
Dynamic Shadow Unveils Invisible Semantics for Video Outpainting

— Supplementary Material —

Ruilin Li¹ Hang Yu^{1*} Jiayan Qiu²

¹School of Computer Engineering and Science, Shanghai University, China

²School of Computing and Mathematical Sciences, University of Leicester, United Kingdom
{ruilinli, yuhang}@shu.edu.cn, jq46@leicester.ac.uk

A Detailed Implementation

Network Architecture The latent diffusion model employs an autoencoder architecture consisting of an encoder and a decoder. The encoder maps the original video frames into a compact latent space, while the decoder reconstructs the video frames from these latent representations. To ensure coherent reconstructions across frames, we introduce additional temporal layers in the decoder, following the approach of [2]. Importantly, the encoder remains unchanged from its original image-based training, enabling direct reuse of the image diffusion model to process video frames in the latent space.

We build upon Stable Diffusion v1-5 as our core denoising backbone. The U-Net is conditioned on text embeddings derived from CLIP via cross-attention mechanisms. To extend its applicability to the spatiotemporal domain of video, we adapt the architecture by inflating its 2D convolutions and 2D group normalization layers into pseudo-3D convolutions and 3D group normalizations, respectively. Specifically, each spatial 2D convolutional layer is followed by a temporal 1D convolution to model temporal dynamics. Our 3D U-Net consists of four downsampling and four upsampling stages, with each stage outputting feature channels of [320, 640, 1280, 1280], following the design in [5].

Shadow-aware Instance Prediction Module The image encoder is a convolutional neural network responsible for extracting high-level semantic features from each padded input frame \hat{I}^t . We adopt a modified ResNet-50 backbone with the first convolutional layer using a 5×5 kernel, stride 2, and 64 output channels, followed by a 3×3 max pooling with stride 2. The encoder includes four residual stages with output channel sizes of 256, 512, 1024, and 2048 respectively. To reduce the feature dimensionality and align with downstream modules, an additional 1×1 convolution is appended after the final stage to produce a uniform 256-dimensional feature map.

The mask encoder is a lightweight convolutional network designed to extract features from binary instance masks, such as the visible object mask $M_{o_vis}^t$ and shadow mask $M_{s_vis}^t$. It consists of three convolutional layers. The first layer has a 3×3 kernel, stride 1, and 64 output channels; the second layer uses a 3×3 kernel with stride 2 and 128 output channels; and the third layer uses a 3×3 kernel with stride 2 and 256 output channels. Each convolution is followed by a ReLU activation and batch normalization. The final output feature is a 256-dimensional mask embedding.

The feature encoder is used to generate the spatiotemporal token representation z^t at each timestep by fusing the corresponding image, object, and shadow features. This module is composed of a three-layer convolutional network. The first layer uses a 3×3 kernel with stride 1 and 512 output channels; the second layer also uses a 3×3 kernel with stride 2 and 512 output channels; and the third layer uses a 1×1 kernel with stride 1 and 256 output channels to produce the final embedding. The input to this encoder is a channel-wise concatenation of the three feature maps F_i^t , F_o^t , and F_s^t .

*Corresponding author

The inter-frame fusion module is a Transformer encoder that models temporal dependencies across frames for each shadow-object pair. The input sequence consists of $T + 1$ tokens: one learnable global token z^0 and T frame-wise tokens z^1 to z^T . This module contains 4 transformer blocks, each composed of multi-head self-attention with 8 heads, an embedding dimension of 256, and a feedforward network with a hidden dimension of 1024. Residual connections and layer normalization are applied after each sub-layer. Learnable positional encodings [4] are added to the tokens to inject temporal information.

The inter-pair fusion module is another Transformer encoder, designed to fuse information spatially across N shadow-object pairs within the same frame. For each timestep t , the inter-frame fused features $h^{t,n}$ serve as input tokens. This module uses 2 transformer layers, each with 4 attention heads, a hidden size of 256, and a feedforward layer with 1024 hidden latent. Relative positional encoding [8] is used to better capture the spatial relationships between different object pairs.

The mask decoder is a convolutional network responsible for predicting the object and shadow masks from the fused features $\hat{h}^{t,n}$. It consists of three convolutional layers: the first uses a 3×3 kernel with stride 1 and 128 output channels; the second is a 3×3 convolution with stride 1 and 64 output channels; and the final prediction layer uses a 1×1 convolution to produce two channels, corresponding to the object and shadow mask logits respectively. The output logits are upsampled using bilinear interpolation to match the spatial resolution of the input masks.

The learnable global Token z^0 is a trainable embedding vector initialized randomly and jointly optimized during training. It shares the same embedding dimension (256) as other tokens and is used to aggregate temporal information across frames for each shadow-object pair. The output global token h^0 serves as a compact representation for each pair and is later used to guide video outpainting and temporal flow completion modules.

Inference Details The input video resolution is set to $256 \times 256 \times 3$. During testing, we perform inference with a batch size of 4 using a 48GB RTX A6000 GPU. For the instance adapter and flow adapter, we set the weighting parameters to $\gamma_1 = 1$ and $\gamma_2 = 0.7$, as empirical results indicate that these values yield high-quality outpainting performance. In terms of inference time at 256×256 resolution, M3DDM requires 48 seconds, MOTIA takes approximately 8 minutes, while our method completes the task in 55 seconds.

B Benchmark Details

B.1 Evaluation Metrics

We employ the popular metrics including Peak Signal to Noise Ratio (PSNR), Structural Similarity Index Measure (SSIM) [11], Learned Perceptual Image Patch Similarity (LPIPS) [14], and Frechet Video Distance (FVD) [9], similar to previous works [5, 10]. Note that PSNR measures pixel-wise differences between two videos, and SSIM measures the similarity between two videos in brightness, contrast, and structure, and then Perceptual similarity has been recently demonstrated to be an effective method for comparing the similarity of videos based on the convolutional feature distance between the prediction and the ground-truth. We use the same FVD evaluation metric with 16 frames for each video.

Shadow Statistics We perform a statistical analysis on the DAVIS and YouTube-VOS datasets using ViShadow to detect shadow pairs. Specifically, we compute the average number of frame pairs per video and the average duration (in frames) that the target object is present in the video. We also report the same statistics for our SOBA-VID dataset for comparison, shown as Fig. 1.

Table 1: Statistics of shadow pairs detected by ViShadow across different datasets. We report the average number of frame pairs per video and the average duration (in frames) during which the target object is present.

Dataset	Avg. Pairs per Video	Avg. pair Presence Duration (frames)
DAVIS [7]	0.9	38.5
YouTube-VOS [13]	1.2	24.2
SOBA-VID [12]	2.2	20.4

C Additional Comparison Results

Our supplementary materials provide additional experimental results and comparisons with other inpainting and outpainting methods to better evaluate our approach. Specifically, we compare with ProPainter [15] and Infinite-Canvas [3], two recent works in the domain of video content generation and editing.

ProPainter is a state-of-the-art (SOTA) video inpainting method. Its core idea is to first perform flow completion to recover the optical flow information in the video. Then, guided by the completed optical flow, it propagates information to inpaint the occluded or missing regions in the video frames. Compared to directly generating missing content, this two-stage approach better preserves temporal consistency and structural coherence, leading to higher-quality video inpainting results. We further evaluate ProPainter in an outpainting setting, where horizontal masks are applied to the left and right 25% of the frame, as well as a setting with 66% of the frame masked, and report the average performance, shown as Tab. 2. This demonstrates that inpainting methods struggle when large portions of the frame are masked.

Infinite-Canvas, on the other hand, targets high-resolution video outpainting, incorporating global layout awareness and relative positional information between image regions. However, a fair comparison with Infinite-Canvas is challenging for two main reasons. First, while both our method and MOTIA use BLIP [6] to automatically generate prompts based on video content, Infinite-Canvas adopts a different visual-language model, Qwen-VL [1], for prompt generation. Second, Infinite-Canvas is trained on a randomly selected subset (approximately 1M samples) of the Panda-70M dataset, which is not commonly used in other approaches. Despite these differences, we adapt our method to match the settings of Infinite-Canvas as closely as possible to enable a meaningful comparison, as shown in Tab. 3. For Infinite-Canvas, we directly report the results as provided in their original paper. Note that due to differences in experimental settings, our quantitative results here may differ from those reported in the main paper.

Table 2: Quantitative comparison with the SOTA inpainting method (ProPainter) under outpainting settings, where the left and right 25% and 66% of each frame are masked. Results are reported on DAVIS [7] and YouTube-VOS [13]. \uparrow indicates higher is better, while \downarrow indicates lower is better.

Method	DAVIS				YouTube-VOS			
	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	FVD \downarrow	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	FVD \downarrow
ProPainter [15]	18.35	0.7103	0.2435	352.1	20.07	0.7021	0.1870	71.23
Ours	20.81	0.7254	0.1842	234.7	20.32	0.7719	0.1793	40.78

Table 3: Comparison with Infinite-Canvas under its video outpainting settings on DAVIS [7]. Results for Infinite-Canvas are taken directly from their paper.

Method	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	FVD \downarrow
Infinite-Canvas [3]	20.80	0.726	0.160	242.8
Ours	20.83	0.730	0.181	232.8

D Outpainting on in-the-wild scene videos

To evaluate the temporal and spatial consistency of our method in more complex and unconstrained settings, we conduct outpainting on in-the-wild scene videos. As shown in Fig. 1, our approach generates extended regions that are not only visually coherent with the original frames but also exhibit smooth temporal transitions across consecutive frames.

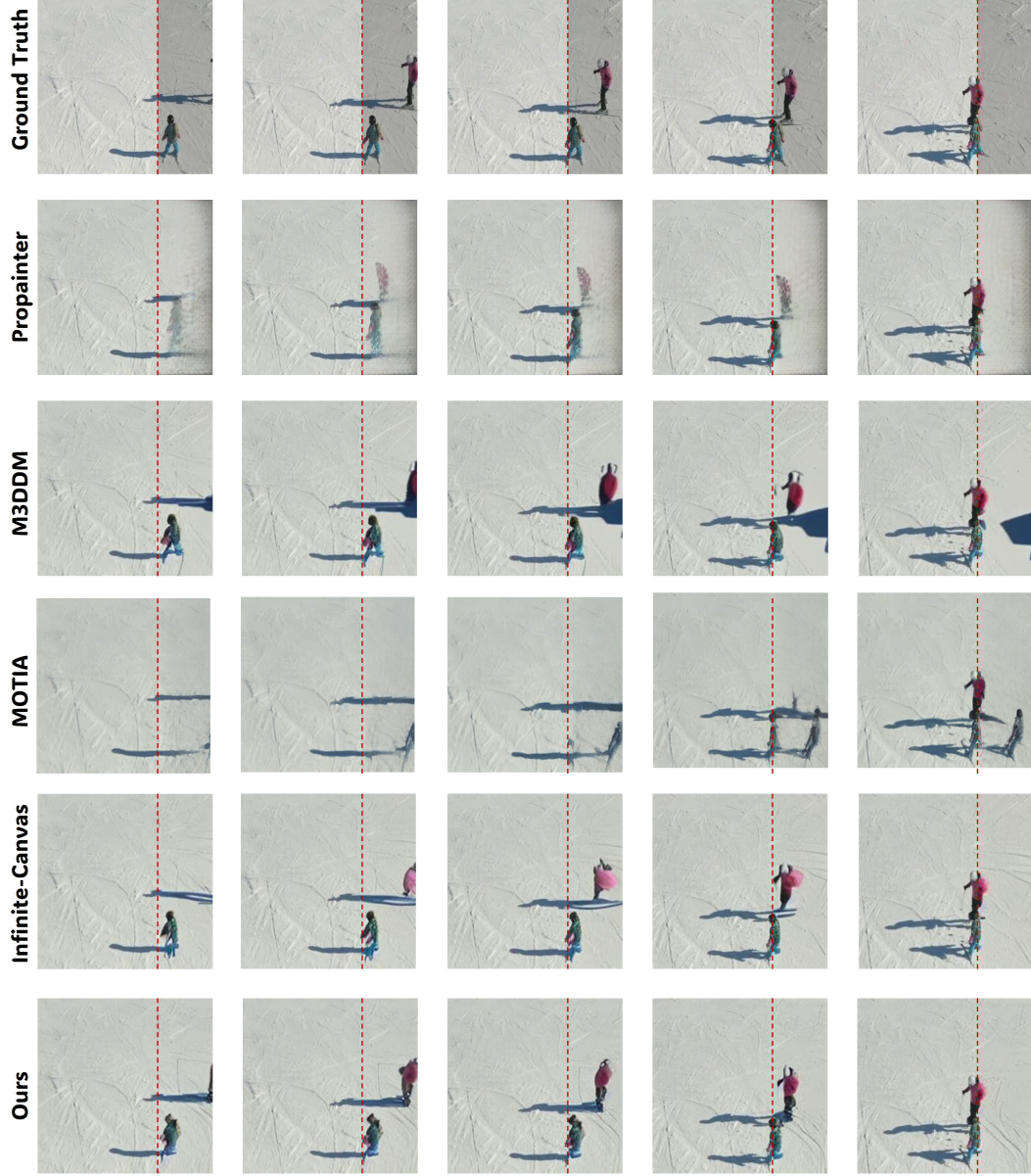


Figure 1: Outpainting results on in-the-wild scene videos. Our method produces visually coherent extensions of the original frames while maintaining smooth temporal consistency across sequences, demonstrating robustness in complex, unconstrained environments.

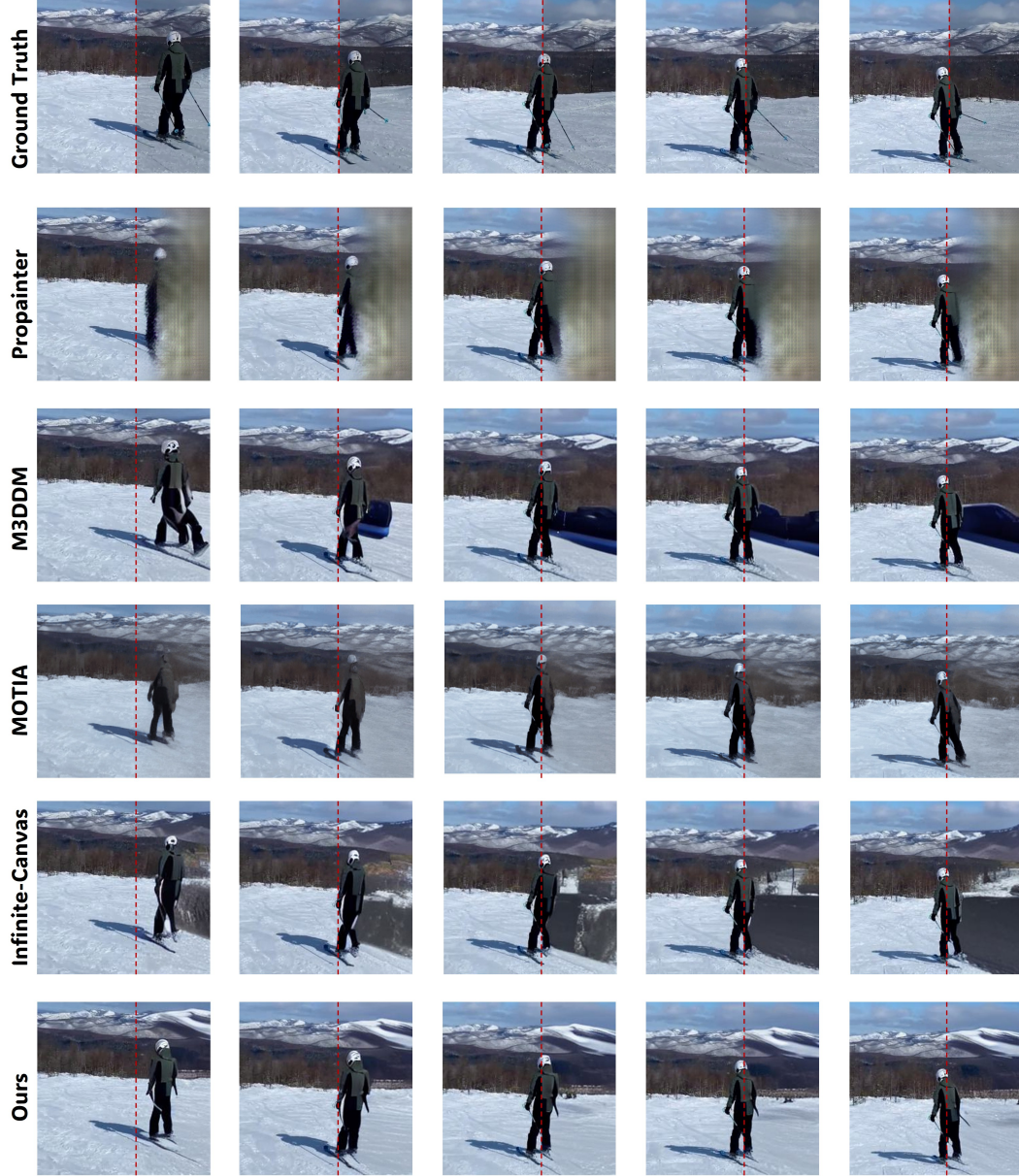


Figure 2: Outpainting results on in-the-wild scene videos. Our method produces visually coherent extensions of the original frames while maintaining smooth temporal consistency across sequences, demonstrating robustness in complex, unconstrained environments.

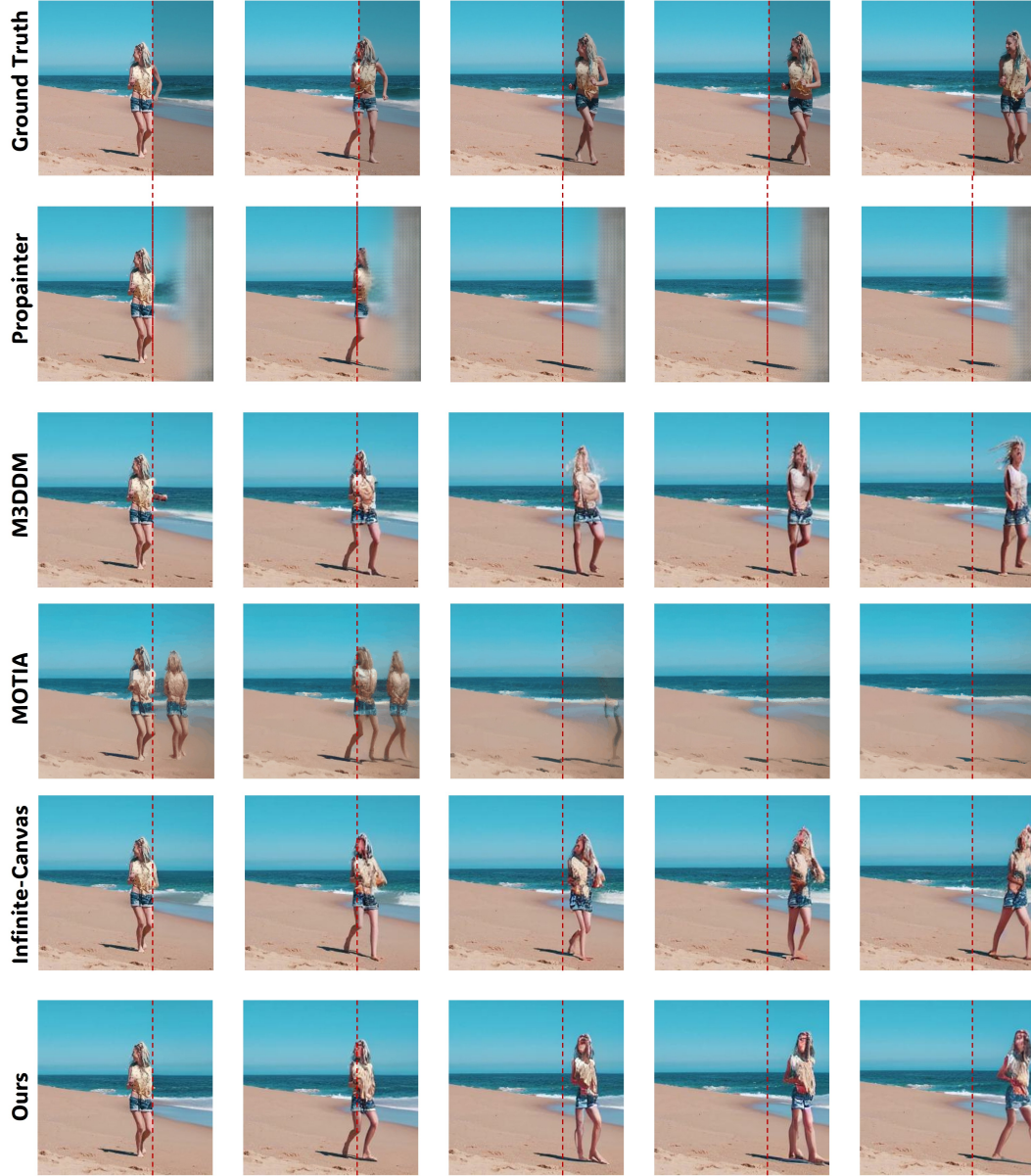


Figure 3: Outpainting results on in-the-wild scene videos. Our method produces visually coherent extensions of the original frames while maintaining smooth temporal consistency across sequences, demonstrating robustness in complex, unconstrained environments.

References

- [1] Jinze Bai et al. “Qwen-VL: A Frontier Large Vision-Language Model with Versatile Abilities”. In: *arXiv preprint arXiv:2308.12966* (2023).
- [2] Andreas Blattmann et al. “Align your latents: High-resolution video synthesis with latent diffusion models”. In: *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2023, pp. 22563–22575.
- [3] Qihua Chen et al. “Infinite-Canvas: Higher-Resolution Video Outpainting with Extensive Content Generation”. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 39. 2. 2025, pp. 2150–2158.
- [4] Alexey Dosovitskiy et al. “An image is worth 16x16 words: Transformers for image recognition at scale”. In: *arXiv preprint arXiv:2010.11929* (2020).
- [5] Fanda Fan et al. “Hierarchical masked 3d diffusion model for video outpainting”. In: *Proceedings of the 31st ACM International Conference on Multimedia*. 2023, pp. 7890–7900.
- [6] Junnan Li et al. “Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation”. In: *International conference on machine learning*. PMLR. 2022, pp. 12888–12900.
- [7] Federico Perazzi et al. “A benchmark dataset and evaluation methodology for video object segmentation”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 724–732.
- [8] Peter Shaw, Jakob Uszkoreit, and Ashish Vaswani. “Self-attention with relative position representations”. In: *arXiv preprint arXiv:1803.02155* (2018).
- [9] Thomas Unterthiner et al. “Towards accurate generative models of video: A new metric & challenges”. In: *arXiv preprint arXiv:1812.01717* (2018).
- [10] Fu-Yun Wang et al. “Be-your-outpainter: Mastering video outpainting through input-specific adaptation”. In: *European Conference on Computer Vision*. Springer. 2025, pp. 153–168.
- [11] Zhou Wang et al. “Image quality assessment: from error visibility to structural similarity”. In: *IEEE transactions on image processing* 13.4 (2004), pp. 600–612.
- [12] Zhenghao Xing et al. “Video Instance Shadow Detection Under the Sun and Sky”. In: *IEEE Transactions on Image Processing* (2024).
- [13] Ning Xu et al. “Youtube-vos: A large-scale video object segmentation benchmark”. In: *arXiv preprint arXiv:1809.03327* (2018).
- [14] Richard Zhang et al. “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.
- [15] Shangchen Zhou et al. “Propainter: Improving propagation and transformer for video inpainting”. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2023, pp. 10477–10486.