_i}^{14} \left( [F^1_{t_i}, F^2_{t_i}] \right) \]  

(14)

Table 3
<table>
<thead>
<tr>
<th>Method</th>
<th>RMSE</th>
<th>CC</th>
<th>NIQE</th>
</tr>
</thead>
<tbody>
<tr>
<td>XVFI + Bicubic</td>
<td>7.9079</td>
<td>0.98486</td>
<td>19.7265</td>
</tr>
<tr>
<td>XVFI + EDVR</td>
<td>5.1998</td>
<td>0.99299</td>
<td>18.4472</td>
</tr>
<tr>
<td>XVFI + RBPN</td>
<td>5.2150</td>
<td>0.99305</td>
<td>18.1580</td>
</tr>
<tr>
<td>STARNet</td>
<td>5.0999</td>
<td>0.99310</td>
<td>18.0570</td>
</tr>
<tr>
<td>ZoomingSloM</td>
<td>5.1987</td>
<td>0.99326</td>
<td>18.0953</td>
</tr>
<tr>
<td>Ours</td>
<td>5.1054</td>
<td>0.99336</td>
<td>17.8575</td>
</tr>
</tbody>
</table>
Our method proposes the idea of deep coupling to realize the information coupling in the intermediate process of the two estimations and finally mixes the intermediate coupling results to achieve the purpose of deep coupling. In other words, this design can make the two methods complement each other earlier. The experimental results are shown in Table 5. The parameters of the two coupling approaches remain unchanged, but the deep coupling can exceed the shallow coupling. The training process is shown in Fig. 6 (b).

4.3. Discussion on multi-scale strategy

In this part, we prove that our strategy of matching multi-scale patches in multiple heads is more conducive to mining the scale variable information of satellite videos. In other words, this design can make the two methods complement each other earlier. The experimental results are shown in Table 5. The parameters of the two coupling approaches remain unchanged, but the deep coupling can exceed the shallow coupling. The training process is shown in Fig. 6 (b).

4.4. Discussion on model efficiency

We explored the structure and efficiency of the model and proved that our method is lightweight and efficient. As shown in Fig. 6(d), we find that the performance of using 20 residual blocks is almost the same as that of 40 residual blocks. Therefore, we select 20 residual blocks for reconstruction. We also calculated FLOPs and a more intuitive indicator average processing time of each frame to better measure model complexity. The results are shown in Table 7. Compared with the two-stage method, the complexity of our model is within an acceptable range, and the processing time is significantly faster. Compared with the one-stage method, our model has achieved performance by a great margin. Hence, we achieved the best trade-off between efficiency and performance.

4.5. Discussion the effect of ST-SR on moving object tracking

To explore the significance of SRVSR for moving object tracking, we have tracked the moving vehicle in a SkySat scene by Zhang et al. (2020). The original video size is 320 × 320 and the number of frames is 50, after STVSR, the resolution increased to 1280 × 1280 and the number of frames was 99. We calculated the average recall of all frames, and the results are shown in the Table 8. In the original 50 frames, the recall increased by 4.7%, and for the entire 99 frames it increased by 5.3%. This illustrates that the increase in resolution and the number of frames is beneficial for moving object tracking.
4.6. Discussion of guidelines for satellite video ST-SR

Although ST-SR can achieve end-to-end spatial and temporal resolution improvements, its design is still not sophisticated enough compared to the independent tasks S-SR and T-SR, which have been widely studied so far. For arbitrary factors of T-SR, some works have attempted to synthesize missing frames at any temporal location through modulation networks (Xu et al., 2021) or controlled optical flow estimation (Shi et al., 2021), which may be the guidelines of future research. For large scale of S-SR ($\times8$, $\times16$), most methods have poor generalization performance when facing such a more challenging
situation. More effort needs to be put into the large factor S-SR problem.

5. Conclusion

We present a lightweight framework that can jointly enhance the spatial and temporal resolution of satellite video in one-stage manner. A missing frame prediction module was proposed to predict the latent motion information with various scales. Under the constraints of optical flow, multi-scale deformable convolution can better converge and adaptively learn the supplementary motion information. The proposed multi-scale spatial–temporal transformer can effectively aggregate the contextual information in long-time series frames. Experiments on Jilin-1 satellite video demonstrate that our method can accurately predict the nonexistent frames and enhance the spatial resolution simultaneously.

In ongoing work, we will focus on the ST-SR under extreme events or large motion in satellite videos. In addition, owing to the insufficient data sets in real scenes, the performance of models trained on simulated degradation will severely drop in the real world, it is worth to build a real world data set that can reflect the complex degradation in satellite video.

CRediT authorship contribution statement


Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

This work was supported in part by the National Natural Science Foundation of China (41922008,61971319) and the Hubei Science Foundation for Distinguished Young Scholars (2020CFA051).

References


