ProPainter: Improving Propagation and Transformer for Video Inpainting
— Supplementary Materials —

Shangchen Zhou Chongyi Li Kelvin C.K. Chan Chen Change Loy
S-Lab, Nanyang Technological University
{s200094, chongyi.li, chan0899, ccloy}@ntu.edu.sg
https://shangchenzhou.com/projects/ProPainter

In this supplementary materials, we provide additional details, further discussions, and more results to supplement the main paper. In Sec. A, we present the architecture details and loss functions of our proposed ProPainter. In Sec. B, we provide in-depth analysis of the performance improvement achieved by our method and highlight its advantages. Sec. C contains more quantitative evaluations and visual comparisons.

A. Architecture and Loss Details

A.1. Architecture

Our network adopts two distinct deformable alignment modules in the recurrent flow completion network (RFC) and feature propagation, respectively. To provide further clarity, we present a detailed illustration of the former alignment module (w/o flow guided) in Figure 1, which can be easily compared with the latter alignment module (w/ flow guided) depicted in Figure 3 of the main paper. There are two main differences between the two modules: 1) different condition pools were employed to predict the parameters of the deformable convolutional networks (DCN); 2) the former predicts the DCN offsets directly, while the latter uses optical flow as the base offset of DCN and predicts the residual offsets to the flow fields.

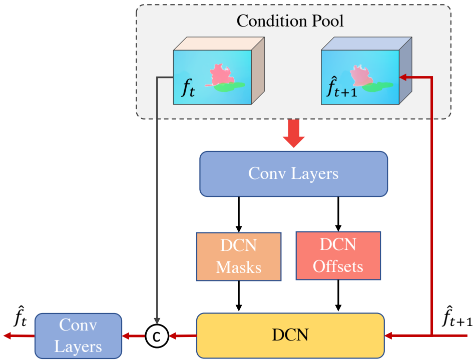


Figure 1: An illustration of deformable alignment module that is adopted in the recurrent flow completion network.

A.2. Loss Functions

Loss Functions of RFC network. To train the recurrent flow completion network (RFC), we utilize two losses. The first one is the reconstruction loss that is applied to both valid and invalid regions, as depicted in the following equation:

$$\mathcal{L}_{rec}^{flow} = \frac{\left\| M_t \odot (\hat{F}_t - F_t) \right\|_1}{\|M_t\|_1} + \frac{\left\| (1 - M_t) \odot (\hat{F}_t - F_t) \right\|_1}{\|1 - M_t\|_1}, \tag{1}$$

where $\odot$ denotes the dot product. The second one is second-order smooth loss [5] that encourages the smooth and coherent completed flow fields, which is a critical property for the subsequent propagation modules. The loss can be expressed as:

$$\mathcal{L}_{smooth}^{flow} = \left\| \triangle \hat{F}_t \right\|_1, \tag{2}$$

where $\triangle$ denotes the divergence operator. The overall loss function of RFC is: $\mathcal{L}^{flow} = \alpha_1 \mathcal{L}^{flow}_{rec} + \alpha_2 \mathcal{L}^{flow}_{smooth}$, where we set $\alpha_1 = 1, \alpha_2 = 0.5$ in our experiments.

Loss Functions of ProPainter. Our ProPainter is trained using two types of loss. For reconstruction loss, we use L1 loss to measure the distance between output video sequence $\hat{Y}$ and ground-truth one Y:

$$\mathcal{L}_{rec} = \left\| \hat{Y}_t - Y_t \right\|_1 . \tag{3}$$

Furthermore, we introduce an adversarial training procedure with a T-PatchGAN based discriminator D [1] to enhance the quality and coherence of generated videos by differentiating between real and reconstructed videos:

$$\mathcal{L}_D = \mathbb{E}_{\mathrm{Y}} \Big[\log D(\mathrm{Y}) \Big] + \mathbb{E}_{\hat{\mathrm{Y}}} \Big[1 - \log D(\hat{\mathrm{Y}}) \Big]. \tag{4}$$

For the generator, the GAN loss is formulated as:

$$\mathcal{L}_G = -\mathbb{E}_{\hat{\mathrm{Y}}} \Big[\log D(\hat{\mathrm{Y}}) \Big]. \tag{5}$$

Thus, the objective of ProPainter learning is: $\mathcal{L}^{inpaint} = \lambda_1 \mathcal{L}_{rec} + \lambda_2 \mathcal{L}_G$, where we set $\lambda_1 = 1, \lambda_2 = 0.01$.

B. More Discussions

As indicated in Table 1 of the main paper, our proposed ProPainter outperforms the state-of-the-art networks by a large margin on all quantitative metrics, especially on the DAVIS [6] dataset. In this section, we explore the primary factor that contributes to these remarkable performance gains and discuss the situations in which our approach has a competitive edge.

B.1. Factor Behind Improved Performance

Our proposed ProPainter benefits greatly from global image propagation, which significantly reduces the difficulty of learning for subsequent modules. As shown in Figure 2, image propagation has filled the majority of the masks and even entirely completed masked regions. This means that modules following image propagation only need to refine and complement the completed contents of image propagation instead of learning the entire inpainting process. Our method differs from previous approaches [2, 9, 10] in several aspects: 1) In contrast to earlier image propagation methods that are independent of network training, which prevent the network from correcting propagation errors, our proposed image propagation is involved in the model training, enabling subsequent models to fix any texture misalignment or artifacts caused by image propagation; 2) We employ a more reliable propagation strategy, which is compared in Figure 6 of the main paper; 3) Unlike previous methods that are implemented on the CPU and involved some complex and time-consuming processes, such as indexing pixel-wise flow trajectories and Poisson blending, we implement a more efficient image propagation with GPU acceleration.

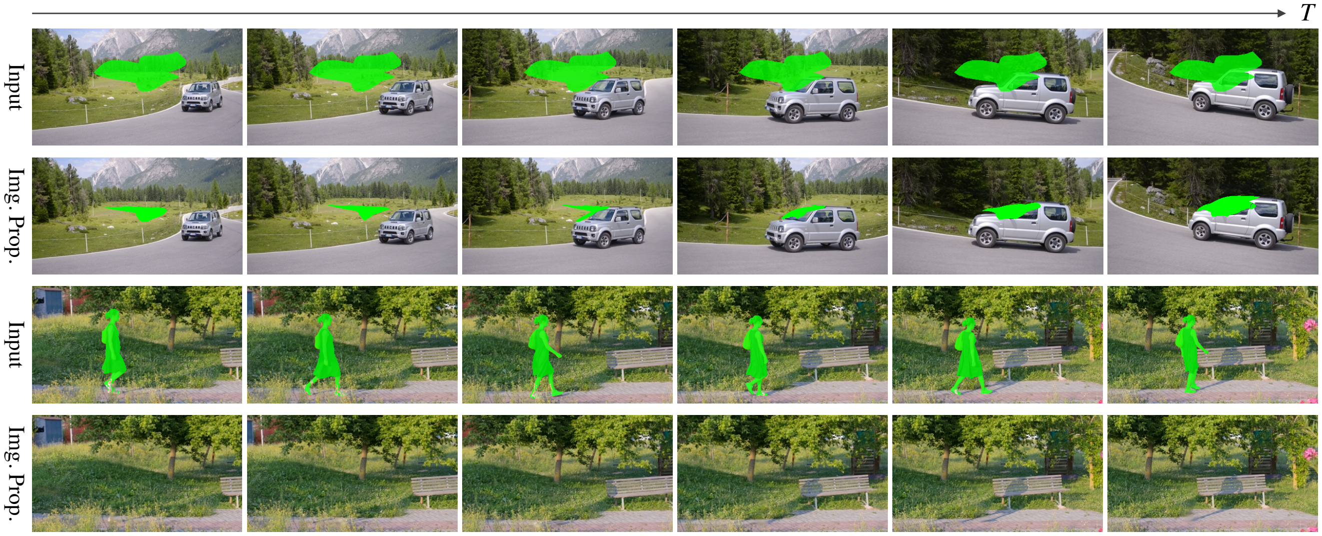


Figure 2: The initial results and updated masks after our global image propagation. Image propagation shows effective to fill most or entire masked regions, which significantly alleviates the learning difficulty experienced by video inpainting networks.

B.2. Motion Distribution

In the main paper, Table 1 shows that ProPainter's performance improvement is more noticeable on the DAVIS [6] dataset than on the YouTube-VOS [7] dataset. Our ablation study and analysis in the main paper attribute the performance gains primarily to the design of dual-domain propagation, which relies on motion flow fields to propagate information across videos. However, we have observed that many videos in the YouTube-VOS dataset have almost stationary scenes without motion, which limits the effectiveness of our dual-domain propagation module. Moreover, we have analyzed the motion magnitude distribution on both datasets and found that the YouTube-VOS dataset contains a greater proportion of regions with small motion, as presented in Figure 3.

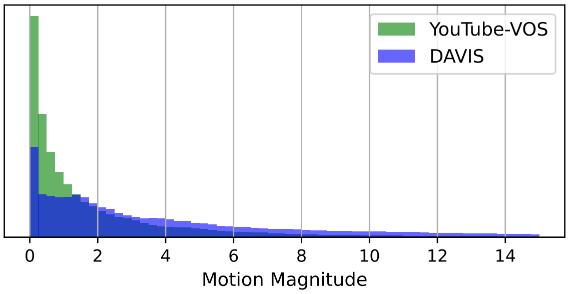


Figure 3: Motion magnitude distribution on YouTube-VOS [7] and DAVIS [6] datasets.

C. More Results

C.1. Quantitative Evaluation on 480p Videos

Table 1 presents a quantitative comparison on the DAVIS [6] dataset with 480p (864 × 480) videos. The comparison only includes STTN [8] and E^{2}FGVI [3], since other methods require memory demands exceeding 32G (such as TSAM [11], FuseFormer [4], and FGT [9]) or excessively long time for inference on a 480p video. Runtimes are measured on an NVIDIA Tesla V100 (32G) GPU. This comparison suggests that our method exhibits benefits in terms of both accuracy and efficiency even at a high resolution.

Table 1: Quantitative comparisons on DAVIS [6] dataset with 480p (864 × 480) videos.

	PSNR ↑	SSIM ↑	VFID ↓	Runtime (s/frame) ↓
STTN [8]	30.72	0.9534	0.055	0.262
E^{2}FGVI [3]	32.98	0.9693	0.041	0.332
ProPainter (Ours)	**33.81**	**0.9739**	**0.035**	**0.249**

C.2. Qualitative Comparisons on Flow Completion

In Figure 4, we provide a visual comparison of flow completion performance between our recurrent flow completion network and previous methods, including FGVC [2], FGT [9], and ISVI [10]. The results show that our recurrent flow completion network outperforms other methods in producing complete and accurate flow fields. As a result, the subsequent dual-domain propagation module can rely more on accurate optical flows, leading to a more reliable and precise propagation in later stages.

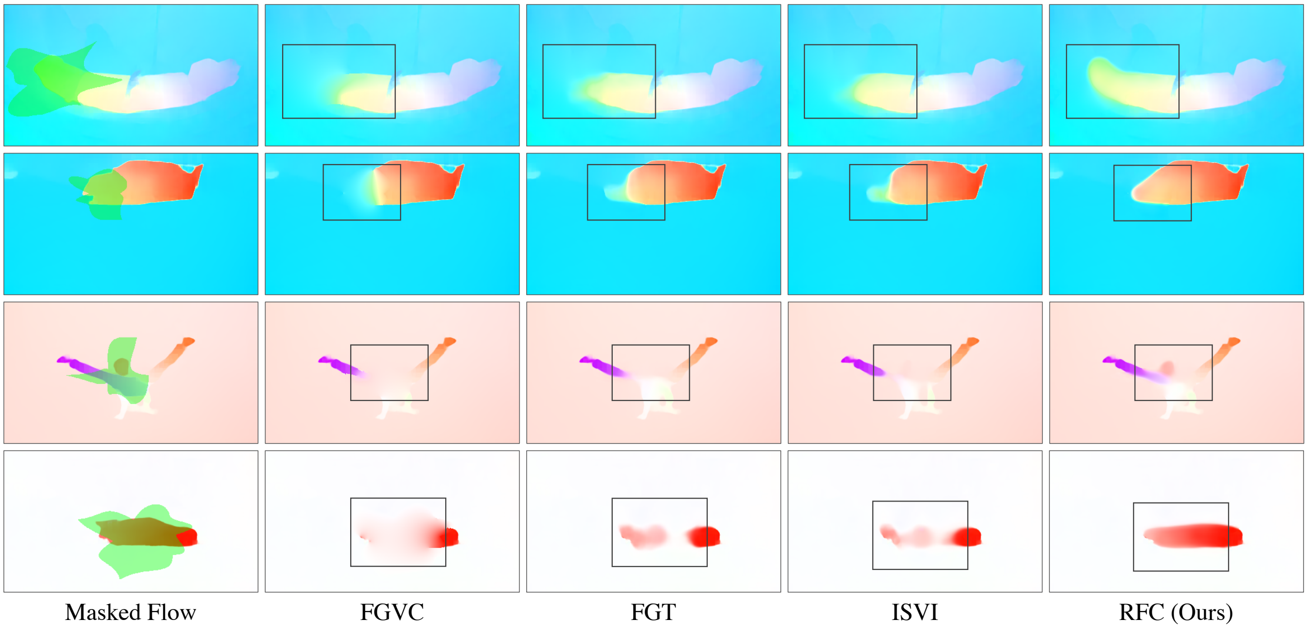


Figure 4: Qualitative comparisons of flow completion. Our recurrent flow completion network exhibits superiority in generating complete and faithful flow fields, thereby facilitating more precise and reliable propagation for ProPainter.

C.3. Qualitative Comparisons

In this section, we provide additional visual comparisons of our method with the state-of-the-art methods, including FuseFormer [4], FGT [9], and E^{2}FGVI [3]. Figures 5 and 6 present the comparisons of video completion performance on the YouTube-VOS [7] and DAVIS [6] datasets, respectively.

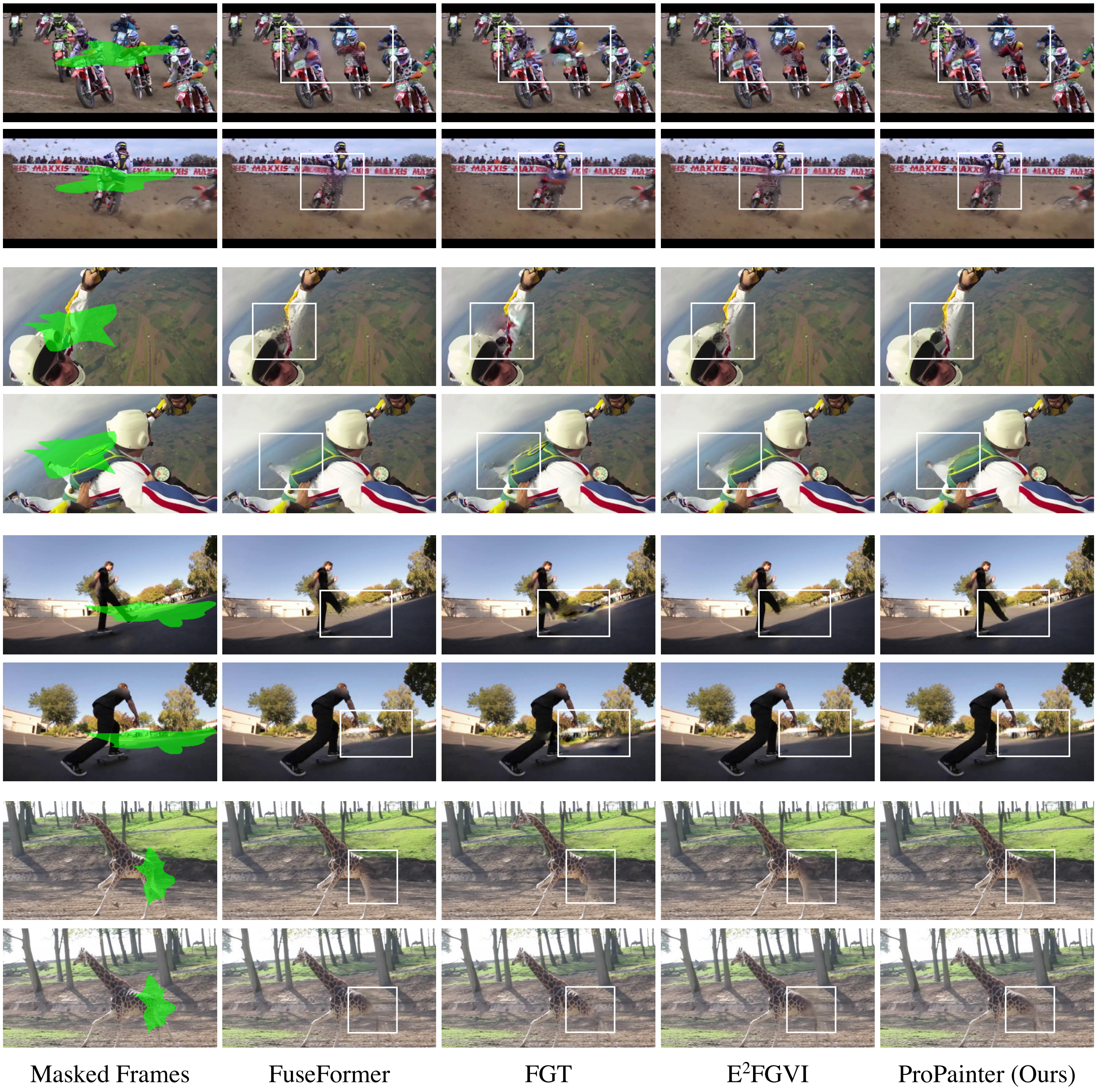


Figure 5: Qualitative comparisons on YouTube-VOS [7] dataset. Our ProPainter exhibits superiority in producing complete and faithful textures, resulting in enhanced spatiotemporal coherence for video inpainting. (**Zoom in for best view.**)

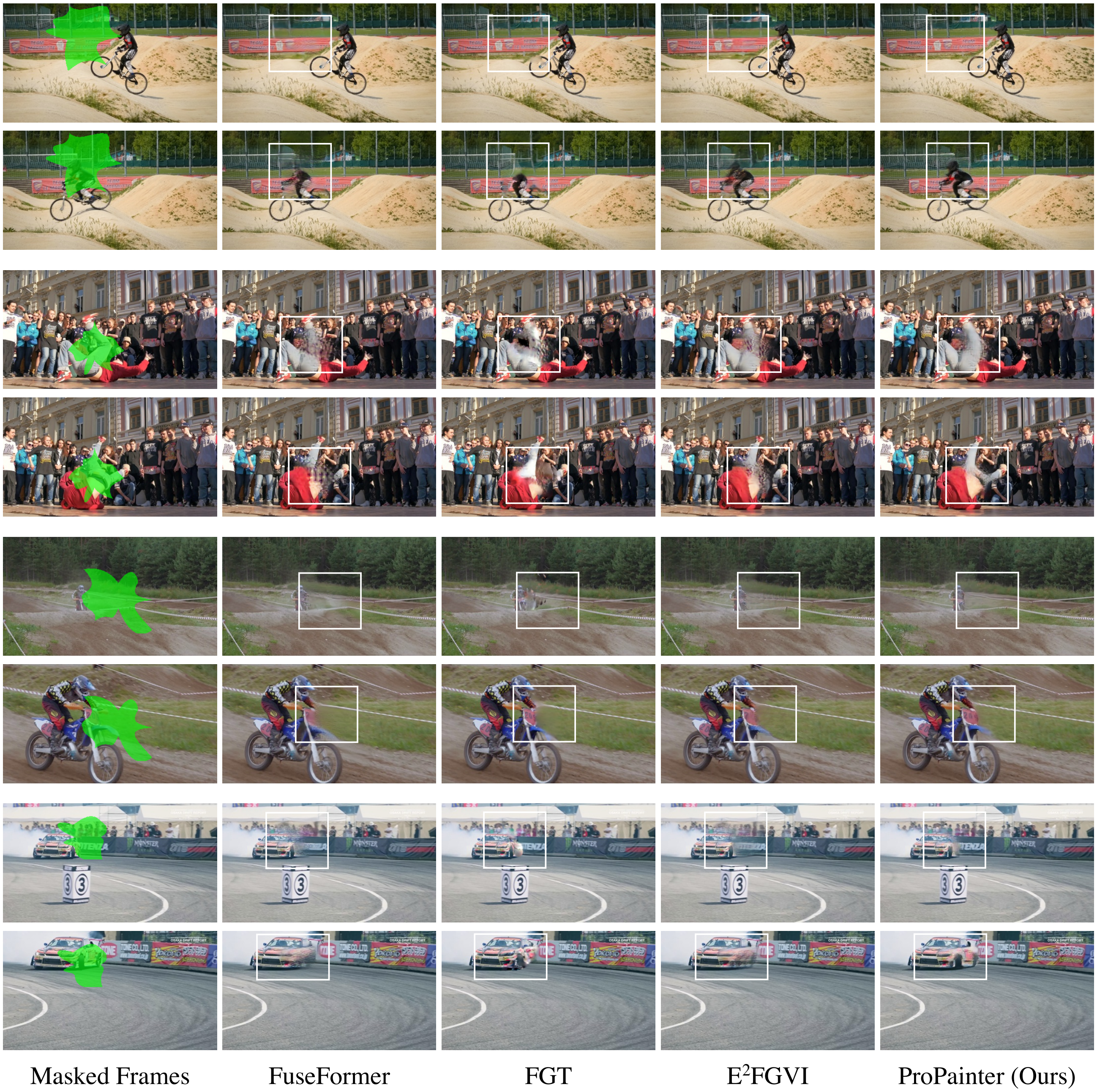
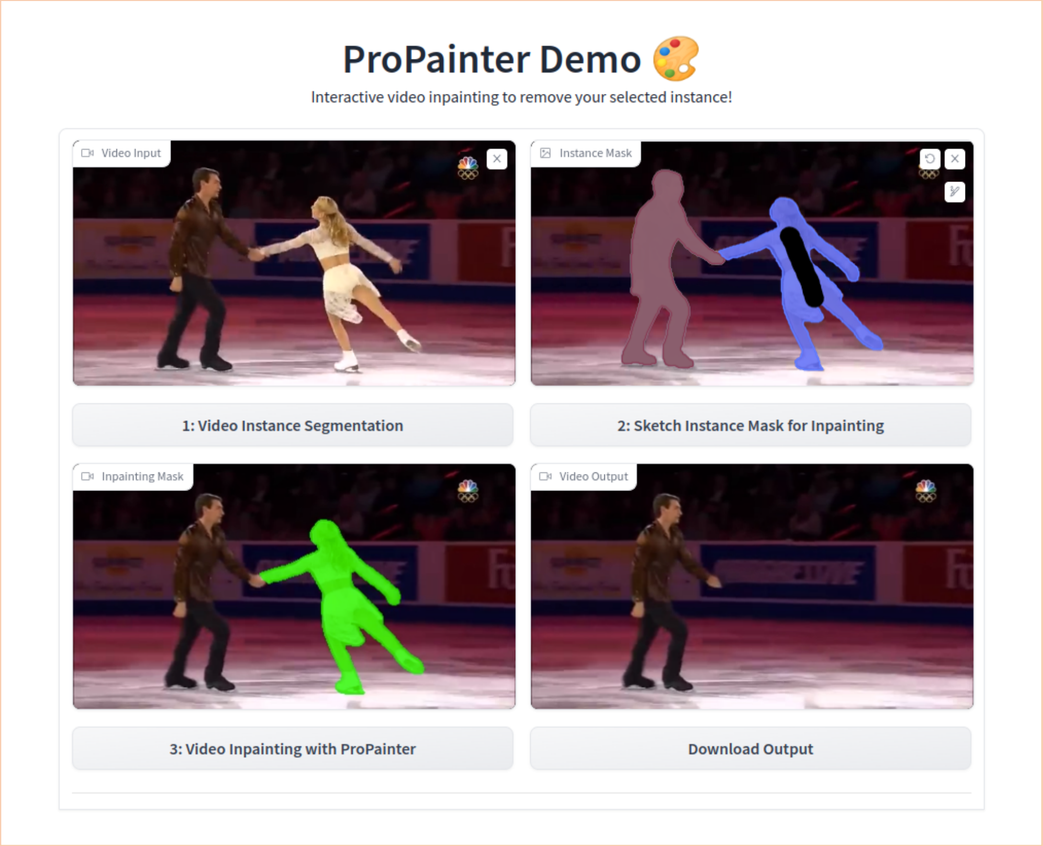


Figure 6: Qualitative comparisons on DAVIS [6] dataset. Our ProPainter exhibits superiority in producing complete and faithful textures, resulting in enhanced spatiotemporal coherence for video inpainting. (**Zoom in for best view.**)

Furthermore, our [project page] provides a video demo that showcases some results of object removal, along with an interactive demo using ProPainter. This demo incorporates a video instance segmentation network and enables users to select and remove specific objects from the video. A screenshot of this demo is presented in Figure 7.



ProPainter Demo 🎨

Interactive video inpainting to remove your selected instance!

Video Input
Instance Mask
1: Video Instance Segmentation
2: Sketch Instance Mask for Inpainting
Inpainting Mask
Video Output
3: Video Inpainting with ProPainter
Download Output

Figure 7: A screenshot of the interactive ProPainter demo.

References

[1] Ya-Liang Chang, Zhe Yu Liu, Kuan-Ying Lee, and Winston Hsu. Free-form video inpainting with 3d gated convolution and temporal patchgan. In *ICCV*, 2019. 2

[2] Chen Gao, Ayush Saraf, Jia-Bin Huang, and Johannes Kopf. Flow-edge guided video completion. In *ECCV*, 2020. 2, 4

[3] Zhen Li, Cheng-Ze Lu, Jianhua Qin, Chun-Le Guo, and Ming-Ming Cheng. Towards an end-to-end framework for flow-guided video inpainting. In *CVPR*, 2022. 3, 5

[4] Rui Liu, Hanming Deng, Yangyi Huang, Xiaoyu Shi, Lewei Lu, Wenxiu Sun, Xiaogang Wang, Jifeng Dai, and Hongsheng Li. Fuseformer: Fusing fine-grained information in transformers for video inpainting. In *ICCV*, 2021. 3, 5

[5] Simon Meister, Junhwa Hur, and Stefan Roth. Unflow: Unsupervised learning of optical flow with a bidirectional census loss. In *AAAI*, 2018. 1

[6] Federico Perazzi, Jordi Pont-Tuset, Brian McWilliams, Luc Van Gool, Markus Gross, and Alexander Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *CVPR*, 2016. 2, 3, 5, 6

[7] Ning Xu, Linjie Yang, Yuchen Fan, Jianchao Yang, Dingcheng Yue, Yuchen Liang, Brian Price, Scott Cohen, and Thomas Huang. YouTube-VOS: Sequence-to-sequence video object segmentation. In *ECCV*, 2018. 3, 5

[8] Yanhong Zeng, Jianlong Fu, and Hongyang Chao. Learning joint spatial-temporal transformations for video inpainting. In *ECCV*, 2020. 3

[9] Kaidong Zhang, Jingjing Fu, and Dong Liu. Flow-guided transformer for video inpainting. In *ECCV*, 2022. 2, 3, 4, 5

[10] Kaidong Zhang, Jingjing Fu, and Dong Liu. Inertia-guided flow completion and style fusion for video inpainting. In *CVPR*, 2022. 2, 4

[11] Xueyan Zou, Linjie Yang, Ding Liu, and Yong Jae Lee. Progressive temporal feature alignment network for video inpainting. In *CVPR*, 2021. 3