EAI Endorsed Transactions
on e-Learning
Research Article EAI.EU

Transformer-Guided Video Inpainting Algorithm Based on Local Spatial-Temporal joint

Jing Wang [1*] and Zongju Yang[1]

[1]1st School of Software, Henan Polytechnic University, Jiaozuo, 454003, China.

Abstract

INTRODUCTION: Video inpainting is a very important task in computer vision, and it's a key component of various practical applications. It also plays an important role in video occlusion removal, traffic monitoring and old movie restoration technology. Video inpainting is to obtain reasonable content from the video sequence to fill the missing region, and maintain time continuity and spatial consistency.
OBJECTIVES: In previous studies, due to the complexity of the scene of video inpainting, there are often cases of fast motion of objects in the video or motion of background objects, which will lead to optical flow failure. So the current video inpainting algorithm hasn't met the requirements of practical applications. In order to avoid the problem of optical flow failure, this paper proposes a transformer-guided video inpainting model based on local Spatial-temporal joint.
METHODS: First, considering the rich Spatial-temporal relationship between local flows, a Local Spatial-Temporal Joint Network (LSTN) including encoder, decoder and transformer module is designed to roughly inpaint the local corrupted frames, and the Deep Flow Network is used to calculate the local bidirectional corrupted flows. Then, the local corrupted optical flow map is input into the Local Flow Completion Network (LFCN) with pseudo 3D convolution and attention mechanism to obtain a complete set of bidirectional local optical flow maps. Finally, the roughly inpainted local frame and the complete bidirectional local optical flow map are sent to the Spatial-temporal transformer and the inpainted video frame is output.
RESULTS: Experiments show that the algorithm achieves high quality results in the video target removal task, and has a certain improvement in indicators compared with advanced technologies.
CONCLUSION: Transformer-Guided Video Inpainting Algorithm Based on Local Spatial-Temporal joint can obtain high-quality optical flow information and inpainted result video.

Keywords: video inpainting algorithm, flow-guided, attention mechanism, spatial-temporal transformer, Deep Flow Network, video target removal

Received on 18 March 2023, accepted on 16 June 2023, published on 15 August 2023

doi: 10.4108/eetel.3156

1. Introduction

Video inpainting refers to filling the content of damaged regions in the video with the generated seemingly reasonable content [1], including inpainting damaged pixels, removing tears or blurring [2], etc. Video inpainting is a very classical algorithm in the field of computer vision, which usually involves complex image processing technology, including image enhancement [3], denoising [4], interpolation [5], etc. As an important algorithm in the field of inpainting application, video inpainting can support many tasks of computer vision, such as video frame insertion and super-resolution, video denoising [6], damaged video inpainting [7], etc. These inpainting tasks of computer vision play an indispensable role in black and white video coloring, old video inpainting, and video

*Corresponding author. Email: wjasmine@hpu.edu.cn

watermark removal. The application of video inpainting algorithm is shown in Fig. 1. In Fig. 1(a), you can see the application of black and white video coloring, which is an old film and television work from modern China. The video inpainting algorithm can use a deep neural network trained in a large number of color images to color a single frame of black and white video.The application of 4k video inpainting can be seen in Fig. 1(b), the inpainting algorithm uses enhancement techniques to upgrade video to Ultra High Definition Standard in terms of physical resolution, video frame rate, color gamut, and color depth. The application of video watermark removal can be seen in Fig. 1(c). The inpainting algorithm erases the watermark in the video through a deep neural network.

(a) black and white video coloring

(b) 4k video inpainting

(c) video watermark removal

Fig. 1 Video inpainting application

1.1. Research Background and Significance

In recent years, with the economic development and scientific progress, artificial intelligence has gradually become an indispensable technology in people's life. As an important part of artificial intelligence, computer vision has been integrated into our life, including detection technology [8], segmentation technology [9], inpainting technology [10], etc. Among them, inpainting is a technology to complete the object without knowing the missing part information of the object. As early as the 15th century, many artists used their imagination to manually inpaint medieval works of art (such as murals) in the way of filling [11], making the structure of the inpainted works of art more complete and the texture clearer. However, this early inpainting technology has the disadvantage that it is difficult to restore after the inpainting error, and the art inpainted in this way is also easy to be damaged by others. Until later 2000, Mr. Bertalmio et al. [12] proposed for the first time to simplify image inpainting into a mathematical expression, which is a digital image inpainting method and can fill the texture and structure of the missing region of the image. So far, artists are not inpainting the entity of artworks, but only scanning the artworks and old books to the computer for inpainting. This method of digital image inpainting saves human resources while protecting literary and artistic films.

Video inpainting technology is widely used in the restoration of old film and television works [13, 14], the restoration of occluded video [15-17], and the removal of video objects [18]. With the development of digital devices, from digital cameras to smart phones, there are more and more ways to capture and save videos. Due to the problems of blurred flicker, too many times of playing, and film damage of old literary and artistic films, it is often necessary to use digital video inpainting technology to inpaint the video. Moreover, in the process of video shooting, uncontrollable factors such as external noise and object occlusion in the video often occur, resulting in missing regions in the video. In addition, it also plays an important role in video surveillance [19, 20], remote sensing satellites [21, 22] and other fields. In the task of plant protection, the remote sensing satellite image video information can monitor the condition of plants and prevent pests from eating. In the field of security, video surveillance combined with face recognition technology can be used to hunt down criminal gangs. Video is a four-dimensional sequence, so the video inpainting algorithm must meet the spatial consistency of each frame in the inpainted result video and the time continuity of all sequences. The research significance of video inpainting is as follows:

(1) Restoration of old film and television works

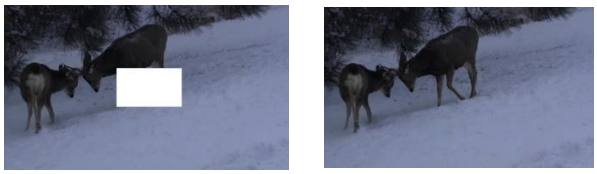
Old film and television works may be damaged after unreasonable play, storage or scanning, so the preservation and recovery of old film and television works is a very important task. The old video image may be less than 10 frames per second, and the Playback will be very slow. Therefore, the video inpainting uses the frame filling method to make the old video become more than 60 frames per second, and uses the video coloring method to make the black and white video into color video, which looks more comfortable. As shown in **Fig. 2** In addition, the old film and television works also include video noise reduction, video resolution expansion and other methods. Video noise reduction is aimed at the case of unexpected noise in the process of saving old video; Video resolution amplification technology aims at 4k video inpainting to improve the definition of video.

(a) old video (b) frame recovery and coloring

Fig. 2 Recovery of old video

(2) Corruped video inpainting

Corrupted video is defined as occlusion (such as video watermark) in some regions in a frame of video. It is necessary to use video inpainting technology to remove these corruped regions. Generally speaking, the position of the occluded region is fixed. **Fig. 3** shows the video image of the middle fixed occlusion region and the inpaintinged video.



(a) corruped video (b) inpainting result

Fig. 3 Inpainting of corruped video

(3) Video target removal

Nowadays, with the popularity of short video platforms, more than 400 million creative videos are submitted every day. However, due to the limitations of shooting conditions and equipment performance, some materials may have problems. For example, when shooting in busy streets, unexpected objects often enter the camera by mistake. In view of this situation, video inpainting technology can help post-production personnel remove redundant targets in the video and ensure the quality of the final video work. As shown in Fig. 4.

Fig. 4. Video object removal

For the above problems, video inpainting algorithm based on Deep Flow Network has very important research significance in practical applications. However, achieving high-quality inpainting is still a difficult task in the field of computer vision. Therefore, it is a valuable research topic to solve the problems of video inpainting on the basis of Deep Flow Network.

1.2. Research Background and Significance

Before the popularity of deep learning, most image inpainting methods were mainly divided into these categories: diffusion-based inpainting, texture-based inpainting and block-based inpainting. First of all, the diffusion-based inpainting method diffuses information in the neighborhood around the target missing region, so that the pixel value of the missing region can be calculated from the surrounding neighborhood pixels, so as to achieve the purpose of inpainting the image. It can effectively deal with the noise and distortion in the image, but this method is difficult to inpaint the image of a large region of missing regions. Secondly, the method based on texture synthesis is also a common image inpainting method. It synthesizes the texture of the missing region in the image and the texture of the surrounding region, so as to achieve the purpose of inpainting the image. It has high-quality inpainted results for inpainting images with large regions of missing regions. The block-based inpainting method [23, 24] divides the original image into several blocks, samples information from known blocks similar to the missing blocks, and fills them into the missing blocks to complete the inpainting. Finally, the block-based image inpainting method can effectively solve the local missing in the image, but the effect is poor when processing the image with large region missing, and the algorithm complexity is high. These traditional image inpainting methods still have defects, the main reason is that these methods are based on the known information of the image to fill the information of the missing region of the image. After the deep learning method became popular, the learning-based method began to penetrate into the field of image inpainting. This kind of method can generate some new content that is not in the original image.

These traditional image inpainting methods have been extended to the field of video inpainting. Traditional methods may not maintain the time continuity of inpainted results in the process of video inpainting. More importantly, unlike the deep learning method, traditional methods can't capture high-level semantic information. Therefore, traditional methods can't inpaint video content in complex motion regions containing one or more objects. After years of research and development, video inpainting technology is mainly divided into two categories: block-based video inpainting, and video inpainting based on deep neural network.

1.2.1. Block-based video inpainting algorithm

Early video inpainting methods mainly described the inpainting process as a block-based optimization problem. The block-based method is roughly the same as the image inpainting method. According to the set priority, the best matching block is searched from the known blocks similar to the missing block, and the missing block is filled. In 2006, Shiratori et al. [25] proposed a method based on sports field transmission to complete video content. This method fills the missing region of the video by sampling the spatial-temporal block of local motion instead of sampling the color. Once the local motion field is

calculated in the missing region of the video, the color can be propagated to produce a coherent seamless video. In 2007, Wexler et al. [26] proposed a new block-based video completion method. It takes the video completion task as a global optimization problem of an objective function, by sampling a 5 × 5 × 5 and iterate to search the nearest neighbor block and calculate the weighted average to fill the missing region. In 2012, Granados et al. In 2014, Strobel et al. [27] extended the image inpainting method based on sample blocks proposed by Criminis et al. [28] to the field of video inpainting. This method first proposes a color and flow inpainting to ensure the time consistency of the inpainted results even under the complex motion of the foreground object and the background object. In the same year, Newson et al. [29] proposed an automatic video inpainting algorithm based on global function optimization of 3D blocks. This method can deal with various complex situations in video inpainting, such as accurate reconstruction of dynamic texture, multiple moving objects and moving background, and is mainly used to solve the inpainting of dynamic video (such as video shoting by handheld camera).

In general, block-based video inpainting algorithms have been widely studied at new technologies, such as deep learning, are constantly introduced to improve the inpainting effect and operational efficiency of the algorithm. However, block-based inpainting algorithms usually assume that there is a uniform motion field in the missing region, and in general, the algorithm is limited by complex motion. In addition, block-based inpainting algorithms often have high computational complexity, which isn't feasible for real-time applications.

1.2.2. Video inpainting algorithm based on deep neural network

In 2019, Wang et al. [30] proposed a data-driven video inpainting method for recovering missing regions in video. For the first time, they proposed a new deep network framework that combines 3D and 2D full convolution. This framework includes two subnetworks: time structure estimation network and spatial detail inpainting network. The time structure estimation network is built on the 3D full convolution network. Because of the high computational cost of 3D convolution, only low-resolution video frames are learned to inpaint. However, the inpainted results of this method in complex scenes are fuzzy and have certain requirements for memory. The use of 3D convolution architecture limits the resolution of video. In the same year, Kim et al. [31] proposed a deep cyclic neural network model based on encoder-decoder that can quickly inpainting video. The model uses a fixed time domain window to collect and optimize information from a small number of adjacent frames and synthesize unknown regions. The time continuity of the output results is guaranteed through the cycle feedback and time storage module. However, because the algorithm uses a fixed time window, it can't transfer the content that is far away in time. In the same year, Xu et al. [32] proposed a video inpainting method based on optical flow-guided. They use the proposed Deep Flow Completion network to generate the relevant optical flow field in the spatial-temporal domain through coarse-to-fine optimization, and then guide the pixel propagation process by predicting the optical flow field to fill the missing region in the video. Under the guidance of the complementary optical flow field, the network can fill the missing regions in the video. In the same year, Murase et al. [33] used convolutional neural network to effectively estimate the optical flow field in the occluded background region. This method is applicable to all kinds of videos, but the completed video often contains visible artifacts, especially the spatial-temporal seams in the dynamic region. In 2020, Zeng et al. [34] proposed an aggregated spatial-temporal transformer network for video inpainting. The network fills the damaged regions in all video frames through the multi-head self-attention mechanism, and optimizes the aggregated spatial-temporal Transformer network by using the spatial-temporal counter loss. In the same year, Gao et al. [35] proposed a flow-based video completion algorithm, which solved the problem that the flow-guided completion method couldn't maintain the sharpness of the moving edge. This method first extracts and complements the moving edges, and uses them to guide the segmented smooth complementing flow of sharp moving edges. However, this method has limitations. They input frames with missing regions into the Deep Flow Network. the existence of masks may interfere with the motion of objects in the video. This has high quality results for video target removal, but it will appear blurred in the video inpainting scene. In 2022, Zhang et al. [36] proposed the Transformer model based on flow-guided, which uses the movement difference of flow to guide the attention recovery in the Transformer, and designed an flow completion network to complete the damaged flow through the optical flow characteristics of the reference frame, transmit the content across the video frame, and finally use the flow guidance transformer to synthesize the remaining missing regions. However, the calculation speed of the Transformer model based on flow guidance is slower than that of other methods, and it is very dependent on the complete optical flow. When inpainting video with large displacement motion, it will be unable to use the guidance of the complete flow.

From the research status we can see that video inpainting algorithm has entered a stage of rapid development. The video inpainting technology based on deep neural network can inpainting more complex scenes, and has a high-quality inpainting result. The general video inpainting technology based on the Deep Flow Network has achieved some results in the video target removal task, but for more effective optical flow-guided video inpainting algorithm, further exploration is needed. Therefore, the focus of this paper is to obtain high-quality flow, as well as effective optical flow-guided video inpainting algorithm.

1.3. The Innovation Main Content

Video inpainting is a common task in computer vision, and it is also widely used in the fields of restoration of old cultural relics, video monitoring technology and remote sensing satellites. In recent years, with the rapid development of deep neural networks, the research of video inpainting algorithms has made great progress. However, there are still many problems in the video inpainting algorithm based on Deep Flow Network. For example, in the scene of missing region in the middle of the video, if the motion, displacement or background of the object changes greatly, the optical flow will fail or the inpainted result will appear large area blur and artifacts. The existing video inpainting algorithm based on deep optical flow network does not completely solve these problems, and more exploration and research are needed. To solve the problem of obtaining high-quality predicted optical flow, we propose an optical flow-guided video inpainting algorithm based on spatial-temporal transformer. First, according to the strong spatial-temporal information that optical flow may have in the near time domain, a local time sampling window is designed to sample the frames in the neighborhood of the damaged frames, and a group of damaged local frames is obtained. On the one hand, take this group of damaged frames as the input of the LSTN, and obtain the roughly inpainted local frames through the encoder, decoder and spatial-temporal transformer. On the other hand, input this group of damaged frames into another optical flow completion network, calculate a group of local forward and backward optical flow diagrams with time-space information, and then pass through the LFCN with pseudo three-dimensional convolution layer, channel attention module and spatial attention module, the complete forward and backward local optical flow map is obtained. Finally, the rough inpainted local frame and the complete local optical flow map with spatial-temporal complementary information are sent into the spatial-temporal transformer module to obtain the final local inpainted results. The algorithm process is iterative until all damaged frames are inpainted. The experimental results show that the proposed algorithm can still obtain reasonable visual inpainted results while increasing the complexity of the model.

In general, the methods proposed in this paper have the following contributions:

1. To solve the problem of video target removal, we propose a flow-guided video inpainting algorithm based on spatial-temporal transformer.

2. The damaged frame is sampled through the local time sampling window, and Local Spatial-Temporal joint Network is designed. The network inputs the sampled reference frame and outputs the roughly inpainted local frame.

3. LFCN is designed, which combines the Deep Flow Network to predict the optical flow of the local damaged frame.

The structure of this paper is as follows:

We first review relevant work in Section 2, explain the network model framework of video inpainting algorithm in Section 3, and introduce the comparison of experimental results and indicators in Section 4. Finally, conclusions and prospects are given in Section 5.

2. Dataset

DAVIS dataset [37] is a publicly annotated dataset for video segmentation, which is characterized by dense pixel-level annotation of moving objects in each frame. The DAVIS-2016 dataset contains 90 video sequences, and the DAVIS-2017 dataset contains a total of 150 video sequences, including 60 train sequences and 90 test sequences, including various video scenes, such as moving objects, occluded objects, light changes and the background is complex. The DAVIS-2016 dataset annotates all frames of each video at the pixel level. The number of frames of each video in the DAVIS-2017 dataset ranges from 20 to 70, and annotates the moving objects of each video at the pixel level, which provides more fine-grained annotation information, such as the shape and posture of the target, while reducing the workload. Each video sequence has a ground truth for foreground and background segmentation. The pixel value of the background object is marked as 0, and the pixel value of the foreground object is marked as 1. The annotation information of DAVIS-2017 dataset can be used to evaluate the accuracy and robustness of video processing algorithms. The annotation information is shown in **Fig. 5**.

Table 1 shows the comparison of the number of videos, frames and objects in the DAVIS-2016 and DAVIS-2017 datasets.

Fig. 5 Diagram of DAVIS-2017 dataset annotation

Table 1 Data comparison between DAVIS-2016 and DAVIS-2017 datasets

	DAVIS-2016		DAVIS-2017	
	train	test	train	test
Number of sequences	30	20	60	90
Number of frames	2079	1376	4219	6240
Number of objects	30	20	138	238

Due to the finer granularity of the data annotation information of DAVIS-2017 dataset, several challenging

scenarios have been added, such as fast motion, blurring, low contrast, etc., and the openness of the dataset has promoted the research progress in the field of video processing, making it the benchmark dataset for many new video processing algorithms, attracting more and more researchers' attention and use.

Different from the small-scale DAVIS dataset, the YouTube-VOS dataset [38] is a large-scale dataset used for video algorithm research. The dataset was originally used for video target segmentation, including 3471 video train sequences with densely sampled multi-target annotations, 541 video test sequences and 507 video Validation sequences, with an average of 150 frames per video sequence. Each video sequence contains one or more foreground objects, and the foreground objects have a pixel-level annotation information. The YouTube-VOS dataset has large-scale video sequences, covering different scenes from different angles, such as outdoor, indoor, sports, etc.; It also contains many challenging video sequences, such as occluded objects, complex background, fast motion scenes. In addition, because the dataset searches data from YouTube video sharing platform, it contains some unique scenes and objects, such as humans, various animals and different types of vehicles. In short, YouTube-VOS dataset is widely used in the video field and has become one of the important benchmark datasets for evaluating and comparing various video processing algorithms.

Although both DAVIS and YouTube-VOS datasets are designed for video object segmentation, because the sequences in the datasets can extract rich motion information, this paper conducts model training and testing on these two datasets.

3. Methodology

With the rapid development and progress of science, Transformer has been introduced into various computer vision tasks, such as image classification [39-42], object detection [43, 44], motion detection [45], segmentation [46], due to its excellent ability to model and capture remote features. Video inpainting depends on the transmission of relevant content across frames in the spatial-temporal domain[47], so researchers have also applied Transformer to the video inpainting task. The video inpainting algorithm has obtained great achievements from the traditional block-based method to the flow-based method, and then from the convolutional neural network method to the attention mechanism based Transformer method. Transformer's development in the field of video inpainting is mainly based on its application in the field of natural language processing, and attempts to inpainting the frames of video. Later research proved that this idea is feasible. Transformer can inpainting the noise and missing frames in the video to improve the quality and visualization of the inpainted video. Generally, there are two transformer-based video inpainting methods. One transformer-based method is to use the self-attention mechanism to capture the correlation and dependency between video frames, and use the attention mechanism to inpaint video frames [36, 48, 49]. Generally speaking, the transformer-based method encodes the entire video through the encoder, and then uses the decoder to predict missing and damaged frames. In the decoder, the self-attention mechanism is used to calculate the correlation between the current frame and other frames, and the inpaintinged frame is generated according to the correlation. Another transformer-based video inpainting method is to use a multi-task learning framework to simultaneously perform video inpainting and inter-frame correlation prediction [105, 106]. Different from the first method, this method uses an encoder to extract features from the video and uses a self-attention mechanism to capture the relationship between frames. Then, the missing and damaged frames are predicted by two decoders. In general, the video inpainting method based on Transformer has solved some problems in the field of video inpainting to a certain extent, and has broad application prospects. However, there are still some limitations and challenges in the current method, such as the need for a large amount of computing resources and time, strong dependence on data, inaccurate attention retrieval, which need further research and improvement.

In addition, the previous flow-based video inpainting method [32, 35] divides the inpainting process into three steps: optical flow completion, pixel propagation and image synthesis. In the process of optical flow completion, the damaged flow is processed by completing the missing regions in the optical flow map. In the process of pixel propagation, the pixel value of the reference frame is propagated to the missing region under the guidance of the completion flow. Finally, in the process of image synthesis, the incomplete missing regions in the process of optical flow guided pixel propagation need to be synthesized using image inpainting methods. Compared with the method of directly synthesizing pixels, the flow-based method has better time consistency, because the flow-based method can explicitly propagate pixel values from adjacent frames[50]. However, flow-based methods may have defects in the process of optical flow completion[51]. For example, in the process of optical flow prediction, it is necessary to capture the spatial-temporal relationship from the damaged flow of other frames. Inaccurate optical flow estimation, nonlinear motion and fast motion between frames will cause the flow of adjacent frames to also have different values and directions, resulting in the predicted flow not maintaining time consistency. Therefore, this paper uses a local time window to sample the local reference frame and extract the optical flow map between the reference frames without considering the influence of the spatial-temporal information of other flows. Because the optical flow in the local time window is related to some extent, the method in this paper combines the characteristics of the local flow and fully utilizes the spatial-temporal relationship and complementary characteristics of the local flow to make the compensation accuracy of the damaged flow higher.

In this paper, a local time sampling window is designed to extract the local frames in the neighborhood of the damaged frame. On the one hand, rough inpainted frames are deduced through LSTN with attention mechanism; On the other hand, the local damaged optical flow map in the local time window is calculated, and LFCN is used to complete the optical flow map. Finally, the local damaged frame, the completed optical flow amp and the rough inpainted result are sent to the spatial-temporal transformer to get the final inpainted result video frame.

3.1. Network structure

Suppose $f = \{f_1, f_2, \dots, f_N\}$ is a video sequence with N frames, and the mask sequence corresponding to each video frame is $M = \{m_1, m_2, \dots, m_N\}$. Input video sequence V and corresponding mask sequence M into the video inpainting algorithm to obtain the inpainted result video sequence $\tilde{Y} = \{\widetilde{y_1}, \widetilde{y_2}, \widetilde{y_3}, \dots, \widetilde{y_T}\}$. The network framework of the inpainting algorithm is shown in **Fig. 6**, which is mainly divided into the following three steps: 1) Sampling local frames from the neighborhood of the damaged frame through the local time sampling window and inputting them into the LSTN, which outputs the roughly inpainted local frames. 2) The RAFT and LFCN are used to extract and complete the local reference optical flow map from the local time sampling window. 3) The spatial-temporal complementary relationship of the complete optical flow, the roughly inpainted frame and the partially damaged frame are input into the spatial-temporal transformer in parallel to obtain the final video inpainted result.

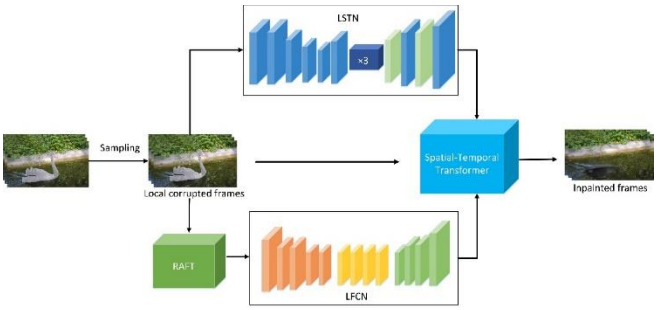


Fig. 6 Structure graph of Transformer Guided Video Inpainting Algorithm Based on Local Spatial-Temporal joint

3.2. Network structure

When inpainting videos of complex scenes, due to the fast motion of objects in the video, the motion of multiple background objects, the extraction and completion of the optical flow map by the Deep Flow Network may fail [36]. However, in a local time neighborhood in the motion field, there may be very strong correlation between frames. Therefore, this paper designs a local time sampling window to sample the neighborhood of the damaged frame, and takes five frames of images in the forward and backward of the damaged frame respectively. In addition, the local reference frame obtained through the local time sampling window is also very useful for the subsequent algorithm flow. The method in this paper can further predict and complete the flow through the relevant time relationship existing in the local optical flow [52]. In addition, it is difficult to complete the damaged optical flow without considering the damaged video content, so this paper designs a LSTN that can obtain RGB pixel values. The network includes an encoder, a decoder and three Transformer modules. The method inputs local adjacent frames and corresponding masks into the LSTN. The encoder in the network will extract the features of each local adjacent frame. After capturing the feature map, the Transformer module will fuse all the missing region information in the deep coding space of the local reference frame and send it to the decoder. The decoder will roughly complete the RGB pixels and reconstruct the spatial-temporal consistent local frame, this process is shown in formula (1).

$$\tilde{f} = \sum_i^k \boldsymbol{L}(\boldsymbol{f_i}, \boldsymbol{m_i}) \quad (1)$$

Where, L represents the LSTN model, k represents the size of the reference frame sampling window, and f_i represents the i th local frame, m_i represents the mask corresponding to the i th local frame, and the output result $\tilde{x}$ of the final network [53]. It is the roughly inpainted reference frame content, which can provide video reference information for the subsequent streamcompletion process.

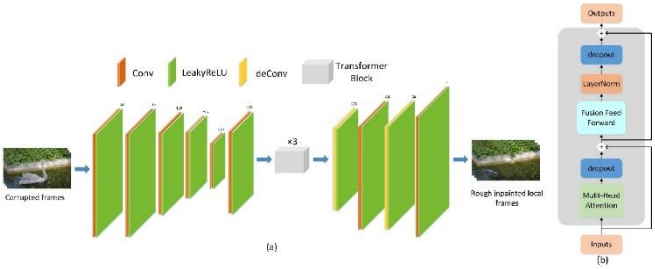


Fig. 7 Framework of Local Spatial-Temporal joint Network

The LSTN network structure is shown in **Fig. 7** (a). Among them, the orange block represents the two-dimensional convolution layer, and The size of convolution kernel used in this paper is 3 × 3, The light green color block represents the activation function layer and uses the LeakyReLU activation function; The yellow block represents the deconvolution layer, and the convolution core size is also 3 × 3. The gray rectangular block represents the Transformer module. The method in this paper uses three Transformer modules. Its algorithm details are shown in **Fig. 7** (b). The Mulit-Head Attention layer is used to form attention with multiple subspaces, which helps the model to focus on and capture the feature information of multiple layers. The dropout layer is used to prevent over-fitting and increase the robustness of the

network. In addition, because the matrix multiplication in the multi-head self-attention mechanism is linear transformation and the learning ability isn't as good as nonlinear transformation, the method in this paper introduces a feed forward network layer and uses a nonlinear activation function in its internal structure. Finally, the reconstruction loss of corrupted region and valid region is used to guide the training process. The reconstruction loss of corrupted region and valid region is shown in formula (2) and formula (3).

$$L_c = \frac{\|m_i \odot (\tilde{x}_i - x_i)\|_1}{\|m_i\|_1} \quad (2)$$

$$L_v = \frac{\|(1-m_i) \odot (\tilde{x}_i - x_i)\|_1}{\|1-m_i\|_1} \quad (3)$$

Where, L_c represents the reconstruction loss value of the corrupted region, L_v represents the reconstruction loss value of the valid region, $\|m_i\|_1$ represents the L_1 norm of the mask corresponding to the i th frame, and $\odot$ represents the Hadamard Product.

3.3. Local Flow Completion Network

In this paper, after the locally damaged frames are sampled through the local time window, in order to obtain the complementary characteristics of optical flow between them, the RAFT is first used to calculate the locally damaged optical flow map. In the process of predicting optical flow, it's necessary to use two frames of image input RAFT to estimate local flow, so the spatial-temporal information of other optical flows can't be considered. In addition, because every moving object has a degree of resistance to the change of its own motion state, there is a certain correlation between the motion of the object in the local time neighborhood of the motion field. The complementary characteristics of the flow can be combined through the local motion window, which can effectively improve the accuracy of the optical flow complement. In addition, 3D convolution is very expert in capturing the relevant information of spatial-temporal dimension in video, but 3D convolution has some defects, such as the huge amount of network parameters and the large amount of video memory required for calculation, which makes network optimization more complex.

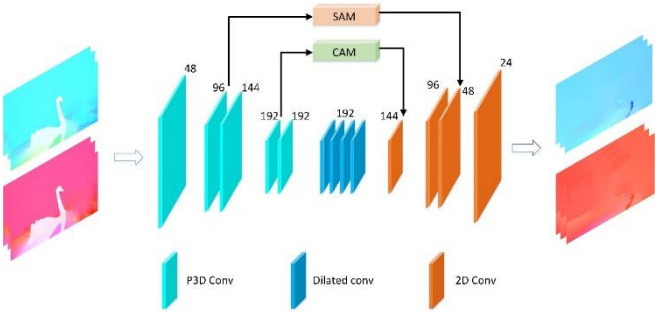


Fig. 8 Structure graph of Local Flow Completion Network

Therefore, in view of [54] using $3 \times 3 \times 3$ to simulate the method of 3D convolution. This paper proposes LFCN, as shown in Fig. 8. The encoder of this network combines the spatial-temporal characteristics of local flow through pseudo 3D modules, and uses the Spatial Attention Module (SAM) [55] and the Channel Attention Module (CAM) [56] to collect the relevant information between local flows, the receptive field is expanded while without increasing model parameters by using hole convolution. The spatial attention module and channel attention module are shown in Fig. 9 (a) and Fig. 9 (b) respectively. The method in this paper uses F_i represents the optical flow map of the i th video frame, $F^{f/b}$ represents the forward or backward damaged optical flow, n represents half the size of the local window, and $F^{f/b} = [F^{\overleftarrow{f}}_{b\,i-n\to i-n+1}, \ldots, F^{\overleftarrow{f}}_{b\,i}, \ldots, F^{\overleftarrow{f}}_{b\,i+n-1\to i+n}]$ respectively input into Encoder, and output the optical flow map of feature fusion through pseudo 3D convolution, which can be expressed as formula (4).

$$F_{i+1} = T(S(F_i)) + F_i \quad (4)$$

Where, T represents one-dimensional time convolution and S represents two-dimensional space convolution. In the process of optical flow completion, the method in this paper obtains the fusion features by reducing the resolution of optical flow sequence while keeping the time dimension of optical flow sequence unchanged. Finally, the complete optical flow map $\widetilde{F_{i+1}}$ is obtained from the two-dimensional convolution decoder。

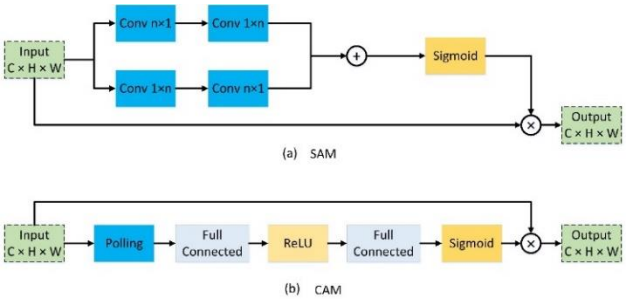


Fig. 9 Spatial Attention Module and Channel Attention Module

In addition, considering that there is no Ground truth after the target of the optical flow map is removed, the method in this paper uses the masks of other videos in the dataset to put into the current video sequence during the training process of LFCN. The only work that the model needs to do is to remove the masks of non-original videos, which solves the problem of no Ground truth. The method in this paper uses L_1. The quality of loss supervision and completion is eliminated by the consistency test of the Ground truth, as shown in formula (5).

$$L_1 = \|F_i - \widetilde{F_i}\|_1 \quad (5)$$

3.4. Spatial-Temporal Transformer

After obtaining the complete optical flow, the method in this paper spreads the optical flow track information from the valid region to the damaged region of the local video sequence, and fills the damaged region. At the end of this process, in view of the article of Zeng et al. [34], this paper designs a spatial-temporal transformer. The difference is that the method in this paper takes the damaged frame as input and adds the roughly inpainted local frame, so that the attention mechanism in the spatial-temporal transformer can better capture the information of neighboring frames and serve as a reference. Since the spatial-temporal complementary information of the complete optical flow map contains the position relationship between the foreground object and the background region, the method in this paper first uses formula (6) to calculate the information $\tilde{F}$ of the complementary optical flow, the damaged frame x and its corresponding mask m carry out the pixel propagation process, and an incomplete inpainted result map can be obtained. Combined with rough inpainted of local reference frame $\tilde{x}$, they are sent into the spatial-temporal Transformer model together, and use the multi-head self-attention mechanism to search information in time and space dimensions. The larger block in the multi-head self-attention module can capture the background information, while the smaller block can capture the information of the moving foreground object. This process is shown in Fig. 10.

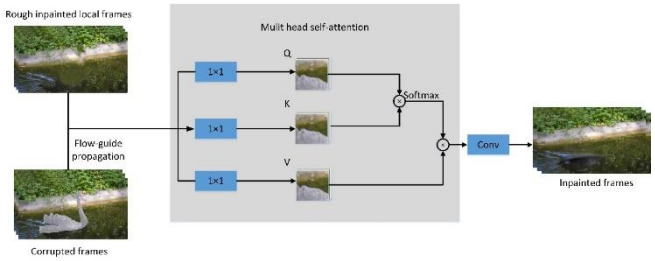


Fig. 10 Structure graph of spatial-temporal attention mechanism

$$\widetilde{F_{i\to j}} = F_n(f_{i_i}, f_j) \tag{6}$$

Where, F_n represents the FlowNet model, f_i and f_j represents two input video frames, $\widetilde{F_{i\to j}}$ represents the extracted optical flow map.

As the core idea in Transformer, the multi-head self-attention module can be divided into three steps: first, through 1 × 1 The convolution layer is used to calculate the weight matrix of each feature, and the formula (7) is used to represent the calculation process.

$$Q_i, K_i, V_i = C_q(x_i), C_k(x_i), C_v(x_i) \tag{7}$$

Where, x_i represents the characteristics of locally damaged frames, $C_q(x_i), C_k(x_i), C_v(x_i)$ is the 1 × 1 Convolution of the query matrix and key-value pair.

Secondly, the method in this paper performs block matching in the attention modules of different heads, and extracts the space block of the shape z1 × z2 × c from the Query and key-value matrices of each frame and the space block is converted into a one-dimensional vector. The similarity between the Query vector of the i th block and the Key vector of the j th block can be calculated using formula (8).

$$S_{i,j} = \frac{P_i^q \times (P_j^k)^T}{\sqrt{z_1 \times z_2 \times c}} \tag{8}$$

Where, P_i^q represents the Query vector of the i th block, P_j^k represents the Key vector of the j th block. The calculation method of the similarity between blocks is to use P_i^q times Transposition of P_j^k. Next, you need to input the Query vector and Key vector into the softmax function to calculate the attachment. As shown in formula (9), use the 1/2 power of the vector dimension as the denominator of the inter-block similarity formula to make $S_{i,j}$ value becomes smaller to avoid the problem of too small gradient of exponential function.

$$Softmax(a_{ij}) = \begin{cases} 0, & if\ is\ a\ damaged\ area \\ \frac{e^{S_{i,j}}}{\sum_m^n e^{S_{i,m}}}, & otherwise \end{cases} \tag{9}$$

Finally, after calculating the attention of all space blocks, use formula (10) to calculate the Value vector of all space blocks.

$$y_i = \sum_{j=1}^{n} a_{i,j} \times P_j^v \tag{10}$$

Where, P_j^v represents the Value vector of the jth block. After the weighted sum of the output of all blocks, the method in this paper splices the output of the attention modules of different heads and reconstructs them into the original size image.

This method uses the Temporal PatchGAN containing 3D convolutions as the discriminator in the model [57-59], which can model the spatial-temporal correlation and perception details of the video sequence, and optimize the adversary loss during the training process. The adversary loss function is shown in formula (11).

$$\begin{cases} L_D = E_{x\sim p_{data}(x)}[\mathrm{ReLU}(1 - D(x))] \\ \quad + E_{z\sim p_y(y)}[ReLU(1 + D(y))] \\ \quad L_{adv} = -E_{z\sim p_y(y)}[D(y)] \end{cases} \tag{11}$$

Where D is discriminator, p_{data} represents the data distribution of Ground truth, p_y represents the result distribution calculated by the spatial-temporal Transformer model.

3.5. Measurement

The evaluation indicators used in this paper include SSIM [60], PSNR [61], LPIPS [62]. Among them, SSIM, PSNR and LPIPS are used to evaluate the image quality.

SSIM: Structural Similarity is an index used to compare the similarity of two input images. SSIM is different from the traditional pixel difference comparison method. It takes into account the perception characteristics of the human visual system to the image [63], so the SSIM

value calculated by different encoding or compression methods for the same image is the same, so SSIM has become a common evaluation index for image quality evaluation and optimization. The core idea of structural similarity is that if two images are similar in structure [64], they are likely to be similar in vision. When calculating SSIM, first divide the original image and the reference image into several sub-blocks of the same size, and then calculate the brightness feature, contrast feature and structure feature of each sub-block. The brightness feature uses the mean value of the image [65, 66], the contrast feature uses the standard deviation of the image, and the structural feature uses the correlation number to measure. Its calculation method is shown in formula (12).

$$\boldsymbol{SSIM}(\boldsymbol{x},\boldsymbol{y}) = \frac{(2\mu_x\mu_y+c_1)(2\sigma_{xy}+c_2)}{(\mu_x^2+\mu_y^2+c_1)(\sigma_x^2+\sigma_y^2+c_2)} \quad (12)$$

Where, assuming that x and y represent two images to be compared, then μ_x is the pixel average of image x, μ_y is the pixel average of image y, σ_x and σ_y represents the variance of pixel values in image x and image y, respectively, σ_{xy} is the covariance of the pixel values in the two images. c_1 and c_2 is two constants.

When calculating SSIM, also need to define a window size, usually is 11 × 11 or 8 × 8. For each window, SSIM value can be calculated. The SSIM value of the entire image can be obtained by averaging the SSIM values of all windows. In addition, the value range of structural similarity should be (0, 1). The larger the calculated value, the more similar the two images are. When the same image is used as x and y for calculation, the calculation result of SSIM will be equal to 1, which means that the two images are identical. Structural similarity can be applied to many fields, such as image compression, image quality evaluation. Compared with other image similarity measurement methods, the structure similarity performance takes into account the human visual system's perception characteristics of image structure, so it more accurately reflects the similarity between images.

PSNR: Peak Signal to Noise Ratio is generally used to compare the difference between an image and its compressed image. It is a widely used indicator to evaluate noise level or image distortion. The larger the PSNR value calculated from the two input images, the less the distortion, and the higher the quality of the generated image. PSNR mainly considers the Mean Square Error (MSE) and Peak Signal strength of the image, and its calculation method is shown in formula (13).

$$\boldsymbol{PSNR} = \boldsymbol{10log_{10}}(\frac{(2^n-1)^2}{\boldsymbol{MSE}}) \quad (13)$$

$$\boldsymbol{MSE} = \frac{1}{\boldsymbol{mn}}\sum_{i=1}^{n}\sum_{j=1}^{m}\|\boldsymbol{K(i,j)} - \boldsymbol{I(i,j)}\|^2 \quad (14)$$

Among them, $(2^n - 1)$ represents the maximum pixel value of the image, which is usually 255 for the 8-bit image, MSE represents the mean square error between two images, and PSNR is in dB, which usually ranges from 20dB to 50dB. The calculation method of MSE is shown in formula (14), where m and n are the total number of pixels of image K and image I, $K(i,j)$ is the pixel value of the original image, and $I(i,j)$ is the pixel value of the processed image.

PSNR has the advantages of simple method and fast calculation speed. However, the disadvantage of PSNR is that it only considers the difference of pixel level, and ignores the perceptual characteristics of the image. There may be errors when evaluating the image quality. In addition, PSNR is sensitive to changes in image brightness and contrast, and can't be applied to all types of images. Therefore, in practical applications, it's usually necessary to use multiple indicators to evaluate image quality.

LPIPS: Learned Perceptual Image Patch Similarity is an image similarity measurement method based on deep learning, also known as perceptual loss, which can be used to measure the similarity of two images under human visual perception. LPIPS focuses on the perception difference of images, which can better simulate the perception process of human visual system. The calculation of this index is based on the deep learning model, which can simulate the perception characteristics of human visual system to images. Specifically, the calculation of LPIPS first needs to preprocess the comparison image and reference image. After scaling to the same size, the two images are divided into image blocks of the same size, and extract the features of each block through the model, and then calculate the similarity scores between different blocks, weighted average the similarity scores of all blocks, and finally obtain the similarity score of the whole image. Its calculation method is shown in formula (15).

$$\boldsymbol{d}(\boldsymbol{x},\boldsymbol{x_0}) = \sum_l \frac{1}{H_l W_l}\sum_{h,w}\|\boldsymbol{d_l} \odot (\hat{\boldsymbol{y}}^{\boldsymbol{l}}_{\boldsymbol{hw}} - \hat{\boldsymbol{y}}^{\boldsymbol{l}}_{\boldsymbol{0hw}})\|_2^2 \quad (15)$$

Where x and x_0 represents the image and reference image for which the index needs to be calculated. The feature vector is extracted from the channel layer l of the deep neural network and conduct unit normalization, and the result is recorded as $\hat{y}^l, \hat{y}_0^l \in R^{H_l \times W_l \times C_l}$。 d represents the distance measurement function between two feature vectors, point multiplication using vector $d_l \in R^{Cl}$ and feature vector, and l_2 is calculated distance, finally calculate the average value in space, sum on the channel, and calculate the distance between x and x_0. h and w are two modules to calculate cross entropy loss and L2 norm respectively. h and w represent the total number of image blocks. The LPIPS indicator can evaluate the perceived similarity between the generated image and the real image. Its value range is [0,1]. The lower the value, the smaller the difference between the images.

It should be noted that the calculation of LPIPS requires a pretrained deep neural network model, which usually uses the model trained by a large amount of data. In addition, the calculation speed of LPIPS is relatively slow, and it also has high requirements for computing resources. There may be errors in the similarity calculation of image blocks of different colors. However, since it can better simulate the perception process of human visual system, it performs well in some image processing tasks.

4. Experiment Result and Discussions

4.1. Qualitative assessment

In order to evaluate the performance of the proposed inpainting algorithm, this paper conducts qualitative analysis on the DAVIS-2017 dataset and compares the results of various inpainting methods. The inpainted results are shown in Fig. 11, and Fig. 11 (a) represents some input video frames. The method in this paper marks and annotates the target to be removed in the original video to achieve the purpose of visual mask. In the inpainted result of three different videos, the target to be removed can be seen to disappear without blur and residue. In the flamingo video, the reflection of the target can still be seen in the water, because the underwater target reflection isn't the region for target removal. In addition, it can be seen from Fig. 11 (b) that after removing the target in the video, the STTN algorithm still has the target artifact and even blurs. This can be seen from the tennis player video in Fig. 11 (b) that the font on the blue sign behind the tennis player becomes blurred. Fig. 11 (c) and Fig. 11 (d) show that the FGVC and the algorithm in this paper have roughly the same effect in the target removal task, but in the details of the flamingo video in Fig. 11 (c), it can be seen that after the target flamingo is removed, there are fuzzy artifacts in small regions.

(a) Input (b) STTN (c) FGVC (d) Ours

Fig. 11 Figure of inpainted results on DAVIS-2017 dataset

4.2. Qualitative assessment

This paper uses the PSNR, SSIM, and LPIPS indicators to verify the effectiveness of the method in this paper. It performs the video target removal task with the advanced inpainting algorithm on the DAVIS-2017 dataset. From the comparison results of the indicators in Table 2, we can see that the PSNR indicators of the method in this paper are 1.2 higher than the best inpainting method in this indicator. The difference between SSIM index and the best inpainting method on this index is 0.007, and the performance on LPIPS index is the best among many inpainting algorithms. In addition, the effectiveness of the method in this paper can be seen from the visualization results in Fig. 11. The indicator values in Table 2 are the average values of indicators obtained from the test of multiple video data in the dataset.

Table 2 Comparison with other inpainting algorithms on DAVIS-2017 dataset

Method	PSNR ↑	SSIM ↑	LPIPS ↓
FGVC[35]	35.310	0.970	0.026

VINet[31]	29.263	0.943	0.052
Lee et al [67]	31.579	0.960	0.044
LGTSM[58]	29.738	0.950	0.060
STTN[34]	31.687	0.950	0.025
DFC-Net[32]	29.671	0.989	0.035
FGT[36]	34.965	0.966	0.029
E2FGVI[68]	33.010	0.972	0.026
Ours	36.130	0.982	0.025

4.3. Ablation experiment of LSTN

The LSTN can combine the reference frame information in the local time window and propagate RGB pixels through the reference frame information to roughly inpaint the video frame. In order to verify the effectiveness of the LSTN proposed in this paper, the LSTN is removed and the residual network is used for rough inpainting. As can be seen in Table 3, the indicators using residual network show lower PSNR and SSIM indicators than LSTN. The indicator values in Fig. 12. are the average values of the indicators obtained from the test of multiple video data in the dataset. The visualization result of the ablation experiment is shown in Fig. 12. It can be seen that after the residual network is used to replace LSTN in the video target removal task, although continuous texture features are obtained and there is no blurred part, there is still color residue of the target in the middle region of the stick (the enlarged yellow box in Fig. 12 (a)), The method in this paper obtains a background without color difference, there is no color residue, and high-quality target removal results are achieved.

(a) residual network (b) LSTN

Fig. 12 Comparison of result of residual network and LSTN

Table 3 Ablation experiment of LSTN

	PSNR↑	LPIPS↓	SSIM↑
Residual Network	34.972	0.025	0.977
LSTN(Ours)	36.130	0.025	0.982

4.4. Ablation experiment of spatial attention module and channel attention module

The spatial attention module in the LFCN can effectively collect the spatial complementary information of the local flow, and the channel attention module can effectively obtain the displacement information of the optical flow map in the horizontal and vertical directions. In order to test the effectiveness of the application of the spatial attention module and the channel attention module in the LFCN, this paper carried out the ablation experiment of removing the spatial attention module and the channel attention module. As can be seen in Table 4, the use of the channel attention module and the spatial attention module shows lower PSNR and SSIM indicators and higher LPIPS indicators than the use of the spatial and channel attention modules. The indicator values in Table 4 are the average values of the indicators obtained from the test of multiple video data in the dataset.

Table 4 Ablation experiment of Spatial Attention Module and Channel Attention Module

	PSNR↑	LPIPS↓	SSIM↑
Without SAM and CAM	35.249	0.026	0.960
With SAM and CAM (Ours)	36.130	0.025	0.982

In addition, the comparison of the results of the ablation experiment is shown in Fig. 13. It can be seen that when the channel attention module and the spatial attention module are removed, the underwater reflection part in the inpainted result image has a small region of distortion (the enlarged yellow box in Fig. 13 (a)), and the reflection region is filled with pixels by the model as the water surface. As can be seen in (Fig. 13 (b)), the method of using spatial attention module and channel attention module can obtain continuous, smooth and reasonable results, and the reflection region is continuously filled.

(a) Without SAM and CAM (b) With SAM and CAM (Ours)

Fig. 13 Comparison of results of with and Without Spatial Attention Module and Spatial Attention Module

4.5. Ablation experiment of flow-guided propagation

After obtaining the complete optical flow map, the damaged frame is inpainted through the optical flow track guided propagation, and then the local damaged frame inpainted by the LSTN is input into the spatial-temporal transformer. In order to verify the effectiveness of flow-guided propagation, the ablation experiment of flow-guided propagation was carried out in this paper. As can be seen in Table 5, the method of with flow-guided propagation showed higher PSNR and SSIM indicators and lower LPIPS indicators compared with the removal of optical flow-guided propagation process. The indicator values in Table 5 are average values of the indicators obtained from the test of multiple video data in the dataset.

Table 5 Ablation experiment of flow-guide propagation

	PSNR↑	LPIPS↓	SSIM↑
Without flow-guide propagation	33.773	0.027	0.953
With flow-guide propagation(ours)	36.130	0.025	0.982

In addition, the comparison of the results of the ablation experiment is shown in Fig. 14. It can be seen that although the target in the video can be removed by removing the flow guided pixel propagation process, there is discontinuity in the spatial domain, and the trunk and seaweed parts are partially disappeared and blurred (the orange frame part in Fig. 14 (a)). The method in this paper has obtained non-blurred and reasonable results.

(a) Without flow-guide propagation (b) With flow-guide propagation (Ours)

Fig. 14 Comparison of results of with and Without flow-guide propagation

5. Conclusion

This paper first introduces the research background and significance of video inpainting, then discusses in detail the inpainting algorithm traditional block-based and based on deep neural network through relevant literature. Finally, the research status of video inpainting algorithms at home and abroad is analyzed, and it's found that the current video inpainting algorithms based on deep neural network still have problems. Most methods may fail to predict optical flow in complex scenes, because they don't perfectly combine the spatial-temporal information between local frames and local optical flow [69]. Therefore, this paper proposes Transformer-Guided Video Inpainting Algorithm Based on Local Spatial-Temporal joint. The algorithm uses local time window to obtain a local frame sequence, uses RAFT as the deep flow network for local optical flow prediction, and completes the optical flow through the LFCN with channel attention module and spatial attention module. In addition, the algorithm designs a LSTN to calculate the rough inpainted results of local frame sequences in parallel. Finally, the rough inpainted results, the complete optical flow map with local spatial-temporal complementary information and the damaged frame are sent to the spatial-temporal transformer with multi-head self-attention mechanism to get the fine inpainted video frame. The experimental results show that the algorithm can ensure the high-quality optical flow map of the model.

The main work of this paper is to capture the algorithm of high-quality optical flow from the video target removal task, but the video inpainting algorithm based on the deep flow network needs more research and exploration, so we intend to carry out further exploration from the following aspects: in the Transformer-Guided Video Inpainting Algorithm Based on Local Spatial-Temporal joint, although it solves the problem of defects in the optical flow completion process of video target fast moving, an effective optical flow map can be obtained, but in scenes with large changes in video background color and displacement, the rough inpainted results will appear large region of blur. Therefore, this paper needs to further explore the video target removal algorithm in complex background.

References

[1] A. S. Al Saadi, "Review on deep neural networks of video inpainting," in *AIP Conference Proceedings*, 2022, vol. 2398, no. 1: AIP Publishing LLC, p. 050017.

[2] Y. Zhang, "Color Image Enhancement based on HVS and PCNN," *SCIENCE CHINA Information Sciences*, vol. 53, no. 10, pp. 1963-1976, 2010.

[3] C. Li, C. Guo, and C. C. Loy, "Learning to enhance low-light image via zero-reference deep curve estimation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 8, pp. 4225-4238, 2021.

[4] C. Tian, L. Fei, W. Zheng, Y. Xu, W. Zuo, and C.-W. Lin, "Deep learning on image denoising: An overview," *Neural Networks*, vol. 131, pp. 251-275, 2020.

[5] S. Fadnavis, "Image interpolation techniques in digital image processing: an overview," *International Journal of Engineering Research and Applications*, vol. 4, no. 10, pp. 70-73, 2014.

[6] H. Ji, C. Liu, Z. Shen, and Y. Xu, "Robust video denoising using low rank matrix completion," in *2010 IEEE computer society conference on computer vision and pattern recognition*, 2010: IEEE, pp. 1791-1798.

[7] Y. Wexler, E. Shechtman, and M. Irani, "Space-time video completion," in *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, 2004, vol. 1: IEEE, pp. I-I.

[8] J. Shayan, S. M. Abdullah, and S. Karamizadeh, "An overview of objectionable image detection," in *2015 International Symposium on Technology Management and Emerging Technologies (ISTMET)*, 2015: IEEE, pp. 396-400.
[9] S. Yuheng and Y. Hao, "Image segmentation algorithms overview," *arXiv preprint arXiv:1707.02051*, 2017.
[10] C. Guillemot and O. Le Meur, "Image inpainting: Overview and recent advances," *IEEE signal processing magazine*, vol. 31, no. 1, pp. 127-144, 2013.
[11] G. Emile-Male, "The restorer's bandbook of easel painting," 1976.
[12] M. Bertalmio, G. Sapiro, V. Caselles, and C. Ballester, "Image inpainting," in *Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, 2000, pp. 417-424.
[13] H. Tang, G. Geng, and M. zhou, "Application of digital processing in relic image restoration design," *Sensing and Imaging*, vol. 21, pp. 1-10, 2020.
[14] S. Setty and U. Mudenagudi, "Region of interest-based 3D inpainting of cultural heritage artifacts," *Journal on Computing and Cultural Heritage (JOCCH)*, vol. 11, no. 2, pp. 1-21, 2018.
[15] Y. Hirohashi, K. Narioka, M. Suganuma, X. Liu, Y. Tamatsu, and T. Okatani, "Removal of image obstacles for vehicle-mounted surrounding monitoring cameras by real-time video inpainting," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2020, pp. 214-215.
[16] J. Zhang, T. Fukuda, and N. Yabuki, "Automatic object removal with obstructed fa çades completion using semantic segmentation and generative adversarial inpainting," *IEEE Access*, vol. 9, pp. 117486-117495, 2021.
[17] B. Bešić and A. Valada, "Dynamic object removal and spatio-temporal RGB-D inpainting via geometry-aware adversarial learning," *IEEE Transactions on Intelligent Vehicles*, vol. 7, no. 2, pp. 170-185, 2022.
[18] Y.-L. Chang, Z. Yu Liu, and W. Hsu, "Vornet: Spatio-temporally consistent video inpainting for object removal," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2019, pp. 0-0.
[19] S. Yuan, Y. Chen, H. Huo, and L. Zhu, "Analysis and synthesis of traffic scenes from road image sequences," *Sensors*, vol. 20, no. 23, p. 6939, 2020.
[20] J. Chen, S. Zhang, X. Chen, Q. Jiang, H. Huang, and C. Gu, "Learning Traffic as Videos: A Spatio-Temporal VAE Approach for Traffic Data Imputation," in *Artificial Neural Networks and Machine Learning–ICANN 2021: 30th International Conference on Artificial Neural Networks, Bratislava, Slovakia, September 14–17, 2021, Proceedings, Part V 30*, 2021: Springer, pp. 615-627.
[21] W. An, X. Zhang, H. Wu, W. Zhang, Y. Du, and J. Sun, "LPIN: A Lightweight Progressive Inpainting Network for Improving the Robustness of Remote Sensing Images Scene Classification," *Remote Sensing*, vol. 14, no. 1, p. 53, 2021.
[22] A. Kuznetsov and M. Gashnikov, "Remote sensing image inpainting with generative adversarial networks," in *2020 8th International Symposium on Digital Forensics and Security (ISDFS)*, 2020: IEEE, pp. 1-6.
[23] M. Bertalmio, L. Vese, G. Sapiro, and S. Osher, "Simultaneous structure and texture image inpainting," *IEEE transactions on image processing*, vol. 12, no. 8, pp. 882-889, 2003.
[24] S. Darabi, E. Shechtman, C. Barnes, D. B. Goldman, and P. Sen, "Image melding: Combining inconsistent images using patch-based synthesis," *ACM Transactions on graphics (TOG)*, vol. 31, no. 4, pp. 1-10, 2012.
[25] T. Shiratori, Y. Matsushita, X. Tang, and S. B. Kang, "Video completion by motion field transfer," in *2006 IEEE computer society conference on computer vision and pattern recognition (CVPR'06)*, 2006, vol. 1: IEEE, pp. 411-418.
[26] Y. Wexler, E. Shechtman, and M. Irani, "Space-time completion of video," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 29, no. 3, pp. 463-476, 2007.
[27] M. Strobel, J. Diebold, and D. Cremers, "Flow and color inpainting for video completion," in *Pattern Recognition: 36th German Conference, GCPR 2014, Münster, Germany, September 2-5, 2014, Proceedings 36*, 2014: Springer, pp. 293-304.
[28] A. Criminisi, P. P érez, and K. Toyama, "Region filling and object removal by exemplar-based image inpainting," *IEEE Transactions on image processing*, vol. 13, no. 9, pp. 1200-1212, 2004.
[29] A. Newson, A. Almansa, M. Fradet, Y. Gousseau, and P. P érez, "Video inpainting of complex scenes," *Siam journal on imaging sciences*, vol. 7, no. 4, pp. 1993-2019, 2014.
[30] C. Wang, H. Huang, X. Han, and J. Wang, "Video inpainting by jointly learning temporal structure and spatial details," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2019, vol. 33, no. 01, pp. 5232-5239.
[31] D. Kim, S. Woo, J.-Y. Lee, and I. S. Kweon, "Deep video inpainting," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 5792-5801.
[32] R. Xu, X. Li, B. Zhou, and C. C. Loy, "Deep flow-guided video inpainting," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3723-3732.
[33] R. Murase, Y. Zhang, and T. Okatani, "Video-rate video inpainting," in *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2019: IEEE, pp. 1553-1561.
[34] Y. Zeng, J. Fu, and H. Chao, "Learning joint spatial-temporal transformations for video inpainting," in *European Conference on Computer Vision*, 2020: Springer, pp. 528-543.
[35] C. Gao, A. Saraf, J.-B. Huang, and J. Kopf, "Flow-edge guided video completion," in *European Conference on Computer Vision*, 2020: Springer, pp. 713-729.
[36] K. Zhang, J. Fu, and D. Liu, "Flow-guided transformer for video inpainting," in *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XVIII*, 2022: Springer, pp. 74-90.
[37] J. Pont-Tuset, F. Perazzi, S. Caelles, P. Arbel áez, A. Sorkine-Hornung, and L. Van Gool, "The 2017 davis challenge on video object segmentation," *arXiv preprint arXiv:1704.00675*, 2017.
[38] N. Xu *et al.*, "Youtube-vos: A large-scale video object segmentation benchmark," *arXiv preprint arXiv:1809.03327*, 2018.
[39] A. Dosovitskiy *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.
[40] S. Bhojanapalli, A. Chakrabarti, D. Glasner, D. Li, T. Unterthiner, and A. Veit, "Understanding robustness of transformers for image classification," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 10231-10241.
[41] H. Fan *et al.*, "Multiscale vision transformers," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 6824-6835.
[42] H. Wu *et al.*, "Cvt: Introducing convolutions to vision transformers," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 22-31.
[43] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16*, 2020: Springer, pp. 213-229.
[44] I. Misra, R. Girdhar, and A. Joulin, "An end-to-end transformer model for 3d object detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 2906-2917.
[45] X. Wang *et al.*, "Oadtr: Online action detection with transformers," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 7565-7575.
[46] H. Wang, Y. Zhu, H. Adam, A. Yuille, and L.-C. Chen, "Max-deeplab: End-to-end panoptic segmentation with mask transformers," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 5463-5474.
[47] K. Zhang, J. Fu, and D. Liu, "Inertia-guided flow completion and style fusion for video inpainting," in *Proceedings of the*

IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 5982-5991.

[48] J. Ren, Q. Zheng, Y. Zhao, X. Xu, and C. Li, "Dlformer: Discrete latent transformer for video inpainting," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 3511-3520.

[49] L. Ke, Y.-W. Tai, and C.-K. Tang, "Occlusion-aware video object inpainting," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 14468-14478.

[50] Z. Wu, C. Sun, H. Xuan, K. Zhang, and Y. Yan, "Divide-and-Conquer Completion Network for Video Inpainting," *IEEE Transactions on Circuits and Systems for Video Technology*, 2022.

[51] J. Kang, S. W. Oh, and S. J. Kim, "Error compensation framework for flow-guided video inpainting," in *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XV*, 2022: Springer, pp. 375-390.

[52] J. Wang, "A review on extreme learning machine," *Multimedia Tools and Applications*, Accessed on: 2021/05/22. doi: 10.1007/s11042-021-11007-7 [Online]. Available: https://doi.org/10.1007/s11042-021-11007-7

[53] J. Wang, "A Review of Deep Learning on Medical Image Analysis," *Mobile Netw. Appl.*, vol. 26, no. 1, pp. 351-380, Feb 2021.

[54] Z. Qiu, T. Yao, and T. Mei, "Learning spatio-temporal representation with pseudo-3d residual networks," in *proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 5533-5541.

[55] M. Jaderberg, K. Simonyan, and A. Zisserman, "Spatial transformer networks," *Advances in neural information processing systems*, vol. 28, 2015.

[56] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7132-7141.

[57] Y.-L. Chang, Z. Y. Liu, K.-Y. Lee, and W. Hsu, "Free-form video inpainting with 3d gated convolution and temporal patchgan," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 9066-9075.

[58] Y.-L. Chang, Z. Y. Liu, K.-Y. Lee, and W. Hsu, "Learnable gated temporal shift module for deep video inpainting," *arXiv preprint arXiv:1907.01131*, 2019.

[59] Y. Zeng, J. Fu, H. Chao, and B. Guo, "Learning pyramid-context encoder network for high-quality image inpainting," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 1486-1494.

[60] J. Nilsson and T. Akenine-Möller, "Understanding ssim," *arXiv preprint arXiv:2006.13846*, 2020.

[61] Q. Huynh-Thu and M. Ghanbari, "Scope of validity of PSNR in image/video quality assessment," *Electronics letters*, vol. 44, no. 13, pp. 800-801, 2008.

[62] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 586-595.

[63] Y. Zhang, "Smart detection on abnormal breasts in digital mammography based on contrast-limited adaptive histogram equalization and chaotic adaptive real-coded biogeography-based optimization," *Simulation*, vol. 92, no. 9, pp. 873-885, September 12, 2016 2016.

[64] Y. Zhang, "Feature Extraction of Brain MRI by Stationary Wavelet Transform and its Applications," *Journal of Biological Systems*, vol. 18, no. S, pp. 115-132, 2010.

[65] S. Wang, "Detection of Alzheimer's Disease by Three-Dimensional Displacement Field Estimation in Structural Magnetic Resonance Imaging," *Journal of Alzheimer's Disease*, vol. 50, no. 1, pp. 233-248, 2016.

[66] S. Wang, "Dual-Tree Complex Wavelet Transform and Twin Support Vector Machine for Pathological Brain Detection," *Applied Sciences*, vol. 6, no. 6, 2016, Art no. 169.

[67] S. Lee, S. W. Oh, D. Won, and S. J. Kim, "Copy-and-paste networks for deep video inpainting," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 4413-4421.

[68] Z. Li, C.-Z. Lu, J. Qin, C.-L. Guo, and M.-M. Cheng, "Towards an end-to-end framework for flow-guided video inpainting," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 17562-17571.

[69] S.-H. Wang, "DenseNet-201-Based Deep Neural Network with Composite Learning Factor and Precomputation for Multiple Sclerosis Classification," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 16, no. 2s, p. Article 60, 2020.