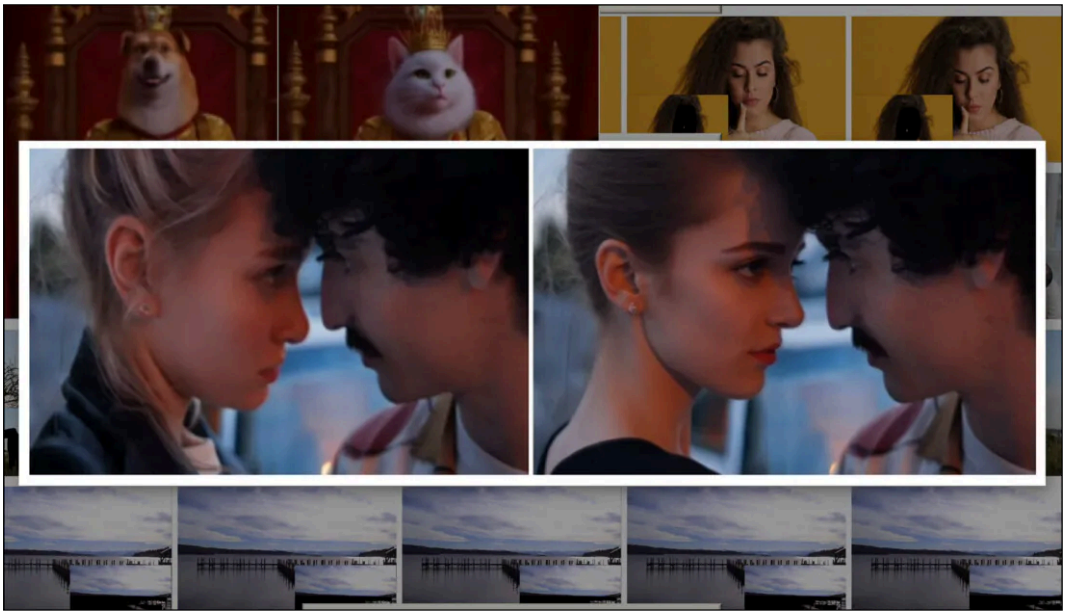
The Road to Better AI-Based Video Editing

Originally published at https://www.unite.ai/the-road-to-better-ai-based-video-editing/



The video/image synthesis research sector regularly outputs video-editing* architectures, and over the last nine months, outings of this nature have become even more frequent. That said, most of them represent only incremental advances on the state of the art, since the core challenges are substantial.

However, a new collaboration between China and Japan this week has produced some examples that merit a closer examination of the approach, even if it is not necessarily a landmark work.

In the video-clip below (from the paper's associated project site, that – be warned – may tax your browser) we see that while the deepfaking capabilities of the system are non-existent in the current configuration, the system does a fine job of plausibly and significantly altering the identity of the young woman in the picture, based on a video mask (bottom-left):

CLICK TO PLAY VIDEO ON SITE

Click to play. *Based on the semantic segmentation mask visualized in the lower left, the original (upper left) woman is transformed into a notably different identity, even though this process does not achieve the identity-swap indicated in the prompt.* Source: https://yxbian23.github.io/project/video-painter/ *(be aware that at the time of writing, this autoplaying and video-stuffed site was inclined to crash my browser). Please refer to the source videos, if you can access them, for better resolution and detail, or check out the examples at the project's overview video at https://www.youtube.com/watch?v=HYzNfsD3A0s*

Mask-based editing of this kind is well-established in static latent diffusion models, using tools like ControlNet. However, maintaining background consistency in video is far more challenging, even when masked areas provide the model with creative flexibility, as shown below:

CLICK TO PLAY VIDEO ON SITE

Click to play. *A change of species, with the new VideoPainter method. Please refer to the source videos, if you can access them, for better resolution and detail, or check out the examples at the project's overview video at https://www.youtube.com/watch?v=HYzNfsD3A0s*

The authors of the new work consider their method in regard both to Tencent's own BrushNet architecture (which we covered last year), and to ControlNet, both of which treat of a dual-branch architecture capable of isolating the foreground and background generation.

However, applying this method directly to the very productive Diffusion Transformers (DiT) approach proposed by OpenAI's Sora, brings particular challenges, as the authors note"

> *'[Directly] applying [the architecture of BrushNet and ControlNet] to video DiTs presents several challenges: [Firstly, given] Video DiT's robust generative foundation and heavy model size, replicating the full/half-giant Video DiT backbone as the context encoder would be unnecessary and computationally prohibitive.*
>
> *'[Secondly, unlike] BrushNet's pure convolutional control branch, DiT's tokens in masked regions inherently contain background information due to global attention, complicating the distinction between masked and unmasked regions in DiT backbones.*
>
> *'[Finally,] ControlNet lacks feature injection across all layers, hindering dense background control for inpainting tasks.'*

Therefore the researchers have developed a plug-and-play approach in the form of a dual-branch framework titled *VideoPainter*.

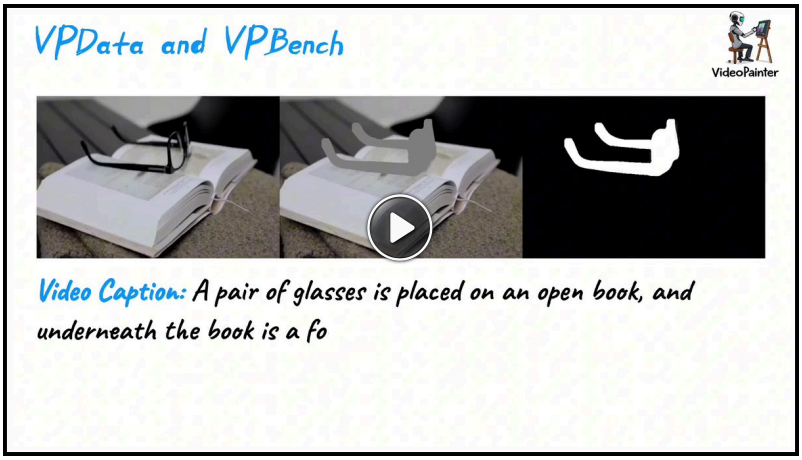
VideoPainter offers a dual-branch video inpainting framework that enhances pre-trained DiTs with a lightweight context encoder. This encoder accounts for just 6% of the backbone's parameters, which the authors claim makes the approach more efficient than conventional methods.

The model proposes three key innovations: a streamlined two-layer context encoder for efficient background guidance; a mask-selective feature integration system that separates masked and unmasked tokens; and an inpainting region ID resampling technique that maintains identity consistency across long video sequences.

By freezing both the pre-trained DiT and context encoder while introducing an ID-Adapter, VideoPainter ensures that inpainting region tokens from previous clips persist throughout a video, reducing flickering and inconsistencies.

The framework is also designed for plug-and-play compatibility, allowing users to integrate it seamlessly into existing video generation and editing workflows.

To support the work, which uses CogVideo-5B-I2V as its generative engine, the authors curated what they state is the largest video inpainting dataset to date. Titled *VPData*, the collection consists of more than 390,000 clips, for a total video duration of more than 886 hours. They also developed a related benchmarking framework titled *VPBench*.



CLICK TO PLAY VIDEO ON SITE

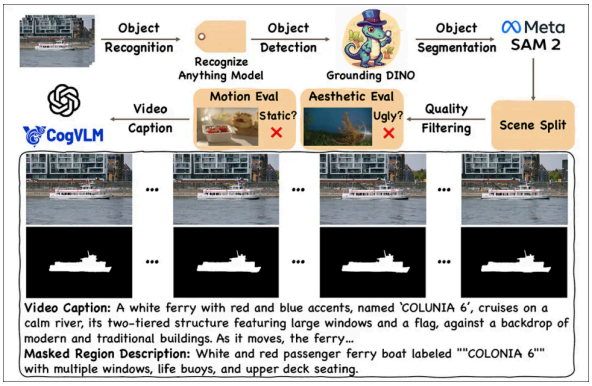
Click to play. *From the project website examples, we see the segmentation capabilities powered by the VPData collection and the VPBench test suite. Please refer to the source videos, if you can access them, for better resolution and detail, or check out the examples at the project's overview video at https://www.youtube.com/watch?v=HYzNfsD3A0s*

The new work is titled *VideoPainter: Any-length Video Inpainting and Editing with Plug-and-Play Context Control*, and comes from seven authors at the Tencent ARC Lab, The Chinese University of Hong Kong, The University of Tokyo, and the University of Macau.

Besides the aforementioned project site, the authors have also released a more accessible YouTube overview, as well a Hugging Face page.

Method

The data collection pipeline for VPData consists of collection, annotation, splitting, selection and captioning:



Schema for the dataset construction pipeline. Source: https://arxiv.org/pdf/2503.05639

The source collections used for this compilation came from Videvo and Pexels, with an initial haul of around 450,000 videos obtained.

Multiple contributing libraries and methods comprised the pre-processing stage: the Recognize Anything framework was used to provide open-set video tagging, tasked with identifying primary objects; Grounding Dino was used for the detection of bounding boxes around the identified objects; and the Segment Anything Model 2 (SAM 2) framework was used to refine these coarse selections into high-quality mask segmentations.

To manage scene transitions and ensure consistency in video inpainting, VideoPainter uses PySceneDetect to identify and segment clips at natural breakpoints, avoiding the disruptive shifts often caused by tracking the same object from multiple angles. The clips were divided into 10-second intervals, with anything shorter than six seconds discarded.

For data selection, three filtering criteria were applied: *aesthetic quality*, assessed with the Laion-Aesthetic Score Predictor; *motion strength*, measured via optical flow using RAFT; and *content safety*, verified through Stable Diffusion's Safety Checker.

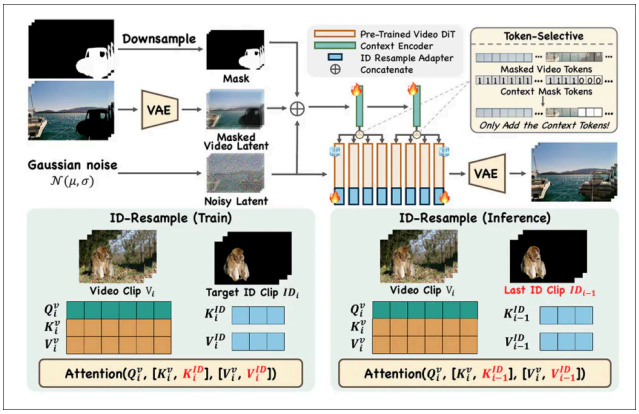
One major limitation in existing video segmentation datasets is the lack of detailed textual annotations, which are crucial for guiding generative models:

Dataset	#Clips	Duration	Video Caption	Masked Region Desc.
DAVIS [Perazzi et al. 2016]	0.4K	0.1h	✗	✗
YouTube-VOS [Xu et al. 2018]	4.5K	5.6h	✗	✗
VOST [Tokmakov et al. 2023]	1.5K	4.2h	✗	✗
MOSE [Ding et al. 2023]	5.2K	7.4h	✗	✗
LVOS [Hong et al. 2023]	1.0K	18.9h	✗	✗
SA-V [Ravi et al. 2024]	642.6K	196.0h	✗	✗
Ours	390.3K	866.7h	✓	✓

The researchers emphasize the lack of video-captioning in comparable collections.

Therefore the VideoPainter data curation process incorporates diverse leading vision-language models, including CogVLM2 and Chat GPT-4o to generate keyframe-based captions and detailed descriptions of masked regions.

VideoPainter enhances pre-trained DiTs by introducing a custom lightweight context encoder that separates background context extraction from foreground generation, seen to the upper right of the illustrative schema below:



Conceptual schema for VideoPainter. VideoPainter's context encoder processes noisy latents, downsampled masks, and masked video latents via VAE, integrating only background tokens into the pre-trained DiT to avoid ambiguity. The ID Resample Adapter ensures identity consistency by concatenating masked region tokens during training and resampling them from previous clips during inference.

Instead of burdening the backbone with redundant processing, this encoder operates on a streamlined input: a combination of noisy latent, masked video latent (extracted via a variational autoencoder, or VAE), and downsampled masks.

The noisy latent provides generation context, and the masked video latent aligns with the DiT's existing distribution, aiming to enhance compatibility.

Rather than duplicating large sections of the model, which the authors state has occurred in prior works, VideoPainter integrates only the first two layers of the DiT. These extracted features are reintroduced into the frozen DiT in a structured, group-wise manner – early-layer features inform the initial half of the model, while later features refine the second half.

Additionally, a token-selective mechanism ensures that only background-relevant features are reintegrated, preventing confusion between masked and unmasked regions. This approach, the authors contend, allows VideoPainter to maintain high fidelity in background preservation while improving foreground inpainting efficiency.

The authors note that the method they proposes supports diverse stylization methods, including the most popular, Low Rank Adaptation (LoRA).

Data and Tests

VideoPainter was trained using the CogVideo-5B-I2V model, along with its text-to-video equivalent. The curated VPData corpus was used at 480x720px, at a learning rate of 1×10^{-5}.

The ID Resample Adapter was trained for 2,000 steps, and the context encoder for 80,000 steps, both using the AdamW optimizer. The training took place in two stages using a formidable 64 NVIDIA V100 GPUs (though the paper does not specify whether these had 16GB or 32GB of VRAM).

For benchmarking, Davis was used for random masks, and the authors' own VPBench for segmentation-based masks.

The VPBench dataset features objects, animals, humans, landscapes and diverse tasks, and covers four actions: *add*, *remove*, *change*, and *swap*. The collection features 45 6-second videos, and nine videos lasting, on average, 30 seconds.

Eight metrics were utilized for the process. For Masked Region Preservation, the authors used Peak Signal-to-Noise Ratio (PSNR); Learned Perceptual Similarity Metrics (LPIPS); Structural Similarity Index (SSIM); and Mean Absolute Error (MAE).

For text-alignment, the researchers used CLIP Similarity both to evaluate semantic distance between the clip's caption and its actual perceived content, and also to evaluate accuracy of masked regions.

To assess the general quality of the output videos, Fréchet Video Distance (FVD) was used.

For a quantitative comparison round for video inpainting, the authors set their system against prior approaches ProPainter, COCOCO and Cog-Inp (CogVideoX). The test consisted of inpainting the first frame of a clip using image inpainting models, and then using an image-to-video (I2V) backbone to propagate the results into a latent blend operation, in accord with a method proposed by a 2023 paper from Israel.

Since the project website is not entirely functional at the time of writing, and since the project's associated YouTube video may not feature the entirety of examples stuffed into the project site, it is rather difficult to locate video examples that are very specific to the results outlined in the paper. Therefore we will show partial static results featured in the paper, and close the article with some additional video examples that we managed to extract from the project site.

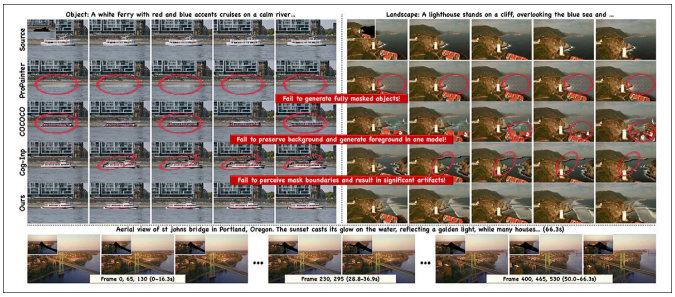
Metrics		Masked Region Preservation					Text Alignment		Video Quality
	Models	PSNR↑	SSIM↑	LPIPS$_{\times10^2}$↓	MSE$_{\times10^2}$↓	MAE$_{\times10^2}$↓	CLIP Sim↑	CLIP Sim (M)↑	FVID↓
VPBench-S	ProPainter	20.97	0.87	9.89	1.24	3.56	7.31	17.18	0.44
	COCOCO	19.27	0.67	14.80	1.62	6.38	7.95	20.03	0.69
	Cog-Inp	22.15	0.82	9.56	0.88	3.92	8.41	21.27	0.18
	Ours	23.32	0.89	6.85	0.82	2.62	8.66	21.49	0.15
VPBench-L	ProPainter	20.11	0.84	11.18	1.17	3.71	9.44	17.68	0.48
	COCOCO	19.51	0.66	16.17	1.29	6.02	11.00	20.42	0.62
	Cog-Inp	19.78	0.73	12.53	1.33	5.13	11.47	21.22	0.21
	Ours	22.19	0.85	9.14	0.71	2.92	11.52	21.54	0.17
Davis	ProPainter	23.99	0.92	5.86	0.98	2.48	7.54	16.69	0.12
	COCOCO	21.34	0.66	10.51	0.92	4.99	6.73	17.50	0.33
	Cog-Inp	23.92	0.79	10.78	0.47	3.23	7.03	17.53	0.17
	Ours	25.27	0.94	4.29	0.45	1.41	7.21	18.46	0.09

Quantitative comparison of VideoPainter vs. ProPainter, COCOCO, and Cog-Inp on VPBench (segmentation masks) and Davis (random masks). Metrics cover masked region preservation, text alignment, and video quality. Red = best, Blue = second best.

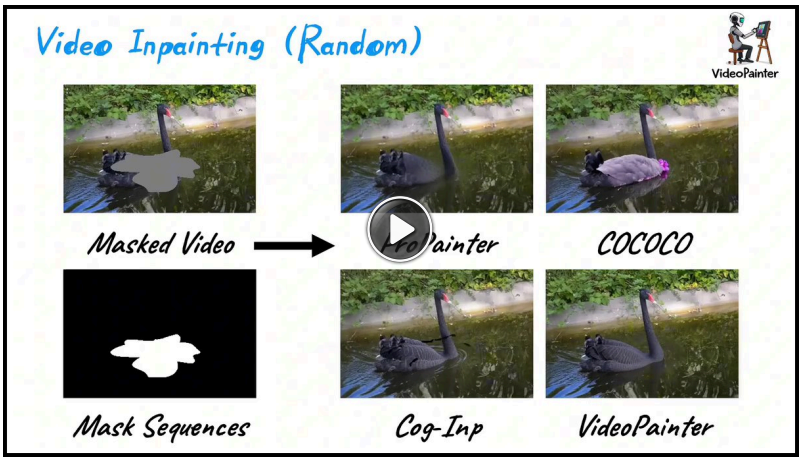
Of these qualitative results, the authors comment:

> *'In the segmentation-based VPBench, ProPainter, and COCOCO exhibit the worst performance across most metrics, primarily due to the inability to inpaint fully masked objects and the single-backbone architecture's difficulty in balancing the competing background preservation and foreground generation, respectively.*
>
> *'In the random mask benchmark Davis, ProPainter shows improvement by leveraging partial background information. However, VideoPainter achieves optimal performance across segmentation (standard and long length) and random masks through its dual-branch architecture that effectively decouples background preservation and foreground generation.'*

The authors then present static examples of qualitative tests, of which we feature a selection below. In all cases we refer the reader to the project site and YouTube video for better resolution.



A comparison against inpainting methods in prior frameworks.



CLICK TO PLAY VIDEO ON SITE

Click to play. *Examples concatenated by us from the 'results' videos at the project site.*

Regarding this qualitative round for video inpainting, the authors comment:

> *'VideoPainter consistently shows exceptional results in the video coherence, quality, and alignment with text caption. Notably, ProPainter fails to generate fully masked objects because it only depends on background pixel propagation instead of generating.*
>
> *'While COCOCO demonstrates basic functionality, it fails to maintain consistent ID in inpainted regions (inconsistent vessel appearances and abrupt terrain changes) due to its single-backbone architecture attempting to balance background preservation and foreground generation.*
>
> *'Cog-Inp achieves basic inpainting results; however, its blending operation's inability to detect mask boundaries leads to significant artifacts.*
>
> *'Moreover, VideoPainter can generate coherent videos exceeding one minute while maintaining ID consistency through our ID resampling.'*

The researchers additionally tested VideoPainter's ability to augment captions and obtain improved results by this method, putting the system against UniEdit, DiTCtrl, and ReVideo.

	Metrics	Masked Region Preservation					Text Alignment		Video Quality
	Models	PSNR↑	SSIM↑	LPIPS$_{\times 10^2}$↓	MSE$_{\times 10^2}$↓	MAE$_{\times 10^2}$↓	CLIP Sim↑	CLIP Sim (M)↑	FVID↓
Standard	UniEdit	9.96	0.36	56.68	11.08	25.78	8.46	14.23	1.36
	DitCtrl	9.30	0.33	57.42	12.73	27.45	8.52	15.59	0.57
	ReVideo	15.52	0.49	27.68	3.49	11.14	9.34	20.01	0.42
	Ours	22.63	0.91	7.65	1.02	2.90	8.67	20.20	0.18
Long	UniEdit	10.37	0.30	54.61	10.25	24.89	10.85	15.42	1.00
	DitCtrl	9.76	0.28	62.49	11.50	26.64	11.78	16.52	0.56
	ReVideo	15.50	0.46	28.57	3.92	12.24	11.22	20.50	0.35
	Ours	22.60	0.90	7.53	0.86	2.76	11.85	19.38	0.11

Video-editing results against three prior approaches.

The authors comment:

> *'For both standard and long videos in VPBench, VideoPainter achieves superior performance, even surpassing the end-to-end ReVideo. This success can be attributed to its dual-branch architecture, which ensures excellent background preservation and foreground generation capabilities, maintaining high fidelity in non-edited regions while ensuring edited regions closely align with editing instructions, complemented by inpainting region ID resampling that maintains ID consistency in long video.'*

Though the paper features static qualitative examples for this metric, they are unilluminating, and we refer the reader instead to the diverse examples spread across the various videos published for this project.

Finally, a human study was conducted, where thirty users were asked to evaluate 50 randomly-selected generations from the VPBench and editing subsets. The examples highlighted background preservation, alignment to prompt, and general video quality.

Task	Video Inpainting			Video Editing		
	Background Preservation	Text Alignment	Video Quality	Background Preservation	Text Alignment	Video Quality
Ours	74.2%	82.5%	87.4%	78.4%	76.1%	81.7%

Results from the user-study for VideoPainter.

The authors state:

> *'VideoPainter significantly outperformed existing baselines, achieving higher preference rates across all evaluation criteria in both tasks.'*

They concede, however, that the quality of VideoPainter's generations depends on the base model, which can struggle with complex motion and physics; and they observe that it also performs poorly with low-quality masks or misaligned captions.

Conclusion

VideoPainter seems a worthwhile addition to the literature. Typical of recent solutions, however, it has considerable compute demands. Additionally, many of the examples chosen for presentation at the project site fall very far short of the best examples; it would therefore be interesting to see this framework pitted against future entries, and a wider range of prior approaches.

** It is worth mentioning that 'video-editing' in this sense does not mean 'assembling diverse clips into a sequence', which is the traditional meaning of this term; but rather directly changing or in some way modifying the inner content of existing video clips, using machine learning techniques*

First published Monday, March 10, 2025. Updated Saturday, July 25, 2026 to replace video.

Writer on machine learning, domain specialist in human image synthesis. Former head of research content at Metaphysic.ai, until its dissolution into DNEG's Brahma.ai.
Portfolio site: martinanderson.ai
Contact: martin@martinanderson.ai