## Response to the Reviewers’ Comments

## Structure-Guided Deep Video Inpainting

Chaoqun Wang, Xuejin Chen, Shaobo Min, Jiaping Wang, Zheng-Jun Zha

Dear Editor:

Thank you very much for managing the review of our manuscript: "Structure-Guided Deep Video Inpainting". We would like to thank the reviewers for their thoughtful and helpful comments and suggestions. By addressing each of the points raised by the reviewers, we believe that the quality and clarity of the manuscript have been improved. The major updates in this revision are summarized as follows:

* Several new experiments have been added and revised according to the reviewers’ suggestions, *e.g.,* comparing SAM with other variants of attention module, trying different input frame gaps, evaluating different architectures of ENet, comparing with two related ICCV papers, and comparing the temporal warping error of our method and baselines.
* More discussions about the difference between our method and related works have been added, *e.g.,* a new discussion part about SAM and self-attention in Sec.III-B and more comparison in Table I. Some related references are also added according to the reviewers’ suggestions.
* More explanations and implementation details have been added and revised to improve understanding of our method, *e.g.,* the detailed network implementation.

These major revisions, as well as a few minor alters, have improved the quality of this work substantially. We present the point-by-point detailed responses to individual concerns of each reviewer below.

\*\*\*\*\*\*\*\***Reviewer 1\*\*\*\*\*\*\*\*\***

**Reviewer Comments 1:** *I think the paper is well presented and the solution is well studied. My only concern is that it lacks comparisons between SAM and other variants of attention module. It would be better to include a comparison of SAM with self-attention, and give more discussions on the advantages of SAM comparing to self-attention?*

**Response:** Thanks for your comments. Indeed, the major implementation difference between our SAM and self-attention lies in the query input. Our query is the edge feature maps, which is a different modality from the key input of video features, thus SAM uses two-layer FCs to explicitly introduce the edge information to video features via attention in Fig. 4. More importantly, our SAM is proved to alleviate the over-blurry problem, showing the importance of edge guidance in video inpainting. This point has been less explored in previous work. Detailed discussion between SAM and self-attention has been added in Sec III-B above Eq. (7) on page 5.

To demonstrate the superiority of SAM over other attention variants, we conducted more comparisons with self-attention and show the results in Table III in our revised version. In Table III, SimATT is a simple attention module which directly generates an attention map from edge feature via 2-layer FC and then applies the attention map to video feature. Thus, compared to SAM, SimATT lacks the edge-texture interaction during attention generation, which performs worse than SAM. SelfATT applies the general self-attention module to the video features only, without edge guidance. It leads to better results than adding simple attention. By adding edge guidance in our SAM, we obtain better results than both SimATT and SelfATT. We added more discussion about the comparison in Sec. IV-D-2) on page 9 in our revised manuscript.

\*\*\*\*\*\*\*\***Reviewer 2\*\*\*\*\*\*\*\*\***

**Reviewer Comments 1:** *There are multiple loss functions defined, and it makes it somewhat confusing to understand which is the final loss. For example, (2) is first presented as the objective, then, it is said that it changes to (9) to incorporate the flow information. I think it can be more streamlined and the presentation can be made more concise.*

**Response:** We are sorry for the confusing expression. We wanted to formulate the simple framework without motion first and then added the motion guidance to our inpainting framework. Therefore, Eq. (2) defines the GAN loss to train the edge inpainting network when we do not utilize motion information. Then the flow is introduced to enhance the temporal consistency of edge generation in Sec. III-C. Thus, the final objective function of the edge inpainting network becomes Eq. (11) in our revised manuscript (Eq. (9) in the original manuscript). We revised the organization in Sec. III to make the idea clearer.

**Reviewer Comments 2:** *The notations are somewhat confusing. For example, what is the difference between the \tilde{\mathbf{E}} in (1) and \tilde{E} in (3)? Also, what is the difference between the two loss terms in (6)? They are not clearly introduced and differentiated, which causes confusion to the readers.*

**Response:**  in Eq. (1) denotes the concatenation of five edge maps. We generate five edge maps in one forward pass. in Eq. (3) denotes one of the five edge maps. The loss in Eq. (3) punishes every frame of the generated edge maps. The texture inpainting network adopts a coarse-to-fine architecture. The former term in Eq. (8) (Eq. (6) in original manuscript) is to regulate the final refined output, while the latter one is for the rough generation of the coarse network. We revised the notation expressions in Sec. III and highlight them in Fig. 2 to make our algorithm clearer.

**Reviewer Comments 3:** *Any justifications for selecting \mathbf{V}=\{V\_{t-7},V\_{t-3},V\_t,V\_{t+3}, V\_{t+7}\}? Why just gaps between the frames? Can we use smaller or larger gaps?*

**Response:** We select neighboring frames with gaps () similarly following previous works (e.g., D. Kim et al, “Deep Video Inpainting” in CVPR2019 with frames ). There is a trade-off between temporal smoothness and spatial details when selecting frame gaps. Usually, a large frame gap enables the model to capture long-distance texture contexts but introduces serious temporal jitters. Conversely, a small frame gap brings smooth inpainting but less temporal complementary information. To justify our selection of frame gap, we added experiments and discussions under different frame gaps in Table VII and Sec. IV-D-7) on page 11 in our revised manuscript. The results show that the small gap () makes inpainting results more temporally smooth with a small temporal warping error, but its frame generation quality (PSNR, SSIM, and FID) is lower due to less auxiliary temporal information from distant frames. Differently, a larger gap () harms both spatial and temporal coherence, because it may introduce noise from long-distance frames. Finally, we choose the gap of , which achieves a balanced performance.

**Reviewer Comments 4:** *In page 3, right column, line 4 from the bottom: it says “... a good trade-off between spatial and temporal coherence”. Any justifications? What kind of trade-off? How does the trade-off look like? Which one, spatial or temporal, is more significant? Such statements sound in vain and it needs to be justified.*

**Response:** The design of ENet in Fig. 2 is a hybrid 3D+2D architecture. The 3D conv. part (3 layers) in the encoder and decoder can capture *temporal coherence* that takeslarge computation consumption. The intermediate 2D conv. part (8 layers) is efficient with large spatial receptive field to capture spatial coherence. Under a limited number of network parameters, we can adopt more 2D conv. layers, thereby obtaining larger spatial receptive field compared to 3D convolutions. Thus, ENet can achieve a balance between spatial (large receptive field of 2D part) andtemporal coherence (3D part).

To justify such a design, we compare our hybrid 3D+2D architecture with two variants: 1) Variant-1: directly replacing the 2D convs. of ENet with 3D convs., which has the same number of layers with our ENet; and 2) Variant-2: replacing the 2D part with fewer 3D convs. layers while keeping a similar number of network parameters with our current architecture. The results are added in Table VI on page 10 in our revised manuscript. We can see that the performance of Variant-1 is slightly better than ours, because 3D convs. capture finer temporal details and introduce spatial information from features of neighbor frames. However, it takes about 2.5 times of parameter numbers over our hybrid 3D+2D architecture. Variant-2 results in lower generation quality because of its shallower intermediate layers. In comparison, our hybrid 3D+2D architecture achieves comparable results with Variant-1 but much fewer parameters, demonstrating its better balance between computational cost and inpainting quality. We added a detailed discussion on the concept of our hybrid 3D+2D architecture in Sec III-A (page 4) and discussed the effects of different architectures of ENet in the ablation study in Sec IV-D-4 (page 10) in our revised manuscript.

**Reviewer Comments 5:** *There are some typos here and there. I would recommend having English editing for the next version.*

**Response:** Thanks for your comments. We carefully proofread our paper and corrected typos in the revised version.

\*\*\*\*\*\*\*\***Reviewer 3\*\*\*\*\*\*\*\*\***

**Reviewer Comments 1:** *The proposed model combines multiple previous ideas, including edge-guided inpainting [11], hybrid 3D-2D convolution network [10], flow-guided temporal consistency loss [8,9,10], asymmetric non-local.*

**Response:** We were inspired by these previous methods when designing our algorithm. However, these ideas are not simply combined in video inpainting task. For example, the edge-guidance [13] ([11] in original manuscript) only considers spatial enhancement in image inpainting. For video inpainting, we have to simultaneously consider both spatial and temporal coherence. Thus, we introduce a novel module SAM to explore the correlation between structure information and video content. Besides, the previous methods did not explore how to adapt the hybrid 3D-2D convolution network [10], flow-guided temporal consistency loss [8,9,10], asymmetric non-local to fuse the edge, motion and texture modality data for video inpainting.

**Reviewer Comments 2:** *Missing comparisons with the state-of-the-art approaches. Please discuss the details and conduct experiment-level comparisons with the following works:  
1) Onion-Peel Networks for Deep Video Completion [Oh et.al., ICCV2019]   
2) Copy-and-Paste Network (Lee et.al., ICCV2019)*

**Response:** Thanks for your comments. We add a detailed discussion and comparison of the differences between our method and these two recent papers. First, in related works of Sec. II-d) on page 3, we introduce the concept difference. OPN [12] uses the non-local formulation to calculate the similarity between pixels in reference and target frames, and then copies the contents based on the similarity. Similarly, Copy-and-Paste [11] copies matching contexts in aligned reference frames and pastes them to the corrupted region. However, both OPN and Copy-and-Paste only consider the texture correlation between neighboring frames, while ignoring high-level structures. In comparison, we explore the correlation between structures (edges) and textures to achieve high-quality inpainting results.

Second, in Table I on page 6 in our revised manuscript, we add the experimental comparison with these two methods. Due to their training codes are unavailable, we directly use their trained models in our testing dataset. From the Table, we can see that the results of both OPN and Copy-and-Paste are worse than ours. OPN and Copy-and-Paste only consider texture interaction between video frames, while ignoring the important edge-texture correlation. Thus, the generation quality of our method is superior to that of both two methods with higher inference speed. It shows that the structure information introduced by edge guidance is important in video inpainting, and our method is capable of exploiting the correlation between video contents and object structures. The detailed analysis is added in Sec. IV-B on page 8.

Finally, the visualized comparisons between OPN, Copy-and-Paste, and our method on both YouTubeVOS and DAVIS are added in Fig. 5 and 7 on pages 7-8. It can be seen that our method obtains better-visualized results than these two methods, and faster inference speed in Table I. The dynamic video inpainting results are also added in the supplementary video material.

**Reviewer Comments 3:** *The proposed framework has a limited temporal window size of 14 (~0.5 seconds). What happens if the necessary information is not in this temporal window size (e.g., slowly moving large objects)? Please provide examples of larger and larger holes and see how much better the proposed model gets compared to the baselines.*

**Response:** Actually, when a large object moves slowly, the inpainting performance will inevitably drop due to unavailable temporal complementary information. However, a larger temporal window size will bring severer temporal jittering problem and worse video quality, as referred to response to Comment 3 of reviewer 2.

According to your advice, we conducted experiments on different sizes of fixed square holes. We use three types of holes, including large size hole of, Medium size hole of , and small size hole of , while the testing frames are in the size of . Experimental results are shown in Table VIII on page 11. The performance of all the methods drops significantly when testing on larger holes. Our method achieves the best performance under most circumstances, except for the PSNR under the large hole size. The PSNR under large holes of our method is slightly worse than that of OPN, which designs an asymmetric attention block to support an unlimited spatial-temporal window. However, our method performs better than OPN under other settings. The results show that the performance of our method is stable on different hole sizes. The results and discussions are added in Sec.IV-D-8) on page 11 in our revised manuscript.

**Reviewer Comments 4:** *Please indicate the memory consumption in the table. This is important since non-local matching, in general, brings heavy computational burdens, limiting the applicability of the framework in high-resolution inpainting.*

**Response:** Thanks for your comments.We include #Params and GFLOPs in Table IV on page 9 to show the computation cost of our method. As shown in the table, the proposed SAM does not bring too much cost to the whole network, according to #Params and GFLOPs. The reason is that the inputs to SAM are down-sampled edge maps and video feature maps, of which the resolution is affordable. The comparable inference speed also proves that, in current setting, SAM can bring performance gains with affordable cost and can be used in practice. However, when coming to high-resolution inpainting, SAM may put a high demand on computational resources. We also revised this part in Sec. IV-D-1) on page 8 in our revised manuscript.

**Reviewer Comments 5:** *Could you please apply the framework on higher-resolution input (1024x512)?*

**Response:** We use models trained on small sizes to test on size. Results are shown in Table IX on page 11. Notably, OPN is not compared because of its significantly high memory cost that cannot be deployed on our GPU cluster. Compared to other methods, our method achieves the best performance with great superiority for high-resolution inpainting, demonstrating the effectiveness and robustness of our method. We have added the analysis on higher resolution in the Sec. IV-D-9) on page 11 in the revised version.

**Reviewer Comments 6:** *Please include Video FID or Temporal warping error in the Table1,2 to quantify that the outputs are indeed temporally consistent.*

**Response:** Thanks for your suggestion. We added the evaluation metric of temporal warping error (Tem-ERR) in Table I (Comparison with state-of-the-art methods) on page 6 and Table IV (Ablation study; Table 2 in original manuscript) on page 9 in our revised manuscript. According to the results, our method is indeed temporal consistent. Detailed analysis of the comparison of temporal warping loss with other methods is added in Section IV-B on page 7. In terms of temporal warping error about temporal smoothness, our method outperforms most of the existing methods, which proves that the temporal consistency constraint can exactly facilitate texture smoothness. Specifically, in Table I, our method is comparable with DFVI and worse than DVI. The performance superiority of DVI mainly comes from the recurrent feedback loop and a memory layer (LSTM) for temporal consistency of inpainted frames. DFVI iteratively propagates pixel colors according to optical flow. Both DFVI and DVI focuses on designing temporal modules to obtain smooth inpainting, thereby obtaining more smoothing results but much slower inference speed than our method. More importantly, under edge guidance, our method generates results with fine structural details, which obtains obviously higher generation quality (PSNR, SSIM, FID) than DFVI and DVI.

While adding temporal warping error in Table IV, we also add discussion in Sec. IV-D-5) on page 10. According to the metric of temporal warping error, our results are temporally smoother when the generation quality improves as gradually adding edge and SAM. Adding the guidance of optical flow brings further enhancement to temporal consistency. This demonstrates the effect of flow guidance.

**Reviewer Comments 7:** *why not using reconstruction loss (e.g., L1) for the edge inpainting ?*

**Response:** In Eq. (2), L1 reconstruction loss is not used due to the great sparsity of edge maps. For very sparse edge maps, L1 loss is very sensitive to small drifts and noises. In order to recover scene structures in the inpainted edge maps, we use feature matching loss instead of L1 loss during edge generation. When we introduce explicit flow guidance to the training of ENet, we add L1 loss  in Eq. (11) to ensure pixel-level consistency between edges from neighboring frames. We have added the explanations in the Sec.III-A and C in the revised version.

**Reviewer Comments 8:** *please include inference time details (is the framework applied to video frames in an auto-regressive manner?)*

**Response:**  We add the inference details in Implementation details part in Section IV-A on page 6. During inference, we sequentially feed five neighboring frames with gaps into the inpainting network and output the reconstructed frame in one forward pass. The computational cost and inference speed in Table IV on page 9 are reported under this setting. We did not implement our framework in an auto-regressive manner. We are going to optimize our implementation to support streaming process.

**Reviewer Comments 9:** *please remove the bold font in equation (6).*

**Response:** We usebold fonts in equation (8) (Eq. (6) in the original manuscript) to denote the set for the five frames of coarse generation. We revised the expressions of Sec. III to make symbol explanations more clear in our paper.

**Reviewer Comments 10:** *please formulate the forward process of TexNet as equation (1).*

**Response:** Thank you for the suggestion. We modified the formulation of TexNet and added more details in Sec. III-B on page 4 in the revised manuscript.

**Reviewer Comments 11:** *please include the details of the input shape. For example, is it concat or masked input in equation (1)?*

**Response:** The input of edge inpainting network is the concatenation of masked grayscale frames, masked edge maps, and the masks. The input of the coarse texture inpainting network is the concatenation of masked images, the synthesized edge maps, and the masks. The input of the refinement texture inpainting network is the concatenation of the rough inpainting results, the synthesized edge maps, and the masks. We have revised these details in Sec. III in our manuscript.

**Reviewer Comments 12:** *please include the details of the network architecture (both the generator and discriminator) as a table.*

**Response:** We provide thedetails of our network implementation in Table II and III on page 6. We are also going to share our codes and data in the future.