Structure-Guided Deep Video Inpainting

***Abstract*—The fundamental challenge in video inpainting is the difficulty of generating video contents with fine details, while keeping spatio-temporal coherence in the missing region. Recent methods focus on synthesizing temporal smooth pixels by exploiting the flow information, while ignoring maintaining the semantic structure coherence between frames. This makes them suffer from over-smooting and blurry contours, which significantly aggravate the visual inpainting quality. To address this issue, we present a novel structure-guided video inpainting approach that introduces temporal structure coherence to improve video inpanting results. In contrast to directly synthesizing the missing pixels, we first complete the edges in the missing regions to well represent scene structure and object shapes via an edge inpainting network with 3D convolutions. Then, we replenish textures using a coarse-to-fine synthesis network with a structure attention module (SAM), under the guidance of the completed edge structure. Specifically, the SAM can model the semantic correlation between video textures and structural edges to generate structure-consistent inpainted video. Besides, the flow information is used as a temporal consistency constraint during both edge inpainting and texture inpainting learning. Consequently, the inpainting results using our approach are visually pleasing with fine details and temporal coherence in low computational cost. Experiments on the YouTubeVOS and DAVIS datasets show that our method obtains state-of-the-art performance in different video inpainting settings.**

***Index Terms*—Video inpainting, Structure guidance, Flow Assistance.**

I. Introduction

Video inpainting aims to recover the missing content of a corrupted video and assist lots of practical applications, *e.g.,* video restoration and watermarking removal. High-quality video inpainting requires not only realistic structures with visual details but also temporal consistency. Though great progress has been made in 2D image inpainting using deep learning techniques [1], [2], directly applying these approaches to each frame individually for video inpainting will lead to flaws, flickers, and jitters due to extra time dimension.

The traditional video inpainting methods are based on patch composition by exploiting complementary information across neighboring frames and compositing visually pleasing content in the missing regions via patches [3], [4], [5]. These methods rely heavily on the hypothesis that the missing content in the corrupted region appears in neighboring frames, which limits their generalization. Recently, deep-learning-based methods achieve great performance improvement. A straightforward solution is to utilize 3D convolution layers to extract spatio-temporal features and predict missing content with smooth motion [6]. To obtain temporally smooth results, optical flow is commonly used by aggregating contextual information from neighboring frames [7], [8], [9]. For example, a deep optical flow completion network is proposed to propagate missing pixels across frames in DFVI [8] . D. Kim *et al.* [7], [9] leverage recurrent feedback to utilize potential information in previous frames. By introducing motion guidance, these methods pay more attention to temporal smoothness; however, structure rationality and object details have not been well explored. Without definite representation and generation of the target structure, these methods tend to produce over-smoothed regions. Similar observations have been obtained in image inpainting [2], [10]. To solve this problem, Foreground-aware Inpainting [2] and Edge-Connect [10] propose to predict object contours or edges as auxiliary information to guide texture synthesis in image inpainting. However, the image inpainting methods cannot be directly adapted to video due to the extra time dimension. This brings the difficulty of simultaneously preserving the detailed structure and temporal coherence in video inpainting.

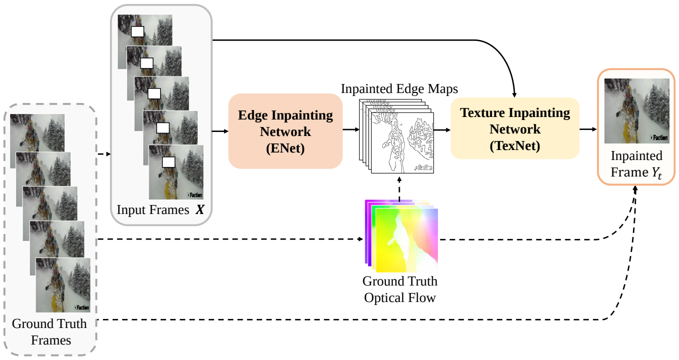
Input Frames X
Inpainted Edge Maps
Edge Inpainting Network (ENet)
Texture Inpainting Network (TexNet)
Inpainted Frame Y_t
Ground Truth Optical Flow
Ground Truth Frames

Fig. 1. Overview of our structure-guided inpainting network. We first complete the missing edges by aggregating information from neighboring frames to represent the target structure using the ENet. Then, the TexNet synthesizes the missing textures under the guidance of structure edges. Besides, the ground truth optical flow is utilized during the training stages of both ENet and TexNet to enforce temporal coherence, which is denoted by dotted lines.

In this paper, we present a novel structure-guided video inpainting approach that effectively exploits the spatio-temporal structure information to improve video inpainting quality. Compared with previous video inpainting methods that only consider the motion information, we explore the correlation among structure, motion, and texture to complete the missing region with valid structure, rich visual details, and temporal coherence. To synthesize the missing content, we first predict sparse edges in the missing region that represent the target structure information, and then fill the textures under the guidance of concise structure. To further enhance the temporal coherence of synthesized frames, we employ motion flows for consistency check of both edge map and inpainted frame generation, during the training stage.

As shown in Fig. 1, our method mainly consists of two modules, which are respectively an edge inpainting network (ENet) and a texture inpainting network (TexNet). Given multiple adjacent frames with masks, ENet first completes

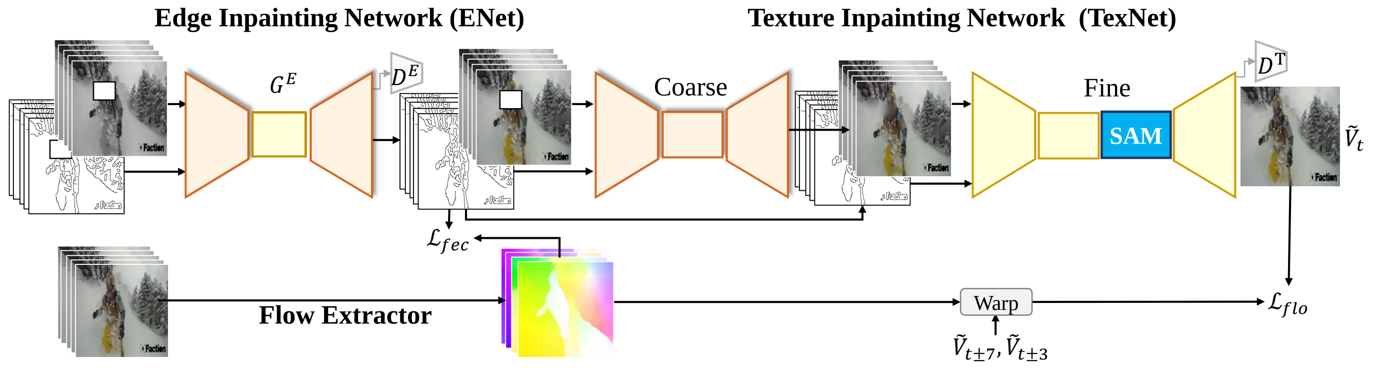
Edge Inpainting Network (ENet)
Texture Inpainting Network (TexNet)
G^E
D^E
Coarse
Fine
SAM
D^T
$\tilde{V}_t$
$\mathcal{L}_{fec}$
Flow Extractor
Warp
$\mathcal{L}_{flo}$
$\tilde{V}_{t\pm7}, \tilde{V}_{t\pm3}$

Fig. 2. The detailed architecture of our network. ENet adopts an encoder-decoder architecture to complete edges in the missing regions. TexNet utilizes a coarse-to-fine manner to inpaint the final frames. Ground truth optical flow between adjacent frames are extracted for edge consistency loss and frame warping loss.

TABLE I

DETAILED COMPARISON BETWEEN OUR METHOD AND RELATED STATE-OF-THE-ART METHODS. OUR METHOD UTILIZES BOTH AUXILIARY TEMPORAL OPTIMIZATION (FLOW) AND SPATIAL OPTIMIZATION (EDGE), WHICH GUARANTEES SPATIO-TEMPORAL COHERENCE EXPLICITLY.

	Temporal Opt.	Spatial Opt.	Explicit Spatio-Temporal Coherence
Edge-Connect [10]	✗	Edge	✗
CombCN [6]	✗	✗	✗
DVI [9]	Flow	✗	✗
DFVI [8]	Flow	✗	✗
Ours	Flow	Edge	✓

edge maps that depict the target structure via a 3D+2D architecture. Then, under the guidance of completed edges, TexNet fills the textures via a coarse-to-fine architecture and a structure attention module (SAM). Specifically, the SAM is designed to guide the texture generation of TexNet by capturing the latent spatial relevance between video textures and completed structure edges. Notably, such a structure-texture relevance can effectively improve the fine-detailed frame generation in TexNet with less cracks and better object contours. Besides, the ground truth optical flow is exploited to guide both ENet and TexNet to generate temporal smooth edge maps and texture results via edge consistency and frame warping constraints. Consequently, the inpainted frames using our approach are not only temporal consistent, but also completed in structure and rich in visual details. The detailed comparison between the proposed method and related state-of-the-art methods is given in Table I.

Experiments on the YouTubeVOS and DAVIS datasets show that the proposed method obtains new state-of-the-art inpainting quality with low time consumption. Our technical contributions can be summarized as follows:

- We propose a novel structure-guided video inpainting method, which explores the correlation among structure, motion, and texture to complete the missing region with valid structure, rich visual details, and temporal coherence.
- A structure attention module is designed to capture the correlation between structure edges and video textures, which can provide better structural guidance for texture synthesis.
- A flow-guided edge warping loss and a frame warping loss are developed to enhance the temporal consistency of both completed edges and video frames.

The rest of this paper is organized as follows. Section II reviews the related works. Section III illustrates the proposed video inpainting method. Section IV provides the experiments on YouTubeVOS and DAVIS datasets, and Section V concludes the whole paper.

II. Related Work

Our method are mostly related to (a) traditional image/video inpainting, (b) CNN-based image inpainting, and (c) deep video inpainting. We will introduce them in this section below.

a) Traditional Image/Video Inpainting: Traditional methods of image and video inpainting can be divided into two categories, diffusion-based and patch-based methods. Diffusion-based methods [11], [12] gradually propagate contents from surrounding areas to the missing region. H. Li *et al.* [13] attempt to solve the problem of localization of diffusion-based inpainted regions. K. Li *et al.*[14] define diffusion coefficients according to the relation between the damaged pixel and neighborhood pixel. Fractional-order nonlinear diffusion driven by difference curvature is proposed to well depict edges [15]. However, this kind of method fails to handle large holes due to its assumption of local smoothness. Patch-based methods, also called exemplar-based methods [16], [17], which are more widely used, formulate the completion task as a patch-based optimization problem. C. Barnes *et al.* [18] use approximate nearest neighbor algorithm to fill the damaged regions. K. Sangeetha *et al.* [19] propose to propagate both linear structure and two-dimensional texture into the target region. T. Ružić *et al.* [20] introduce Markov Random Field (MRF) to help search the most matched candidates. D. Ding *et al.* [21] employ nonlocal texture similarity and local intensity smoothness to produce natural-looking results. Besides, some patch-based methods utilize low rank approximation. For example, Q. Guo *et al.* [22] propose a simple two-stage low rank approximation to recover the corrupted area, which avoids time-consuming iterations. H. Lu *et al.* [23] adopt gradient-based low rank approximation. Patch-based methods fill the missing content by borrowing and aggregating the most similar patches based on low-level image features from known regions and pasting them to unknown parts. However, this type of method will fail when there are insufficient information in known regions or the image textures are too complicated.

As for patch-based video inpainting, a series of methods have been proposed by searching patches across frames [3], extending the 2D PatchMatch algorithm [18] to improve inpainting quality [5], or jointly estimating optical flow and textures to promote temporal coherence [24]. Y. Wexler *et al.* [25] constrain masked regions to synthesize coherent structures with respect to reference examples based on local structures. Y. Umeda *et al.* [26] propose using directional median filter as complementation of patch-based filling. Some methods separate foreground and background apart, and then deal with the two parts respectively with different algorithms, since there naturally exists property differences between them. A. Ghanbari *et al.* [27] first separate the two parts in videos, and fills the two parts accordingly with the help of contours. A. Xia *et al.* [28] make use of Gaussian Mixture Model (GMM) to also distinguish moving foreground and still background, and process them separately.

However, the propagation process makes these methods suffer from high computational complexity, which limits their usage in practical applications.

b) CNN-based Image Inpainting: Recently, deep learning methods have achieved tremendous progress in the filed of computer vision, for example, image recognition [29], [30], object detection [31], [32], and semantic segmentation [33], [34]. The tasks of image and video inpainting also witness great promotion thanks to the capability of deep learning networks to capture high-level semantic information in images and videos. The convolution neural network (CNN) is first introduced for image denoising and inpainting in [35], where CNN is used to directly synthesize image contents in the masked regions. To improve the photorealism of the completed results, a generative adversarial network is employed [36]. Then, C. Yang *et al.* [37] takes advantages of multi-scale representation to boost details generation. Multiple discriminators are used to constrain both global and local coherence of image contents [38]. J. Yu *et al.* [39] proposes the contextual attention module to capture long-range information. Subsequent approaches solve more specific problems in image inpainting. For example, inpainting irregular holes with partial convolution [40] and employing gated convolution [1] for dynamic feature selection. Both these two method want to handle image inpainting with irregular regions, which is hard for vanilla convolutions. While these methods tend to generate over-smoothed and blurry results, a two-stage approach is proposed to hallucinate edges first and then fill image colors using the edges as a prior [10]. Similarly, W. Xiong *et al.* [2] predict contours of foreground objects to guide the inpainting process of masked regions. Though high-quality static images with reasonable structure can be generated using these methods, simply extending it from image to video by 3D convolutions inevitably fails because of no guarantee on the temporal coherence. Besides, these methods simply utilize edges and contours as one of the inputs of the image inpainting network, without the mechanism to utilize the structure information more efficiently.

c) Deep Video Inpainting: Besides the challenge in maintaining temporal coherence, image inpainting methods do not fully utilize useful complementary information in neighboring frames, which could help large hole completion in videos. Several methods regarding video inpainting based on deep neural networks have been proposed just recently. The first deep-learning-based video inpainting method is CombCN [6], which jointly learns temporal structure and spatial details via 3D convolutions. To enforce temporal coherence, a recurrent feedback is employed to connect consecutive frames [7], [9]. Instead of filling pixel colors directly using CNNs, a deep flow completion network is proposed to propagate pixel colors based on the estimated flow in the missing region [8]. However, these existing methods neglect the importance of intact structure in video inpainting and typically suffer from blurs and structural cracks in the synthesized frames. In comparison, we propose to explicitly complete the target structure using edges, which are efficient to predict due to their sparsity. To utilize the information more effectively, we introduce a structure attention mechanism. Under the structural guidance, more pleasing results could be synthesized with reasonable structure and fine details.

III. Approach

The target of our method is to recover the missing contents in a corrupted video with fine details and temporal consistency. In each data batch, total T frames $\boldsymbol{V}$ ($T = 5$), indexed by $\{V_{t-7}, V_{t-3}, V_t, V_{t+3}, V_{t+7}\}$, are fed to our inpainting network, as well as the corresponding masks $\boldsymbol{M}$ that indicate the missing regions. The corresponding ground truth frames are denoted as $\boldsymbol{V}^{gt}$. The final output is the completed frame $\widetilde{V}_t$ at time t.

As Fig. 2 shows, our framework consists of three main components. The first part is an edge inpainting network (ENet) that targets to recover the missing edges, and the second part is a coarse-to-fine texture inpainting network (TexNet) that aims to complete the missing appearance details. The third part leverages the ground truth motion flow as auxiliary constraints to preserve temporal coherence of both completed edge and texture during the training stage.

A. Edge Inpainting Network

Before completing the missing texture in a piece of video, we first predict the sparse edges using an edge inpainting network (ENet). Compared with texture, the edge depicts the object shape and motion, which is much easier to imagine without complex colors. Therefore, under the guidance of well-completed object shape and motion, we can better inpaint the whole texture.

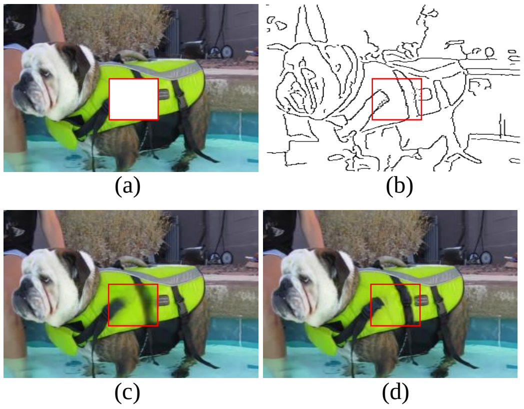

Fig. 3. Given a corrupted frame (a), our ENet first completes sparse edges (b) which well represent the structure of the missing content. Then TexNet progressively replenishes textures under the guidance of synthesized edges from coarse (c) to fine (d).

Let $\boldsymbol{V}^{gt}$ denote ground truth frames, and $\boldsymbol{V}_g^{gt}$ and $\boldsymbol{E}^{gt}$ is corresponding grayscale version of frames and edge maps, respectively. The input of ENet consists of the incomplete grayscale frames $\boldsymbol{V}^g = \boldsymbol{V}_g^{gt} \odot \big(1 - \boldsymbol{M}\big)$, initial edge maps $\boldsymbol{E}^i = \boldsymbol{E}^{gt} \odot \big(1 - \boldsymbol{M}\big)$, and their corresponding masks $\boldsymbol{M}$. As shown in Fig. 2, ENet is denoted by G^E, which is composed of a two-layer 3D encoder, eight 2D residual blocks, and a two-layer 3D decoder. The 3D encoder and decoder are designed to learn the spatio-temporal correlation with the 3-dimensional convolution operations. The intermediate 2D residual blocks are used to enlarge the spatial receptive fields by using 2-dimensional convolution of large kernel size. Such a 3D+2D convolution architecture can achieve a good trade-off between spatial and temporal perception. the inpainted edge maps are obtained by:

$$\widetilde{\boldsymbol{E}} = G^E(\boldsymbol{E}^i, \boldsymbol{V}^g, \boldsymbol{M}). \tag{1}$$

To train the edge generator G^E, ENet plays a minimax game by:

$$\min_{G^E} \max_{D^E} \big(\mathcal{L}_{adv}^E + \lambda_1 * \mathcal{L}_{fm}^E\big), \tag{2}$$

where the discriminator D^E follows the 70 × 70 PatchGAN architecture [41]. $\mathcal{L}_{adv}^E$ and $\mathcal{L}_{fm}^E$ are the adversarial loss and feature matching loss. λ_1 is a hyper-parameter to balance the two terms. In Eq. (2), $\mathcal{L}_{adv}^E$ is an adversarial learning loss to make the predicted edge maps more realistic, which evaluates the image-level similarity between ground truth edge maps and predicted edge maps by:

$$\begin{aligned} \mathcal{L}_{adv}^E = &\mathbb{E}_{(\boldsymbol{E}^{gt}, \boldsymbol{V}^g)}\big[log D^E(\boldsymbol{E}^{gt}, \boldsymbol{V}^g)\big] \\ &+ \mathbb{E}_{(\widetilde{\boldsymbol{E}}, \boldsymbol{V}^g)}\big[log\big(1 - D^E(\widetilde{\boldsymbol{E}}, \boldsymbol{V}^g)\big)\big]. \end{aligned} \tag{3}$$

$\mathcal{L}_{fm}^E$ evaluates the feature-level similarity between ground truth and predicted edge maps, which is defined by:

$$\mathcal{L}_{fm}^E = \sum_{k=1}^{L} \frac{1}{N_k} \Big\| D_k^E(\boldsymbol{E}^{gt}, \boldsymbol{V}^g) - D_k^E(\widetilde{\boldsymbol{E}}, \boldsymbol{V}^g) \Big\|_1, \tag{4}$$

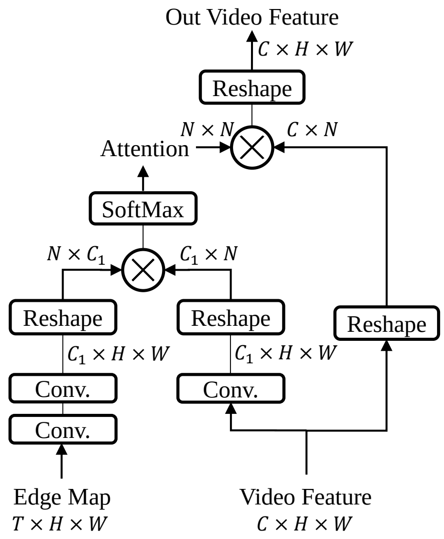

Fig. 4. Architecture of the structure attention module. C is the channel of the input video features, and $N = H \times W$. $\otimes$ represents matrix multiplication. Usually, we set $C_1 = C/8$.

where D_k^E is the output of the k-th layer in L-layer D^E, while N_k is the element number of D_k^E. By considering both feature-level and image-level similarities in Eq. (2), the edge generator G^E can be well-trained to produce plausible and structurally rational edge maps.

Consequently, the 3D+2D architecture and two-level loss function enable the ENet to inpaint the missing structural edges accurately. An example of the generated edges is given in Fig. 3 (a) and (b).

B. Edge-Guided Texture Inpainting Network

With the completed edge maps $\widetilde{\boldsymbol{E}}$ for the T frames $\boldsymbol{V}$, we then fill the image texture using a coarse-to-fine network, *i.e.,* TexNet. Notably, the object structure, *e.g.,* object shapes, has been captured in the completed edge frames $\widetilde{\boldsymbol{E}}$. Thus, it becomes much easier to fill the missing texture with the structure guidance of $\widetilde{\boldsymbol{E}}$.

To synthesize realistic frame texture, the proposed TexNet adopts a coarse-to-fine architecture, as shown in Fig. 2. Specifically, TexNet consists of a coarse inpainting network and a refinement network. First, the coarse inpainting network consists of a set of 3D convolutions to capture the spatio-temporal information, which targets to produce a rough completion $\widetilde{\boldsymbol{V}}^i$ for the T frames with colors and textures. Then, the refinement network takes the rough inpainting results $\widetilde{\boldsymbol{V}}^i$ and the synthesized edge maps $\widetilde{\boldsymbol{E}}$ as inputs to further refine the coarse details in $\widetilde{\boldsymbol{V}}^i$ with the guidance of structural edges $\widetilde{\boldsymbol{E}}$. Notably, only 2D convolution is used in refinement network to improve the inference efficiency.

Besides taking $\widetilde{\boldsymbol{E}}$ as an auxiliary input in refinement network, we further design a structure attention module (SAM) to fully encode the structural information. The detailed implementation of SAM is given in Fig. 4, which is used as an intermediate layer in the refinement network in Fig. 2. The inputs of SAM are the intermediate video features from $\widetilde{\boldsymbol{V}}^i$ and embedded edge features from $\widetilde{\boldsymbol{E}}$, and the output is a matrix that indicates the relationship among different spatial features. First, the intermediate video features and embedded edge features are interacted to calculate the latent structure-texture correlation via matrix multiplication. After a SoftMax operation, the normalized attention map is obtained, which represents correlation between the structure and high-level video features. Then, the normalized attention map is applied to the intermediate video features, and the structure information is thus embedded in TexNet, which can better extract useful structural information brought by edges. After introducing structural guidance, the inpainted content by TexNet becomes more realistic, as Fig. 3 shows.

Specifically, the coarse inpainting network and the refinement network are trained end-to-end by:

$$\min_{G^T} \max_{D^T} \left(\mathcal{L}^T_{rec} + \mathcal{L}^T_{adv} \right), \tag{5}$$

where G^T denotes the TexNet and D^T is the discriminator. Inspired by [10], the first term $\mathcal{L}^T_{rec}$ is the l_1-reconstruction loss to measure the difference between predicted video frames and the ground truth video frames $\boldsymbol{V}^{gt}$. Differently, we penalize both the coarse predictions $\widetilde{\boldsymbol{V}}^i$ and refined frame $\widetilde{V}_t$, given by:

$$\begin{aligned} \mathcal{L}^T_{rec} &= \frac{1}{\|M_t\|_1} \left\| (\widetilde{V}_t - V^{gt}_t) \odot M_t \right\|_1 , \\ &+ \lambda_2 * \frac{1}{\|\boldsymbol{M}\|_1} \left\| (\widetilde{\boldsymbol{V}}^i - \boldsymbol{V}^{gt}) \odot \boldsymbol{M} \right\|_1 , \end{aligned} \tag{6}$$

where V^{gt}_t denotes the ground truth frame at time t, and M_t is the corresponding binary mask. Besides, an extra adversarial loss $\mathcal{L}^T_{adv}$ is introduced in Eq. (5) to promote the visual realism of the generated frame by:

$$\mathcal{L}^T_{adv} = \mathbb{E}[logD^T(V^{gt}_t)] + \mathbb{E}[log\big(1 - D^T(\widetilde{V}_t)\big)]. \tag{7}$$

$\mathcal{L}^T_{adv}$ enforces the generated frame to be more realistic.

From the results in Fig. 3 (c) and (d), it can be seen that the refinement network can exactly refine the inpainting results with more clear contours and textures, and the combined loss of $\mathcal{L}^T_{rec}$ and $\mathcal{L}^T_{adv}$ can well train the coarse and refinement networks jointly.

C. Flow-Guided Temporal Coherence Enhancement

Besides the structure guidance, the motion information is also considered to maintain temporal consistency among frames. To this end, the optical flow is employed in the training stage of both ENet and TexNet.

During training, a set of flow maps $\boldsymbol{O}$ between the current frame V_t and its neighboring frames are generated using a pre-trained flow extraction network, such as FlowNet2.0 [42]. Specifically, $\boldsymbol{O}$ consists of four flow maps $(O_{t\Rightarrow t-7}, O_{t\Rightarrow t-3}, O_{t\Rightarrow t+3}, O_{t\Rightarrow t+7})$. $\boldsymbol{O}$ contains the ground truth motion information between frames, which is important to both edge and texture generation.

In terms of the ENet, *i.e.,* inpainting the missing edges, $\boldsymbol{O}$ is used to first warp the neighboring edge maps to the current frame, and then compute the consistency between neighboring edge maps. Thus, a flow-guided edge consistency loss is defined as:

$$\mathcal{L}^E_{fec} = \sum_k \frac{1}{\|M_t\|_1} \left\| (\widetilde{E}_t - \phi(O_{t\Rightarrow t+k}, \widetilde{E}_{t+k})) \odot M_t \right\|_1 , \tag{8}$$

where $\phi(O_{t\Rightarrow t+k}, \widetilde{E}_{t+k})$ is the warping operation which warps the edge map $\widetilde{E}_{t+k}$ to the target frame according to the generated optical flow $O_{t\Rightarrow t+k}$. k denotes the index of neighboring frames ($k \in \{-7, -3, +3, +7\}$). With the flow-guided edge consistency loss $\mathcal{L}^E_{fec}$, the loss function of Eq. (2) for ENet becomes:

$$\min_{G^E} \max_{D^E} \left(\mathcal{L}^E_{adv} + \lambda_1 * \mathcal{L}^E_{fm} + \mathcal{L}^E_{fec} \right). \tag{9}$$

About the TexNet, we further enforce the temporal coherence of synthesized neighboring textures via a flow warping constraint $\mathcal{L}^T_{flo}$ by:

$$\mathcal{L}^T_{flo} = \sum_k \frac{1}{\|M_t\|_1} \left\| (\widetilde{V}_t - \phi(O_{t\Rightarrow t+k}, \widetilde{V}_{t+k})) \odot M_t \right\|_1 , \tag{10}$$

where k is still in $\{-7, -3, 3, 7\}$. $\phi(O_{t\Rightarrow t+k}, \widetilde{V}_{t+k})$ warps $\widetilde{V}_{t+k}$ to $\widetilde{V}_t$ using flow $O_{t\Rightarrow t+k}$. Finally, $\mathcal{L}^T_{flo}$ is added to Eq. (5), which becomes:

$$\min_{G^T} \max_{D^T} \left(\mathcal{L}^T_{rec} + \mathcal{L}^T_{adv} + \mathcal{L}^T_{flo} \right), \tag{11}$$

After adding the motion information to both training process of ENet and TexNet, the temporal consistency can be preserved in both edge generation and texture inpainting. Since the motion is only used in the training stage, the inference of the proposed structure-guided inpainting method is efficient and effective.

IV. Experiments

To evaluate the effectiveness of our approach, we conduct a series of comparison experiments and ablation studies on two widely used datasets, *i.e.,* YouTubeVOS [43] and DAVIS [44], under different settings.

A. Experimental Settings

Mask Setting. Considering different real-world applications, we test four kinds of mask settings in this paper, which are different in shapes and positions of the missing regions.

- Fixed square mask: The size and position of the missing square region are fixed through the whole video.
- Moving square mask: The position and size of the square mask change over frames.
- Free-from mask: We apply irregular masks which imitate hand-drawn masks on each frame, following [40].
- Foreground object mask: This type of mask is defined to line out foreground objects in videos and used for testing object removal.

Dataset. YouTubeVOS and DAVIS are widely used for evaluating video inpainting results in recent studies. YouTubeVOS consists of 4,453 video clips that contain more than 70 categories of common objects. The videos are split into three

TABLE II
COMPARISONS WITH FOUR STATE-OF-THE-ART METHODS PROPOSED IN 2019 ON YOUTUBEVOS. OUR METHOD OUTPERFORM ALL OTHER METHODS ON THREE METRICS, WITH FAST INFERENCE SPEED.

	Fixed Square Mask			Moving Square Mask			Free-Form Mask			Inference
	PSNR	SSIM	FID	PSNR	SSIM	FID	PSNR	SSIM	FID	Speed (fps)
Edge-Connect [10]	28.6446	0.9484	38.2116	30.7478	0.9647	16.2739	25.6693	0.9088	43.0366	22.81
CombCN [6]	27.9668	0.9515	40.7199	31.5776	0.9678	13.8383	32.1862	0.9626	19.1191	8.1634
DVI [9]	28.0846	0.9468	39.9377	36.8598	0.9728	7.2315	33.5549	0.9646	9.3797	1.2275
DFVI [8]	29.0531	0.9497	32.8860	37.8241	0.9772	6.3746	32.6287	0.9618	11.1501	0.5620
Ours	**30.0590**	**0.9543**	**27.2431**	**38.8186**	**0.9824**	**2.3455**	**35.9613**	**0.9721**	**5.8694**	5.1546

parts, 3,471 for training, 474 for validating, and 508 for testing. Since YoutubeVOS has no dense foreground mask annotations, we only use it for evaluation of mask settings (1), (2), and (3). DAVIS dataset contains 90 video sequences that are annotated with foreground object masks and 60 unlabeled videos for training.

Implementation details and Evaluation Metrics. All the models are test on a TITAN X (Pascal) GPU with frame size 256 × 256. For training data, We randomly sample a training clip every 40 frames from each video in the dataset. Our training consists of three steps. First, we train ENet with learning rate set as $1e-4$ for G^E and $1e-5$ for D^E. Then we train TexNet while fixing ENet. Learning rate is set as $1e-4$ for G^T, and $4e-4$ for D^T. We first train ENet and TexNet without corresponding flow losses, and then add the flow losses to refine the final results when flow assistance is utilized. Adam optimizer with $\beta = (0.9, 0.999)$ is used for all sub-networks. We do not use weight decay in training. As for the hyper-parameters, $\lambda_1 = 10.0, \lambda_2 = 0.2$.

Different data preparations and evaluation metrics are used according to mask settings. a) We randomly generate masks for training videos in terms of mask settings (1), (2), and (3). Masked videos are used for testing. Three commonly-used metrics, including structural similarity index (SSIM) [45], peak signal-to-noise ratio (PSNR), and Fréchet Inception Distance (FID) [46] are used to quantitatively evaluate the performance of our method. b) For mask setting (4), we prepare masks of random shapes and motions to synthesize the foreground object masks in the training stage. The network is first trained on the YoutubeVOS dataset and then finetuned on DAVIS. Since there is no ground truth available for this setting, which means that we can not use quantitive evaluations to measure the output quality, we conduct a user study for video foreground object removal.

B. Main Results on Video Inpainting

We compare the proposed method with four state-of-the-art video inpainting methods [10], [6], [9], [8] for the first three mask settings on the YouTubeVOS dataset. We train [10], [8] using their published codes and re-implement [6] according to their paper. As for [9], we use the officially provided model since there are no available training codes.

The quantitative results and inference speeds are reported in Table II. It shows that our method outperforms state-of-the-art methods on the three metrics, demonstrating the effectiveness of introducing structure guidance into video inpainting. Moreover, our method is also very efficient, e.g., four times faster than [9] and nine times faster than [8]. Models have different performances with different masks. The best performance is obtained when using moving square masks, which is because that the network can learn complementary information from neighboring frames. Some inpainting examples are shown in Fig. 5. Compared with existing methods, the inpainting results predicted by our method are more realistic with finer details. We can observe that the frames completed using our method contain sharper object boundaries. This is achieved by the effectiveness of structure information in video inpainting. When looking at two neighboring frames in each video, it can be seen that our method produces more temporally smooth results.

The inference speed of Edge-Connect [10] is fast, the reason of which is that it does not consider axillary temporal information between frames. Comparing to this 2D image inpainting method, which also predicts edges to represent the structure, our method greatly increases the completion performance by leveraging neighboring frames to complete edges and synthesize textures. Thus, our method can generate more temporal coherent and realistic contents.

Compared with the second-best video inpainting method DFVI [8], our method can produce frames with finer structural details. Besides, optical flow is only used in training period in our method, while DFVI requires iterative optimization with optical flow. Thus the inference speed of our method is much faster than that of DFVI.

It demonstrates that our method is not a naive extension to utilize structure information in video inpainting and also indicates that structure clues can bring strong promotion to video inpainting.

C. Results on Object Removal

In regard to the foreground object mask setting that aims to remove undesired objects in videos, there is no ground truth for quantitative evaluation. Therefore, we conduct a user study on the DAVIS dataset to evaluate the visual quality of our method, compared with four methods [10], [6], [9], [8]. In each test, we show three videos to the subject at the same time. The original video with red masks indicating objects to remove is shown in the middle, while the inpainting results of our method and one of the other four methods are shown on the two sides in random order. The subjects can watch the videos repeatedly to better evaluate the differences. For each video

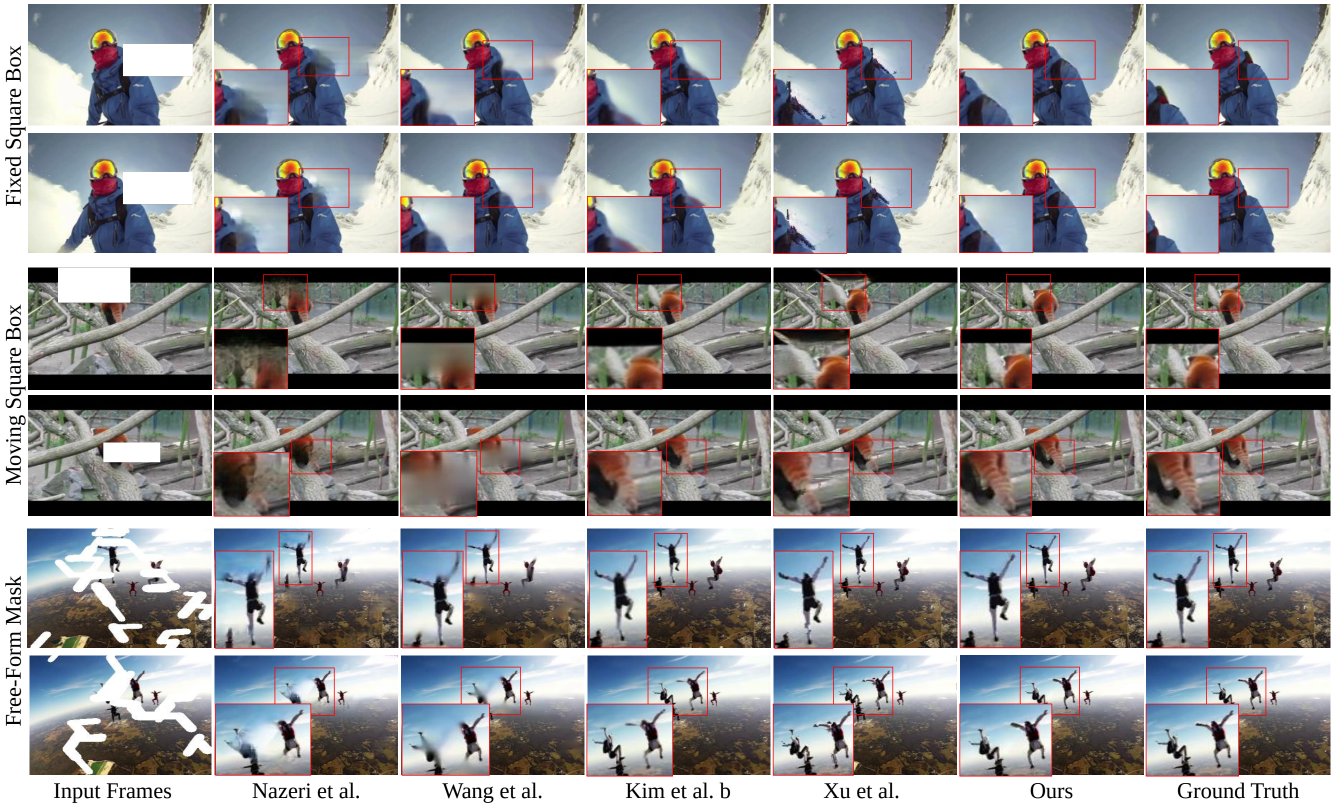

Fig. 5. Visualization comparison on YouTubeVOS with state-of-the-art methods. Our method produces results with more complete object structures and finer details.

triplet, the subject is asked to choose which inpainting video is preferred. 44 subjects participated in our user study. Each participant watched averagely 20 triplets. Therefore, each pair of methods is compared about 220 times.

The preference in the user study is shown in Fig. 7. Comparing to Edge-Connect [10], CombCN [6], DVI [9], our results are preferred by a significantly larger portion of subjects. When comparing with the flow-guided method DFVI [8], our method is preferred by 55.24% of the tests. Notably, our method is much faster than DFVI.

Fig. 6 shows two examples of object removal using different methods. We can see that inpainted results generated by our methods are visually better than existing methods. Compared to the blurry contents in the results of Edge-Connect, CombCN, and DVI, our method produces sharp object boundaries and fine visual details. Notably, other methods Though the completed contents using DFVI have sharp edges, the global structure of the human bodies is corrupted. In comparison, our method achieves more intact and plausible structure with fine details. The results demonstrates the importance of utilizing structure information in video inpainting.

D. Ablation Study

To demonstrate the effectiveness of each component in our network, we conduct a series of ablation studies on the YouTubeVOS dataset with the first three mask settings. We test four variants of our model. The baseline model 'TexNet' only consists of the coarse-to-fine texture inpainting network without using edge maps as input and no SAM in the refinement module. This model simply integrates neighboring frames to predict the missing content for the current frame. Then we add the edge inpainting network and fed the texture inpainting network with the completed edge maps to get the second model '+Edge'. The third model '+SAM' is constructed by adding the structure attention module on the second model. Finally, we add the flow guidance to get our full model 'Ours'. Especially, we only add the flow guidance on training stage, which means it brings no computation costs to test. The quantitative results are reported as in Table III.

1) Effect of Structure Clues: In Table III, '+Edge' brings a large improvement over the baseline model. It indicates that sparse edges can provide effective structural guidance in video inpainting.When we further add SAM to '+Edge', extra improvement is obtained, demonstrating that the spatial correlation between edges and textures can be better embedded and absorbed by the texture inpainting network than simply feeding the completed edge maps as extra channels into TexNet. The above analyses prove that the edge clues are effective guidance in video inpainting, which helps the network to predict more accurate frames. The inpainted edge

TABLE III
ABLATION STUDIES ON YOUTUBEVOS. STRUCTURE INPUT, STRUCTURE ATTENTION MECHANISM, AND FLOW ASSISTANCE ARE DEMONSTRATED EFFECTIVE IN VIDEO INPAINTING.

	Fixed Square Mask			Moving Square Mask			Free-Form Mask			Inference
	PSNR	SSIM	FID	PSNR	SSIM	FID	PSNR	SSIM	FID	Speed (fps)
TexNet	28.0174	0.9494	42.7164	33.8131	0.9705	8.2390	30.0680	0.9390	20.6358	7.6335
+Edge	29.5242	0.9520	36.2097	37.6630	0.9798	3.5161	33.8206	0.9659	6.6651	5.2356
+SAM	29.9918	0.9533	27.4198	38.2433	0.9807	2.5083	35.7783	0.9712	5.8786	5.1546
Ours	**30.0590**	**0.9543**	**27.2431**	**38.8186**	**0.9824**	**2.3455**	**35.9613**	**0.9721**	**5.8694**	5.1546

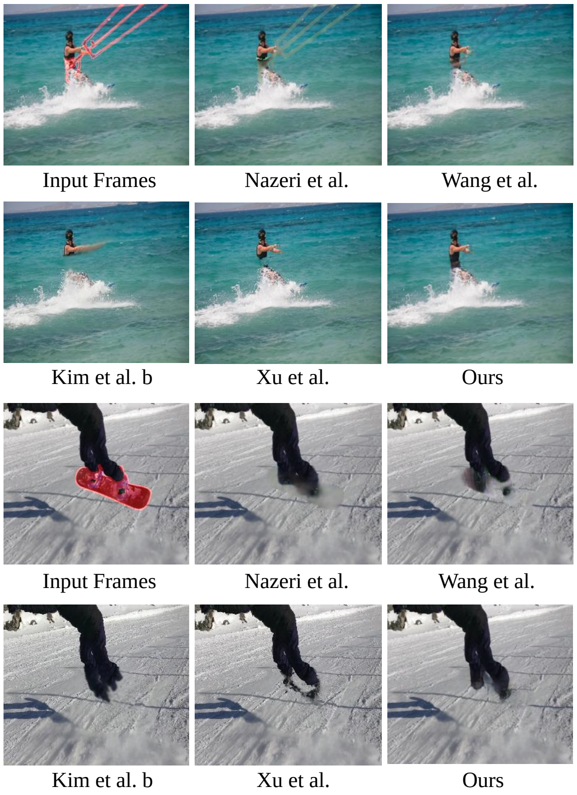

Fig. 6. Results of object foreground removal. The red masks in input frames indicate the objects to be removed. Our method produce results with plausible structures and details.

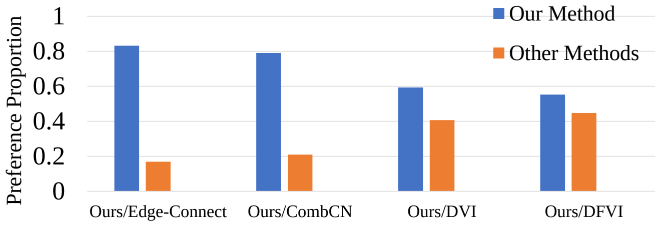

Fig. 7. Results of user study. Ours are preferred by more participants compared to state-of-the-art methods.

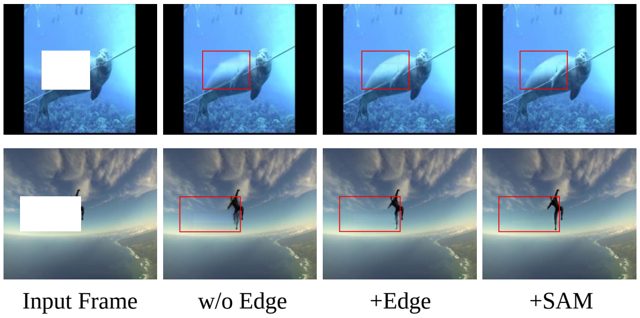

Fig. 8. Effects of structure guidance. By inpainting edge maps first and then filling texture, we generate more completed target structure. Clearer boundaries can be obtained using the structure attention module.

map in Fig. 3 shows that ENet is capable of generating completed and detailed structure. Indeed, the structure module ENet brings extra time consuming to the baseline TexNet from 7.6335 *fps* to 5.2356 *fps*. This is deserved, because the inpainting quality is significantly improved.

Fig. 8 shows the results generated using the three variants. It is obvious that after introducing structure guidance, the inpainted frames become more visually pleasing with sharper object boundaries. Besides, the edge maps predicted by our method are reasonable and clear, which well represent the image structure and show the strong edge inpainting ability of ENet. Thus, it is crucial to explore structural details in video inpainting.

2) Comparison of Proposed Structure Attention Module and Simple Attention: We propose a novel structure attention module (SAM) in TexNet to facilitate exploiting structure information of edge maps more effectively. This module is specifically designed for structural distilling in video inpainting. In this part, we conduct a comparison experiment between the proposed SAM and the commonly used simple 2-layer attention (SimATT) [47], *i.e.,* using a 2-layer convolution network to extract a spatial attention map from predicted edge maps, which is then applied to the same video feature as SAM. The model '+Edge' is used as the baseline. Results are shown in Table IV. The performance of SAM is better than that of SimATT. It is because that the proposed SAM is capable of revealing potential correlation between structure information in edge maps and video contents. Thus it is easier for TexNet to utilize structure information to obtain better results.

3) Effect of Flow for Temporal Coherence: In our method, we utilize temporal information to smoothen artificial flickers via two developed flow-guided warping losses. From Table III, we can see that the quantitative performance is improved on all three settings by adding the flow guidance. Especially, we only

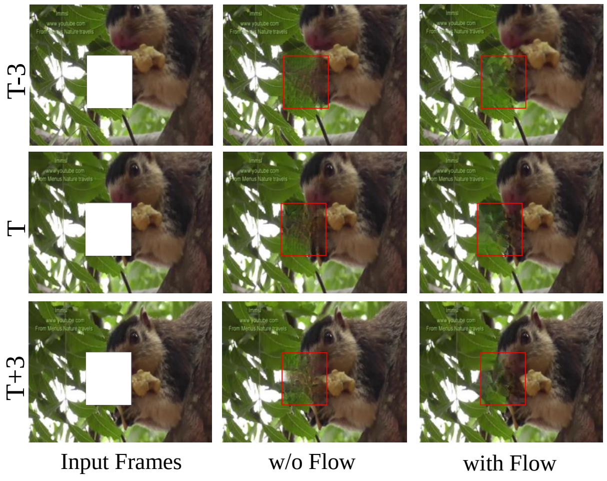

Fig. 9. Inpainting results of three neighboring frames. With the flow assistance, the inpainting results are more temporally consistent without introducing image blurs.

TABLE IV
COMPARISON OF PROPOSED STRUCTURE ATTENTION MODULE (SAM) AND SIMPLE 2-LAYER SPATIAL ATTENTION ON YOUTUBEVOS. SAM IS CAPABLE OF REVEALING POTENTIAL CORRELATION BETWEEN STRUCTURE INFORMATION AND VIDEO CONTENTS.

	Free-Form Mask		
	PSNR	SSIM	FID
+Edge	33.8206	0.9659	6.6651
+SimATT	34.4321	0.9685	6.3125
+SAM	35.7783	0.9712	5.8786

add flow guidance in the training phase, so it can bring gains without extra computation costs during testing. The results show that the improvement from the optical flow is smaller than that from the structure edge. The reason is that the flow is only used in the training stage as a temporal guidance, while the edge is used during the inference. It should be noted that predicting the completed optical flow among several frames takes more time than the edges during inference. Thus, we use the flow in training and the structure edge in both training and testing, which can achieve a good balance between the quality improvement and the inference efficiency.

As shown in Fig. 9, the synthesized contents in neighboring frames become more temporally consistent after adding the flow guidance. This proves that the proposed two flow-guided constraints in edge and texture inpainting network are effective in preserving the temporal consistency.

4) Effects of Different Value of Hyper Parameters: In this part, we conduct experiments to determine the hyper-parameters of λ_1 in Eq. (2) and λ_2 in Eq. (6). We use edge inputs for TexNet in this part, without SAM and flow assistance. When testing λ_1, we first train ENet with different values of λ_1, and then train TexNet with generated edge maps by fixed ENet. The value of λ_2 is set as 0.2. When determining λ_2, we keep the value of λ_1 as 10.0, then train TexNet with different λ_2.

λ_1 is used in Eq. (2), which is used to denote the weight of feature matching loss when training ENet. From the results in Fig. 10, when increasing λ_1 from 0.0 to 2.0, the performance gain is obvious, which proves that the feature matching loss is effective in generating high-quality edge maps used for the final inpainted results. Then when λ_1 is increased from 2.0 to 10.0, slight improvement is obtained. Finally, when $\lambda_1 > 10.0$, the model performance drops. Therefore, we set λ_1 to 10.0 for the experimental settings.

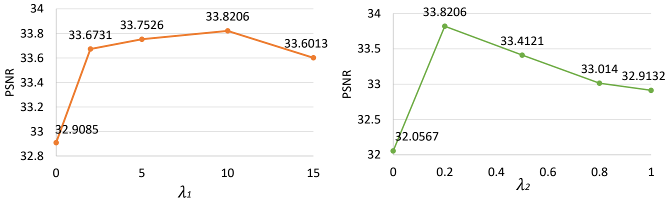

Fig. 10. Results of hyper parameters λ_1 and λ_2. The best result is gained when $\lambda_1 = 10.0$ and $\lambda_2 = 0.2$.

As for λ_2 in Eq. (6), which is the weight of l_1-reconstruction loss of the coarse prediction in TexNet. When λ_2 is 0.0, the TexNet is trained without l_1-reconstruction loss of the coarse prediction, the result is heavily harmed. It demonstrates that the coarse-to-fine architecture is effective in TexNet. The best performance of result is obtained when λ_2 is set as 0.2. And performance drops when $\lambda_2 > 0.2$, which reflects that the constraints on fine predictions are more important than that on coarse ones. So we set λ_2 as 0.2 in experiments.

V. CONCLUSION

In this paper, we propose a novel structure-guided video inpainting approach, which effectively utilizes structure information to generate fine-detailed frames. The edges in the missing region are first estimated to represent the target structure explicitly, using an edge inpainting network. Then under the guidance of the sparse edges, the missing content can be synthesized with intact structure and fine details. Our proposed structure attention module effectively exploits the correlation between structure and textures to improve visual quality. Besides, the temporal coherence of the inpainting frames is further enhanced by our flow-assisted losses. Experiments on YouTubeVOS and DAVIS datasets demonstrate the effectiveness of our method.

REFERENCES

[1] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang, "Free-form image inpainting with gated convolution," *arXiv preprint arXiv:1806.03589*, 2018.

[2] W. Xiong, J. Yu, Z. Lin, J. Yang, X. Lu, C. Barnes, and J. Luo, "Foreground-aware image inpainting," in *The IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[3] K. A. Patwardhan, G. Sapiro, and M. Bertalmío, "Video inpainting under constrained camera motion," *IEEE Transactions on Image Processing*, vol. 16, no. 2, pp. 545–553, 2007.

[4] 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.

[5] A. Newson, A. Almansa, M. Fradet, Y. Gousseau, and P. Pérez, "Video inpainting of complex scenes," *SIAM Journal on Imaging Sciences*, vol. 7, no. 4, pp. 1993–2019, 2014.

[6] 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.

[7] D. Kim, S. Woo, J.-Y. Lee, and I. S. Kweon, "Deep blind video decaptioning by temporal aggregation and recurrence," in *The IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[8] R. Xu, X. Li, B. Zhou, and C. C. Loy, "Deep flow-guided video inpainting," in *The IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[9] D. Kim, S. Woo, J.-Y. Lee, and I. S. Kweon, "Deep video inpainting," in *The IEEE Conference on Computer Vision and Pattern Recognition*, 2019.

[10] K. Nazeri, E. Ng, T. Joseph, F. Qureshi, and M. Ebrahimi, "EdgeConnect: Generative image inpainting with adversarial edge learning," *arXiv preprint arXiv:1901.00212*, 2019.

[11] 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.

[12] C. Ballester, M. Bertalmio, V. Caselles, G. Sapiro, and J. Verdera, "Filling-in by joint interpolation of vector fields and gray levels," *IEEE Transactions on Image Processing*, vol. 10, no. 8, pp. 1200–1211, 2001.

[13] H. Li, W. Luo, and J. Huang, "Localization of diffusion-based inpainting in digital images," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 12, pp. 3050–3064, 2017.

[14] K. Li, Y. Wei, Z. Yang, and W. Wei, "Image inpainting algorithm based on tv model and evolutionary algorithm," *Soft Computing*, vol. 20, no. 3, pp. 885–893, 2016.

[15] G. Sridevi and S. S. Kumar, "Image inpainting based on fractional-order nonlinear diffusion for image reconstruction," *Circuits, Systems, and Signal Processing*, pp. 1–16, 2019.

[16] 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.

[17] A. A. Efros and W. T. Freeman, "Image quilting for texture synthesis and transfer," in *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, 2001.

[18] C. Barnes, E. Shechtman, A. Finkelstein, and D. B. Goldman, "Patchmatch: A randomized correspondence algorithm for structural image editing," in *ACM Transactions on Graphics (ToG)*, vol. 28, no. 3, 2009, p. 24.

[19] K. Sangeetha, P. Sengottuvelan, and E. Balamurugan, "Combined structure and texture image inpainting algorithm for natural scene image completion," *Journal of Information Engineering and Applications*, vol. 1, no. 1, pp. 7–12, 2011.

[20] T. Ružić and A. Pižurica, "Context-aware patch-based image inpainting using markov random field modeling," *IEEE Transactions on Image Processing*, vol. 24, no. 1, pp. 444–456, 2014.

[21] D. Ding, S. Ram, and J. J. Rodríguez, "Image inpainting using nonlocal texture matching and nonlinear filtering," *IEEE Transactions on Image Processing*, vol. 28, no. 4, pp. 1705–1719, 2019.

[22] Q. Guo, S. Gao, X. Zhang, Y. Yin, and C. Zhang, "Patch-based image inpainting via two-stage low rank approximation," *IEEE Transactions on Visualization and Computer Graphics*, vol. 24, no. 6, pp. 2023–2036, 2018.

[23] H. Lu, Q. Liu, M. Zhang, Y. Wang, and X. Deng, "Gradient-based low rank method and its application in image inpainting," *Multimedia Tools and Applications*, vol. 77, no. 5, pp. 5969–5993, 2018.

[24] J.-B. Huang, S. B. Kang, N. Ahuja, and J. Kopf, "Temporally coherent completion of dynamic video," *ACM Transactions on Graphics (TOG)*, vol. 35, no. 6, p. 196, 2016.

[25] Y. Wexler, E. Shechtman, and M. Irani, "Space-time completion of video," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, no. 3, pp. 463–476, 2007.

[26] Y. Umeda and K. Arakawa, "Removal of film scratches using exemplar-based inpainting with directional median filter," in *2012 International Symposium on Communications and Information Technologies (ISCIT)*. IEEE, 2012, pp. 6–11.

[27] A. Ghanbari and M. Soryani, "Contour-based video inpainting," in *2011 7th Iranian Conference on Machine Vision and Image Processing*. IEEE, 2011, pp. 1–5.

[28] A. Xia, Y. Gui, L. Yao, L. Ma, and X. Lin, "Exemplar-based object removal in video using gmm," in *2011 International Conference on Multimedia and Signal Processing*, vol. 1. IEEE, 2011, pp. 366–370.

[29] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[30] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[31] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, 2015, pp. 91–99.

[32] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.

[33] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 3431–3440.

[34] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 4, pp. 834–848, 2017.

[35] J. Xie, L. Xu, and E. Chen, "Image denoising and inpainting with deep neural networks," in *Advances in neural information processing systems*, 2012.

[36] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros, "Context encoders: Feature learning by inpainting," in *The IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[37] C. Yang, X. Lu, Z. Lin, E. Shechtman, O. Wang, and H. Li, "High-resolution image inpainting using multi-scale neural patch synthesis," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 6721–6729.

[38] S. Iizuka, E. Simo-Serra, and H. Ishikawa, "Globally and locally consistent image completion," *ACM Transactions on Graphics (ToG)*, vol. 36, no. 4, p. 107, 2017.

[39] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang, "Generative image inpainting with contextual attention," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5505–5514.

[40] G. Liu, F. A. Reda, K. J. Shih, T.-C. Wang, A. Tao, and B. Catanzaro, "Image inpainting for irregular holes using partial convolutions," in *The European Conference on Computer Vision*, 2018.

[41] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *The IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[42] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox, "Flownet 2.0: Evolution of optical flow estimation with deep networks," in *The IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[43] N. Xu, L. Yang, Y. Fan, J. Yang, D. Yue, Y. Liang, B. Price, S. Cohen, and T. Huang, "YouTube-VOS: Sequence-to-sequence video object segmentation," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 585–601.

[44] J. Pont-Tuset, F. Perazzi, S. Caelles, P. Arbeláez, A. Sorkine-Hornung, and L. Van Gool, "The 2017 DAVIS challenge on video object segmentation," *arXiv:1704.00675*, 2017.

[45] Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli *et al.*, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.

[46] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "GANs trained by a two time-scale update rule converge to a local nash equilibrium," in *Advances in Neural Information Processing Systems*, 2017, pp. 6626–6637.

[47] S. Min, X. Chen, Z.-J. Zha, F. Wu, and Y. Zhang, "A two-stream mutual attention network for semi-supervised biomedical segmentation with noisy labels," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, 2019, pp. 4578–4585.