
ActAnywhere: Subject-Aware Video Background Generation

Boxiao Pan*
Stanford University

Zhan Xu
Adobe Research

Chun-Hao Paul Huang
Adobe Research

Krishna Kumar Singh
Adobe Research

Yang Zhou
Adobe Research

Leonidas J. Guibas
Stanford University

Jimei Yang
Runway

Abstract

We study a novel problem to automatically generate video background that tailors to foreground subject motion. It is an important problem for the movie industry and visual effects community, which traditionally requires tedious manual efforts to solve. To this end, we propose ActAnywhere, a video diffusion model that takes as input a sequence of foreground subject segmentation together with an image of a novel background, and generates a video of the subject interacting in this background. We train our model on a large-scale dataset of 2.4M videos of human-scene interactions. Through extensive evaluation, we show that our model produces videos with realistic foreground-background interaction while strictly following the guidance of the condition image. Our model generalizes to diverse scenarios including non-human subjects, gaming and animation clips, as well as videos with multiple moving subjects. Both quantitative and qualitative comparisons demonstrate that our model significantly outperforms existing methods, which fail to accomplish the studied task. Please visit our project webpage at <https://actanywhere.github.io>.

1 Introduction

Compositing an acting video of a subject onto a novel background is central for creative story-telling in filmmaking and visual effects. The key requirement is seamlessly integrating the foreground subject with the background in terms of camera motions, interactions, lighting and shadows, so that the composition looks realistic and vivid as if the subject acts physically in the scene. In movie industry, this process is often conducted by virtual production [1] that requires artists to first create a 3D scene and then to film the acting video in an LED-walled studio or to render the video in 3D engines. This process is not only tedious and expensive, but most importantly, prevents artists from quickly iterating their ideas.

Inspired by this artistic workflow, we study the novel problem of *automated* subject-aware video background generation. As shown in fig. 1, given a foreground segmentation sequence that provides the subject motion, as well as a condition frame that describes a scene, we aim to generate a video that adapts the subject to this scene with realistically synthesized foreground-background interaction. This condition frame can be either a background-only image, or a composite frame consisting of both background and foreground created via photo editing tools [3] or generative image model [32].

This problem, at its core, requires retaining part of the input video while generating the rest to adapt to it. To the best of our knowledge, the closest works to this setting are those on video editing and inpainting / outpainting. Video editing methods assume a source video as input and make edits

*Work done during an internship at Adobe.



Figure 1: Given a sequence of foreground segmentation as input (top), and one frame that describes the background as the condition (left), ActAnywhere generates coherent video background that adapts to the subject motion. We show two subjects here, each with two generated samples. ActAnywhere is able to generate videos consistent with the condition frame with highly realistic details such as splatting water, moving smoke and flame, shadows, duck feet, etc. It generalizes to a diverse distribution of subjects and backgrounds, including non-human subjects. Our method works with both composited frames and background-only images as the condition.

based on condition signals, such as natural language or image. In comparison, our task poses unique challenges that these methods cannot solve. First, the foreground of the video needs to be retained and sometimes refined on the boundary (as shown in fig. 8 in appendix), while prior works generally perform holistic changes to the entire video [5, 8, 13, 16, 22, 24, 42]. Second, existing methods most commonly condition the editing process on a short text prompt [5, 8, 13, 16, 22, 24, 42], which only poses loose constraints that are not enough for artists’ creative intentions. Those that can condition on an image [13] are not able to generate contents that strictly follow the guidance. We demonstrate such comparisons in fig. 4. On the other hand, video inpainting / outpainting methods [14, 43, 46, 49] aim to perform context-aware removal / expanding for a video. However, these methods focus on pixel harmonization, while our emphasis is on taking additional control signal that indicates the desired background, and synthesizing large and dynamic background region with reasonable interaction with the given foreground.

To this end, we propose a video diffusion model that explicitly controls the foreground motion with the foreground segmentation sequence, and additionally, conditions the generation on an image that describes the desired background. We train the model with a designed self-supervised learning procedure that takes the foreground segmentation sequence of the input video to predict the original video, conditioned on a randomly sampled frame. This training procedure enables the model to 1) retain the foreground subject; 2) hallucinate missing details from imperfect segmentation; and 3) adhere to the guidance of the condition frame while being robust to the subject’s pose in it.

Moreover, we propose to express the condition signal as the CLIP features of the image, which we show empirically outperforms its alternatives. We also concatenate a background mask with the segmentation as the input to better indicate the region to be generated.

We train our model on a large-scale dataset [26] that consists of 2.4M videos of human-scene interactions and evaluate both on a held-out set as well as on videos from DAVIS [30]. ActAnywhere is able to generate highly realistic videos that follow the condition frame, and at the same time synthesizes video background that conforms to the foreground motion. Notably, despite being trained solely on videos of humans, ActAnywhere generalizes to non-human subjects, such as animals and man-made objects, in a zero-shot manner.

In summary, our contributions are:

1. We introduce a novel problem of automated subject-aware video background generation.
2. We propose ActAnywhere, a video diffusion model to solve this task, and train it on a large-scale human-scene interaction video dataset in a self-supervised manner.
3. Extensive evaluations demonstrate that our model generates coherent videos with realistic subject-scene interactions, camera motions, lighting and shadows, and generalizes to out-of-distribution data including non-human subjects, gaming and animation clips, and videos with multiple moving subjects.

2 Related Work

Video generation. There have been a long thread of works on video generation. The core architecture has evolved from GANs [11, 38, 41] to more recent transformers [15, 40, 46, 49] and diffusion models [6, 9, 17, 19, 20, 24, 47]. Below we review the most related diffusion-based works. Most of these works leverage temporal self-attention blocks inside the denoising U-Net in order to acquire temporal awareness. On top of that, Text2Video-Zero [24] introduces additional noise scheduling to correlate the latents in a video. LVDM [19] and Align Your Latents [6] both design a hierarchical approach to generate longer-term videos. Align Your Latents additionally fine-tunes a spatial super-resolution model for high-resolution video generation. AnimateDiff [17] proposes to train the temporal attention blocks on a large-scale video dataset, which can then be inserted into any text-to-image diffusion models (given that the architecture fits) to turn that into a text-to-video model, in a zero-shot manner. VideoCrafter1 [9] further uses dual attention to enable joint text and image-conditioned generation. These works focus on unconditional generation or with text or image conditioning, but are not able to follow the guidance of additional foreground motion.

Video editing. Another thread studies the problem of video editing, where a source video is given as input, and edits are performed according to some condition signals. Text2Live [5] uses pre-trained video atlases of the input video, and performs text-guided edits on the foreground or background. Gen1 [13] leverages depth maps estimated by a pre-trained network [33] as an additional condition to improve the structural consistency. Tune-A-Video [42] proposes to finetune only part of the spatial-attention blocks and all of the temporal-attention blocks on a single input video. TokenFlow [16] uses latent nearest neighbor fields computed from the input video to propagate edited features across all frames. Both VideoControlNet [22] and Control-A-Video [10] adopt a ControlNet [48]-like approach to condition the video diffusion process with additional signals such as depth maps or Canny edges extracted from the input video. As stated above, these works apply holistic changes to the entire video and are not able to retain the subject. Moreover, they either take only loose constraints from text conditioning, or are not able to strictly follow the image guidance.

One major downside of these works is hence that the generated videos tend to keep the spatial structure from the source video, which limits the edits that the model can perform to stylistic changes. In our work, we propose to condition on the foreground segmentation for the motion, while extract the background information only from one condition frame. In particular, using the masked foreground as input endows a nice separation as in what to preserve and what to generate.

Image and video inpainting. Image / video inpainting aims to fill a missing region, often expressed as a mask. Image inpainting methods either take condition signals such as natural language and image [34, 44, 45], or rely solely on the context outside the masked region [14, 36, 46, 49]. Recent diffusion-based image inpainting methods use a combination of masked image and the mask itself, and condition the diffusion process either on natural language [34, 44] or an image of the condition

object [45], or perform unconditional diffusion [36]. For video inpainting, MAGVIT [46] proposes a generative video transformer trained through masked token prediction, and is able to inpaint small masked regions afterwards. ProPainter [49] designs a flow-based method by propagating pixels and features through completed flows. M3DDM [14] leverages a video diffusion model, and conditions the diffusion process on global video features extracted by a video encoder. Different from these works, we aim to generate large background regions that strictly follow the condition frame. Moreover, the generated background needs to adapt to the foreground subject motion in a coherent way. This poses significant challenges that previous inpainting methods cannot tackle.

3 Method

We first provide essential preliminary background on latent diffusion in section 3.1. We then formally define our problem in section 3.2 and delve into our model design in section 3.3. Finally, we specify the training details in section 3.4.

3.1 Preliminaries on Latent Diffusion Models

Diffusion models such as DDPM [21], encapsulate a forward process of adding noise and a backward process of denoising. Given a diffusion time step τ , the forward process incrementally introduces Gaussian noises into the data distribution $x_0 \sim q(x_0)$ via a Markov chain, following a predefined variance schedule denoted as β :

$$q(\mathbf{x}_\tau | \mathbf{x}_{\tau-1}) = \mathcal{N}(\mathbf{x}_\tau; \sqrt{1 - \beta_\tau} \mathbf{x}_{\tau-1}, \beta_\tau \mathcal{I}) \quad (1)$$

For the backward process, a U-Net [35] ϵ_θ is trained to denoise \mathbf{x}_τ and recover the original data distribution:

$$p_\theta(\mathbf{x}_{\tau-1} | \mathbf{x}_\tau) = \mathcal{N}(\mathbf{x}_{\tau-1}; \boldsymbol{\mu}_\theta(\mathbf{x}_\tau, \tau), \boldsymbol{\Sigma}_\theta(\mathbf{x}_\tau, \tau)) \quad (2)$$

$\boldsymbol{\mu}_\theta$ and $\boldsymbol{\Sigma}_\theta$ are parametrized by ϵ_θ . The discrepancy between the predicted noise and the ground-truth noise is minimized as the training objective.

Stable Diffusion [34] further proposes to train the diffusion model in the latent space of a VAE [25]. Specifically, an encoder \mathcal{E} learns to compress an input image x into latent representations $z = \mathcal{E}(x)$, and a decoder \mathcal{D} learns to reconstruct the latents back to pixel space, such that $x = \mathcal{D}(\mathcal{E}(x))$. In this way, the diffusion is performed in the latent space of the VAE.

3.2 Problem Formulation

Given an input video $\mathcal{X} \in \mathbb{R}^{T \times H \times W \times 3}$ featuring a foreground subject, we first deploy a segmentation algorithm, such as Mask R-CNN [18], to obtain a subject segmentation sequence, $\mathcal{S} \in \mathbb{R}^{T \times H \times W \times 3}$, along with the corresponding masks, $\mathcal{M} \in \mathbb{R}^{T \times H \times W \times 1}$. Both \mathcal{S} and \mathcal{M} serve as input to our model. \mathcal{S} contains the segmentation of the foreground subject, with background pixels set to 127 (grey). \mathcal{M} has the foreground pixels set to 0 and background to 1. Across all our experiments, $H = W = 256$ and $T = 16$.

Additionally, we also incorporate a single condition frame $\mathbf{c} \in \mathbb{R}^{H \times W \times 3}$ describing the background that we want to generate. As shown in fig. 2, \mathbf{c} is a randomly sampled frame from \mathcal{X} at training time, while can be either a frame showing foreground-background composition or a background-only image at inference time. The goal is thus to generate an output video \mathcal{V} with the subject dynamically interacting with the synthesized background. The motivation of using an image not language as the condition is that image is a more straightforward media to carry detailed and specific information of the intended background, especially when users already have a pre-defined target scene image.

3.3 Subject-Aware Latent Video Diffusion

We build our model based on latent video diffusion models [17]. In our architecture design, we address two main questions: 1) providing the foreground subject sequence to the network to enable proper motion guidance, and 2) injecting the condition signal from the background frame to make the generated video adhere to the condition.

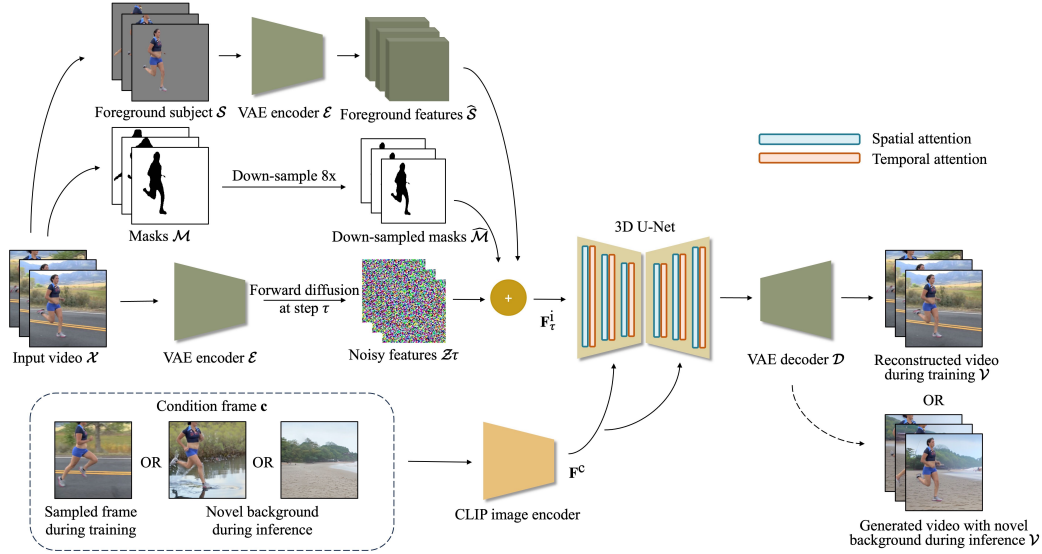


Figure 2: **Architecture overview.** During training, we take a randomly sampled frame from the training video to condition the denoising process. At test time, the condition can be either a composited frame of the subject with a novel background, or a background-only image.

We present our pipeline in fig. 2. For the foreground segmentation sequence \mathcal{S} , we use the pre-trained VAE [34] encoder \mathcal{E} to encode the foreground segmentation into latent features $\hat{\mathcal{S}} \in \mathbb{R}^{16 \times 32 \times 32 \times 4}$. We downsample the foreground mask sequence \mathcal{M} 8 times to obtain the resized mask sequence $\hat{\mathcal{M}} \in \mathbb{R}^{16 \times 32 \times 32 \times 1}$ to align with the latent features $\hat{\mathcal{S}}$. To train the denoising network ϵ_θ , we encode the original frames \mathcal{X} with the same VAE encoder into latent representation $\mathcal{Z} \in \mathbb{R}^{16 \times 32 \times 32 \times 4}$, and add noises at diffusion time step τ with the forward diffusion process denoted in eq. (1) to get noisy latent feature \mathcal{Z}_τ . We subsequently concatenate $\hat{\mathcal{S}}$, $\hat{\mathcal{M}}$ and \mathcal{Z}_τ along the feature dimension, forming a 9-channel input feature $\mathbf{F}_\tau^i \in \mathbb{R}^{16 \times 9 \times 32 \times 32}$ to the U-Net. During inference, \mathcal{Z}_0 is initialized as Gaussian noises, and gets auto-regressively denoised for multiple time steps to sample a final result, according to the backward diffusion process described in eq. (2). The denoised latents are then decoded to a video via the VAE decoder \mathcal{D} .

We build our 3D denoising U-Net based on AnimateDiff [17]. AnimateDiff works by inserting a series of motion modules in between the spatial attention layers in the denoising U-Net of a pre-trained T2I diffusion model. These motion modules consist of a few feature projection layers followed by 1D temporal self-attention blocks.

For the condition image c , we encode it with the CLIP image encoder [31] and take the features from the last hidden layer as its encoding \mathbf{F}^c . These features are then injected into the UNet ϵ_θ through its cross-attention layers, similar to [26, 34]. We empirically find that this method achieves better temporal consistency compared to other alternatives, such as using VAE features for either cross-attention or concatenation with other input features. We ablate on this in table 3.

3.4 Training

Training is supervised by a simplified diffusion objective, namely predicting the added noise [21]:

$$\mathcal{L} = \|\epsilon - \epsilon_\theta(\mathbf{F}_\tau^i, \tau, \mathbf{F}^c)\|_2^2 \quad (3)$$

where ϵ is the ground-truth noise added.

Dataset. We train on the large-scale dataset compiled and processed by [26], which we refer to as HiC+. The resulting dataset contains 2.4M videos of human-scene interactions. It also provides foreground segmentation and masks. We refer the reader to the original paper for more details.

Pre-trained weights. We initialize the weights of our denoising network ϵ_θ with the pre-trained weights from the Stable Diffusion image inpainting model [34][†], which is fine-tuned on top of the original Stable Diffusion on the text-conditioned image inpainting task. We initialize the weights of the inserted motion modules with AnimateDiff v2[‡].

For the CLIP image encoder, we use the “clip-vit-large-patch14” variant[§] provided by OpenAI, whose features from the last hidden layer have a dimension of 1024, while the pre-trained U-Net takes in features of dimension 768 as the condition, which are also in the text feature space. To account for this, we train an additional two-layer MLP to project the features into the desired space.

During training, we freeze the shared VAE and the CLIP encoder, and fine-tune the entire U-Net.

Data processing and augmentation. Obtaining perfect segmentation masks from videos is challenging. The masks may be incomplete, missing some parts of the foreground, or be excessive such that they include leaked background near the boundary. To deal with incomplete segmentation, during training, we apply random rectangular cut-outs to the foreground segmentation and masks. We provide more information on this in section 7.2 of the appendix. To reduce information leak from excessive segmentation, we perform image erosion to the segmentation and masks with a uniform kernel of size 5×5 , both during training and inference.

Random condition dropping. In order to enable classifier-free guidance at test time, we randomly drop the segmentation and the mask, the condition frame, or all of them at 10% probability each during training. In these cases we set them to zeros before passing into the respective encoders.

Other details. We use the AdamW [27] optimizer with a constant learning rate of $3e-5$. We train on 8 NVIDIA A100-80GB GPUs with batch size 4, which takes approximately a week to fully converge.

4 Experiments

We start by describing the data used for evaluation. We then show diverse samples generated from our method in section 4.1, both using an inpainted frame and a background-only frame as the condition. In section 4.2 and section 4.3, we compare with baselines through qualitative and quantitative evaluations, including a user study. In section 4.4, we provide ablation study results on key design choices. In the appendix, we include additional results on general video inpainting / outpainting, generalization to videos from diverse domains, and robustness to inaccurate segmentation, along with further implementation details and a discussion on limitations and potential ethical impacts.

We strongly encourage the reader to check our webpage, where we show extensive videos on video background generation with diverse generated contents and camera motions, and under various condition scenarios. It also contains the video version of the comparison with baselines.

Evaluation data. Following prior works [5, 10, 13, 16, 42], we compare with previous works on videos sampled from the DAVIS [30] dataset. We select videos with both human and non-human subjects. We also evaluate qualitatively and perform ablation study on held-out samples from the HiC+ dataset following the original data splits [26]. Samples with our method are generated with 50 denoising steps, with a guidance scale [34] of 5.

4.1 Diverse Generation with ActAnywhere

In fig. 3, we show results on the held-out segmentation sequences from the HiC+ dataset, using an inpainted frame or a background-only frame as condition. ActAnywhere generates highly realistic foreground-background interactions both at coarse and fine levels. At a coarse level, our model synthesizes road structure, pumpkin field, city views, waves, etc. that align with the subject’s motion. While at a fine level, our method also generates small moving objects that are in close interaction with the subject, such as the buckets, bed sheets, horses and dune buggies, as well as the dog. Moreover, these generation stay consistent across frames, and tightly follow the guidance in the condition frame. The synthesized backgrounds also exhibit coherent scale, lighting, and shadows (also see fig. 1).

[†]<https://huggingface.co/runwayml/stable-diffusion-inpainting>

[‡]<https://github.com/guoyww/animatediff/>

[§]<https://huggingface.co/openai/clip-vit-large-patch14>

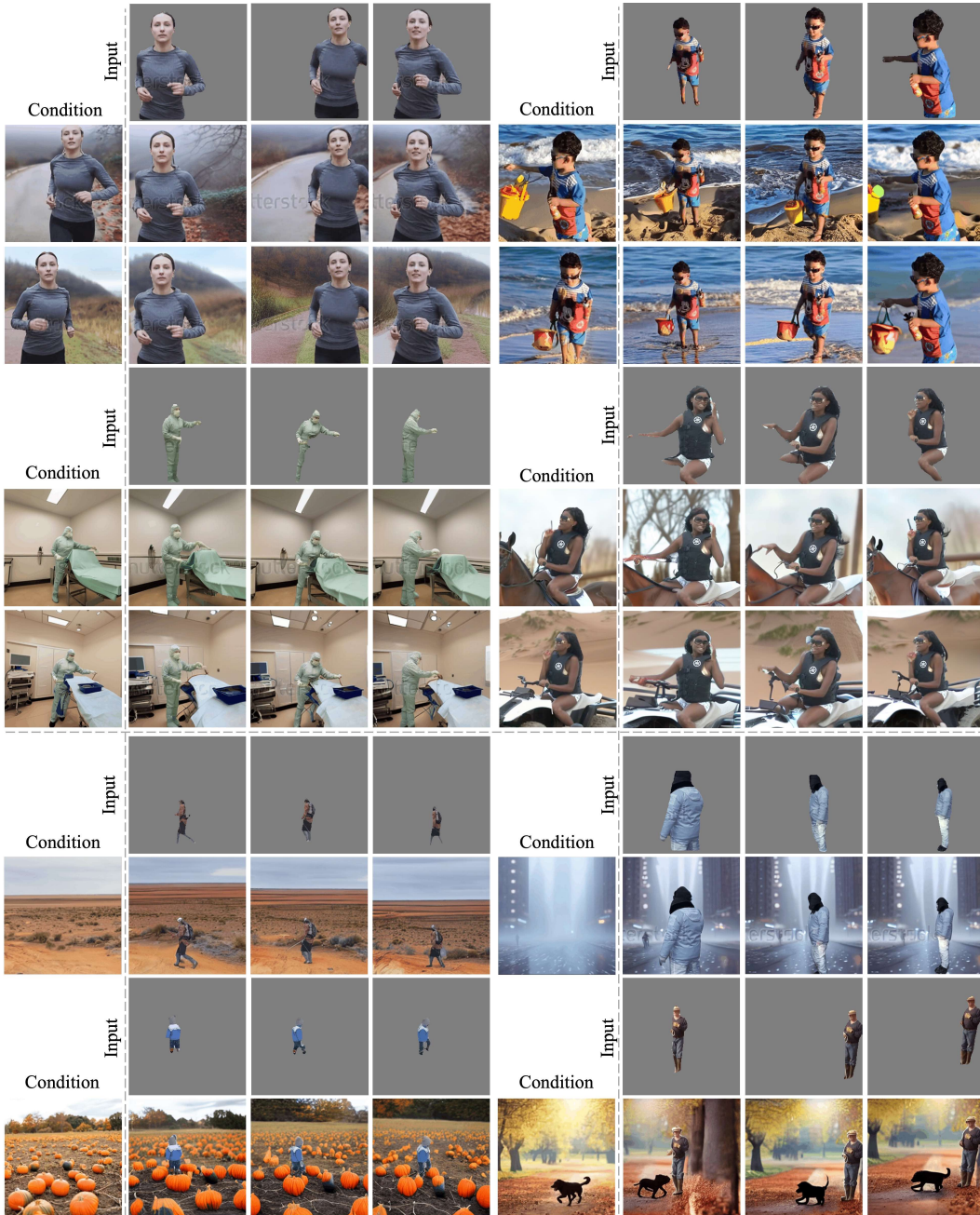


Figure 3: **Diverse results from our method.** The top part shows examples using inpainted frames as condition, while bottom contains examples with background-only conditioning. Foreground sequences are from the held-out set of HiC+.



Figure 4: **Comparison with baselines.** We provide results on two videos sampled from the DAVIS [30] dataset. For each example, we show three representative frames (top) and their corresponding condition signal (left). Note that different methods assume different input, conditioning or pre-trained models, as specified in section 4.2.

4.2 Qualitative Comparison

Baselines. We first clarify that since we study a *novel problem*, and there is no prior work operating under the exact same setting to the best of our knowledge. We hence compare to closest works and adapt some, *i.e.* AnimateDiff [17], if necessary. Nonetheless, we emphasize that the formulation and pipeline are the core contribution of this work.

We compare ActAnywhere to a number of baselines, which we classify based on whether they do (fig. 4 middle) or do not (fig. 4 bottom) take a video as input. For the methods taking a video as input, Gen1 [13] takes an additional image as condition, and also leverages a pre-trained depth-estimation network [33]. Given pre-trained neural atlases [23], Text2LIVE [5] assumes a text prompt as condition to synthesize the edited video. TokenFlow [16] also uses text conditioning. Control-A-Video [10] first extracts Canny edges from the input video, then synthesizes the output video conditioned jointly on the edges and text.

Method	CLIP_cond \uparrow	CLIP_temp \uparrow	FVD \downarrow	Condition	Visual	Temporal
Control-A-Video [7]	0.643	0.942	381.8	97.47%	72.13%	64.36%
TokenFlow [12]	0.762	0.943	323.0			
Gen1 [9]	0.827	0.943	337.2			
Ours	0.862	0.945	313.4			

Table 1: **Quantitative comparison** against baselines.

Table 2: **User study**

For baselines that do not take a video as input, the original AnimateDiff [17] only uses text conditioning. We use the strategy contributed by a public pull request[‡] to make it take additional image conditioning. Specifically, at test time, latent features are first extracted from the condition image with the pre-trained SD VAE encoder [34], which are then merged with the original per-frame Gaussian noises through linear blending. The diffusion process is later conditioned on a text prompt too. VideoCrafter1 [9] provides both a text-to-video and an image-to-video model. We use the latter for a closer comparison setting.

Results. The qualitative comparison on two examples from the DAVIS [30] dataset is shown in fig. 4. Our method generates temporally coherent videos that follow the foreground motion with highly realistic details, *e.g.* falling snow and snow on the car windshield, while strictly following the guidance and constraints given by the condition frame. Baseline methods in the first category generally inherit the structure present in the input video, *e.g.* road direction, horse, etc., and hence they completely fail when fine-grained edits are desired, *e.g.* horse changes to motorcycle in the second case. Methods in the second category generate unconstrained motion due to lack of guidance (VideoCrafter1 in the second example generates backward motion, which is more evident in the video on our webpage).

4.3 Quantitative Comparison

Baselines. We compare with Control-A-Video [10], TokenFlow [16] and Gen1 [13]. We also conduct a user study for preferences in comparison to Gen1.

Metrics. We evaluate the consistency to the condition image, the temporal consistency of the generated videos, as well as the general generation quality. Specifically, we report the average cosine similarity between the CLIP [31] image embeddings of all generated frames and that of the condition image (*CLIP_cond*), between all pairs of generated frames (*CLIP_temp*), and the FVD score [39] against a set of real videos.

For the user study, we ask the participants if they prefer results from our model vs. those from Gen1 on: 1) consistency to condition image; 2) visual quality; and 3) temporal consistency. The results are presented as the percentage of our model being preferred over Gen1.

Results. Results on 30 videos from DAVIS are reported in table 1. Our model outperforms baselines across all metrics, particularly by a big margin on *CLIP_cond*, suggesting that our model is able to generate videos that tightly follow the guidance of the condition frame. For the user study, we randomly select 20 videos and ask 16 participants for their preference over Gen1 [13]. Results are shown in table 2. Our method is strongly preferred on these key aspects.

4.4 Ablation Study

We study different choices of conditioning by experimenting with three variants. *VAE Concat* uses the same VAE encoder \mathcal{E} to extract features for the condition frame c , and concatenates with the input F_τ^i along the feature dimension. *VAE Cross-Attn* leverages the VAE features $\mathcal{E}(c)$ through cross-attention instead of concatenation. *No Mask* does not concatenate the masks \mathcal{M} with the foreground segmentation S and noise Z_τ for the input.

The results on the held-out set of HiC+ are shown in table 3. Providing the condition in the CLIP feature space through cross-attention provides global semantic guidance to the entire diffusion process, while VAE features pose stricter spatial constraints, especially through concatenation, hence making it harder to produce coherent videos that conform to the condition. Providing the mask as

[‡]<https://github.com/guoyww/AnimateDiff/pull/8>

Variants	CLIP_cond ↑	CLIP_temp ↑	FVD ↓
VAE Concat	0.743	0.939	335.2
VAE Cross-Attn	0.823	0.942	326.6
No Mask	0.850	0.943	321.7
Full Model (Ours)	0.857	0.944	315.2

Table 3: **Ablation study** on key design choices.

additional input further indicates the region that the model should generate thus also improves the overall generation performance.

5 Conclusion

We present ActAnywhere, a video diffusion-based model that generates videos with coherent and vivid foreground-background interactions, given an input foreground segmentation sequence and a condition frame describing the background. Our model synthesizes highly realistic details such as moving or interacting objects and shadows. The generated videos also exhibit consistent camera scales and lighting effects. We believe our work contributes a useful tool for the movie and visual effects community, as well as for the general public to realize novel ideas of situating an acting subject in diverse scenes, in a simple and efficient way that is not previously possible.

6 Acknowledgment

We thank the authors of [26] for compiling and processing the dataset HiC+, especially Sumith Kulal for the code and instructions on accessing the data. We also thank Jiahui (Gabriel) Huang from Adobe Research for helping set up the Adobe Firefly GenFill API. Boxiao Pan and Leonidas J. Guibas are supported by a grant from the Stanford Human-Centered AI Institute (HAI) and a Vannevar Bush Faculty Fellowship.

References

- [1] Virtual Production. https://en.wikipedia.org/wiki/On-set_virtual_production
- [2] Adobe: Firefly (2023), <https://www.adobe.com/sensei/generative-ai/firefly.html>
- [3] Adobe: Photoshop, version (2023), <https://www.adobe.com/products/photoshop.html>
- [4] Bain, M., Nagrani, A., Varol, G., Zisserman, A.: Frozen in time: A joint video and image encoder for end-to-end retrieval. In: ICCV (2021)
- [5] Bar-Tal, O., Ofri-Amar, D., Fridman, R., Kasten, Y., Dekel, T.: Text2live: Text-driven layered image and video editing. In: ECCV (2022)
- [6] Blattmann, A., Rombach, R., Ling, H., Dockhorn, T., Kim, S.W., Fidler, S., Kreis, K.: Align your latents: High-resolution video synthesis with latent diffusion models. In: CVPR (2023)
- [7] Brooks, T., Efros, A.A.: Hallucinating pose-compatible scenes. In: ECCV (2022)
- [8] Ceylan, D., Huang, C.H.P., Mitra, N.J.: Pix2video: Video editing using image diffusion. In: ICCV (2023)
- [9] Chen, H., Xia, M., He, Y., Zhang, Y., Cun, X., Yang, S., Xing, J., Liu, Y., Chen, Q., Wang, X., et al.: Videocrafter1: Open diffusion models for high-quality video generation. arXiv preprint arXiv:2310.19512 (2023)
- [10] Chen, W., Wu, J., Xie, P., Wu, H., Li, J., Xia, X., Xiao, X., Lin, L.: Control-a-video: Controllable text-to-video generation with diffusion models. arXiv preprint arXiv:2305.13840 (2023)
- [11] Clark, A., Donahue, J., Simonyan, K.: Adversarial video generation on complex datasets. arXiv preprint arXiv:1907.06571 (2019)
- [12] Diba, A., Fayyaz, M., Sharma, V., Paluri, M., Gall, J., Stiefelhagen, R., Van Gool, L.: Large scale holistic video understanding. In: ECCV (2020)
- [13] Esser, P., Chiu, J., Atighehchian, P., Granskog, J., Germanidis, A.: Structure and content-guided video synthesis with diffusion models. In: ICCV (2023)
- [14] Fan, F., Guo, C., Gong, L., Wang, B., Ge, T., Jiang, Y., Luo, C., Zhan, J.: Hierarchical masked 3d diffusion model for video outpainting. In: ACM MM (2023)
- [15] Ge, S., Hayes, T., Yang, H., Yin, X., Pang, G., Jacobs, D., Huang, J.B., Parikh, D.: Long video generation with time-agnostic vqgan and time-sensitive transformer. In: ECCV (2022)
- [16] Geyer, M., Bar-Tal, O., Bagon, S., Dekel, T.: Tokenflow: Consistent diffusion features for consistent video editing. arXiv preprint arXiv:2307.10373 (2023)
- [17] Guo, Y., Yang, C., Rao, A., Wang, Y., Qiao, Y., Lin, D., Dai, B.: Animatediff: Animate your personalized text-to-image diffusion models without specific tuning. arXiv preprint arXiv:2307.04725 (2023)
- [18] He, K., Gkioxari, G., Dollár, P., Girshick, R.: Mask r-cnn. In: ICCV (2017)
- [19] He, Y., Yang, T., Zhang, Y., Shan, Y., Chen, Q.: Latent video diffusion models for high-fidelity long video generation. arXiv preprint arXiv:2211.13221 (2023)
- [20] Ho, J., Chan, W., Saharia, C., Whang, J., Gao, R., Gritsenko, A., Kingma, D.P., Poole, B., Norouzi, M., Fleet, D.J., et al.: Imagen video: High definition video generation with diffusion models. arXiv preprint arXiv:2210.02303 (2022)
- [21] Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. In: NeurIPS (2020)

- [22] Hu, Z., Xu, D.: Videocontrolnet: A motion-guided video-to-video translation framework by using diffusion model with controlnet. arXiv preprint arXiv:2307.14073 (2023)
- [23] Kasten, Y., Ofri, D., Wang, O., Dekel, T.: Layered neural atlases for consistent video editing. ACM TOG **40**(6) (2021)
- [24] Khachatryan, L., Movsisyan, A., Tadevosyan, V., Henschel, R., Wang, Z., Navasardyan, S., Shi, H.: Text2video-zero: Text-to-image diffusion models are zero-shot video generators. arXiv preprint arXiv:2303.13439 (2023)
- [25] Kingma, D.P., Welling, M.: Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114 (2013)
- [26] Kulal, S., Brooks, T., Aiken, A., Wu, J., Yang, J., Lu, J., Efros, A.A., Singh, K.K.: Putting people in their place: Affordance-aware human insertion into scenes. In: CVPR (2023)
- [27] Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. arXiv preprint arXiv:1711.05101 (2017)
- [28] Monfort, M., Andonian, A., Zhou, B., Ramakrishnan, K., Bargal, S.A., Yan, T., Brown, L., Fan, Q., Gutfreund, D., Vondrick, C., et al.: Moments in time dataset: one million videos for event understanding. IEEE TPAMI **42**(2) (2019)
- [29] OpenAI: ChatGPT (2023), <https://chat.openai.com/>
- [30] Pont-Tuset, J., Perazzi, F., Caelles, S., Arbeláez, P., Sorkine-Hornung, A., Van Gool, L.: The 2017 davis challenge on video object segmentation. arXiv preprint arXiv:1704.00675 (2017)
- [31] Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: ICML (2021)
- [32] Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., Chen, M.: Hierarchical text-conditional image generation with clip latents. arXiv preprint arXiv:2204.06125 (2022)
- [33] Ranftl, R., Lasinger, K., Hafner, D., Schindler, K., Koltun, V.: Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. IEEE TPAMI **44**(3) (2020)
- [34] Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: CVPR (2022)
- [35] Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: MICCAI (2015)
- [36] Saharia, C., Chan, W., Chang, H., Lee, C., Ho, J., Salimans, T., Fleet, D., Norouzi, M.: Palette: Image-to-image diffusion models. In: ACM SIGGRAPH (2022)
- [37] Sigurdsson, G.A., Varol, G., Wang, X., Farhadi, A., Laptev, I., Gupta, A.: Hollywood in homes: Crowdsourcing data collection for activity understanding. In: ECCV (2016)
- [38] Tulyakov, S., Liu, M.Y., Yang, X., Kautz, J.: Mocogan: Decomposing motion and content for video generation. In: CVPR (2018)
- [39] Unterthiner, T., van Steenkiste, S., Kurach, K., Marinier, R., Michalski, M., Gelly, S.: Fvd: A new metric for video generation (2019)
- [40] Villegas, R., Babaeizadeh, M., Kindermans, P.J., Moraldo, H., Zhang, H., Saffar, M.T., Castro, S., Kunze, J., Erhan, D.: Phenaki: Variable length video generation from open domain textual description. arXiv preprint arXiv:2210.02399 (2022)
- [41] Vondrick, C., Pirsaviash, H., Torralba, A.: Generating videos with scene dynamics. In: NeurIPS (2016)
- [42] Wu, J.Z., Ge, Y., Wang, X., Lei, S.W., Gu, Y., Shi, Y., Hsu, W., Shan, Y., Qie, X., Shou, M.Z.: Tune-a-video: One-shot tuning of image diffusion models for text-to-video generation. In: ICCV (2023)
- [43] Wu, J., Li, X., Si, C., Zhou, S., Yang, J., Zhang, J., Li, L., Chen, K., Tong, Y., Liu, Z., Loy, C.C.: Towards language-driven video inpainting via multimodal large language models. arXiv pre-print (2023)
- [44] Xie, S., Zhang, Z., Lin, Z., Hinz, T., Zhang, K.: Smartbrush: Text and shape guided object inpainting with diffusion model. In: CVPR (2023)
- [45] Yang, B., Gu, S., Zhang, B., Zhang, T., Chen, X., Sun, X., Chen, D., Wen, F.: Paint by example: Exemplar-based image editing with diffusion models. In: CVPR (2023)

- [46] Yu, L., Cheng, Y., Sohn, K., Lezama, J., Zhang, H., Chang, H., Hauptmann, A.G., Yang, M.H., Hao, Y., Essa, I., et al.: Magvit: Masked generative video transformer. In: CVPR (2023)
- [47] Yu, S., Sohn, K., Kim, S., Shin, J.: Video probabilistic diffusion models in projected latent space. In: CVPR (2023)
- [48] Zhang, L., Rao, A., Agrawala, M.: Adding conditional control to text-to-image diffusion models. In: CVPR (2023)
- [49] Zhou, S., Li, C., Chan, K.C., Loy, C.C.: Propainter: Improving propagation and transformer for video inpainting. In: ICCV (2023)



Figure 5: **Zero-shot video inpainting.** We show two cases from DAVIS, each with four sampled frames. The yellow regions denote the masked areas to be inpainted.

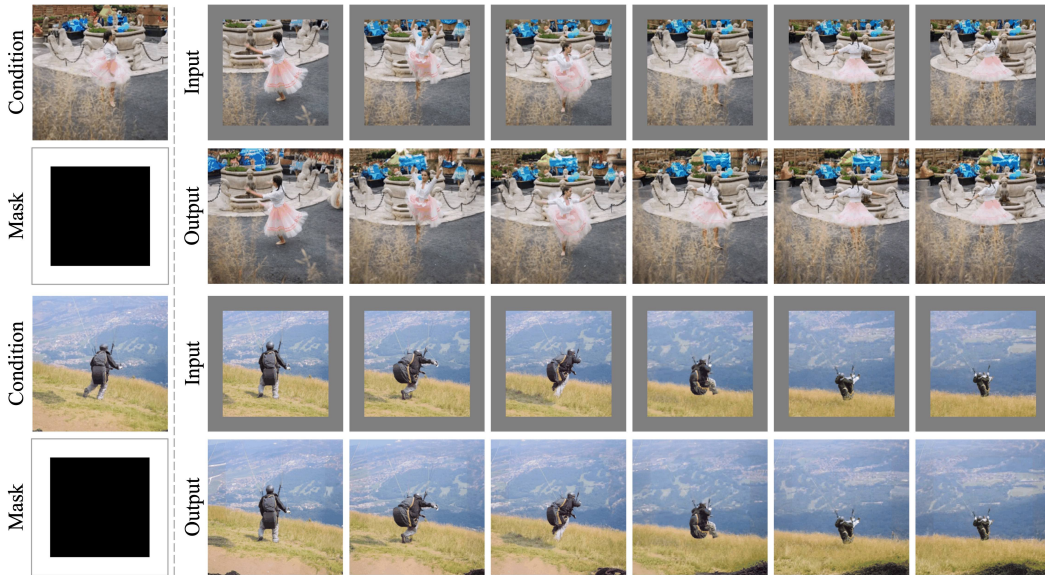


Figure 6: **Zero-shot video outpainting.** Both examples are from DAVIS.

7 Appendix

In this appendix, we first provide additional results and analysis in section 7.1, including inpainting / outpainting capability emerging from our model, generalization to domains different from the training data, robustness to inaccurate foreground segmentation, and inference runtime. We then describe essential processing steps for training and evaluation data in section 7.2. Next, we show failure cases and discuss limitations of our model in section 7.3. Lastly, we conclude by discussing the ethical impact of this work in section 7.4.

7.1 Additional Results and Analysis

General video inpainting / outpainting. Interestingly, once trained, certain general video inpainting / outpainting capability emerges from our model. We perform preliminary experiments by manually creating a mask sequence, and pass those with the foreground sequence as the input to our model. Two cases are shown in fig. 5, where our model is able to inpaint the missing regions, despite not explicitly trained so. Similarly, our model can also be applied to general video outpainting, whose results are shown in fig. 6. Specifically, we resize the original sequence of frames by 0.75, and pad them with gray boundaries. Associated masks are also created to indicate the missing regions. We then randomly select one frame and use Adobe Firefly [2] to outpaint it, with which as condition we outpaint the entire sequence.

These results may suggest that our model learns to approximate the underlying data distribution to a certain degree, possibly benefiting from the random condition dropping during training (section 3.4).

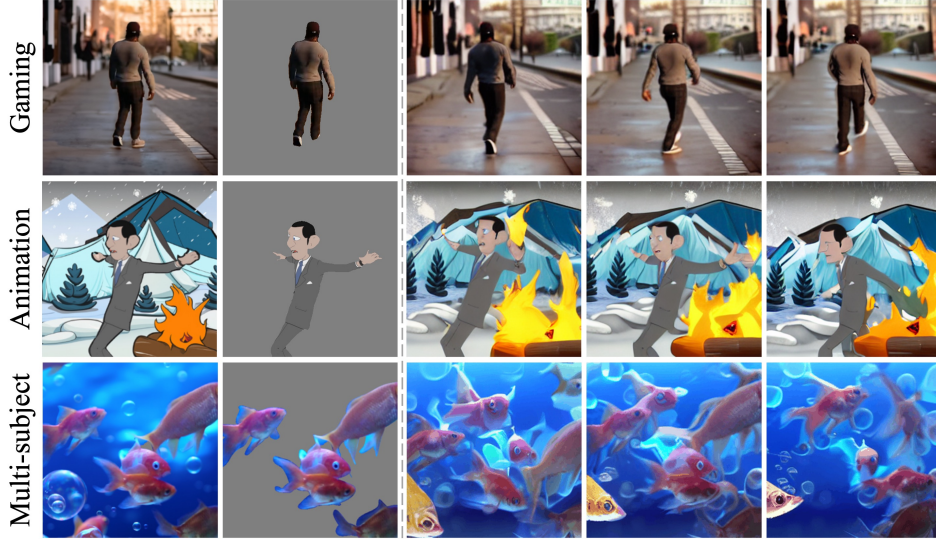


Figure 7: **Videos from various domains.** Gaming video from GTA.

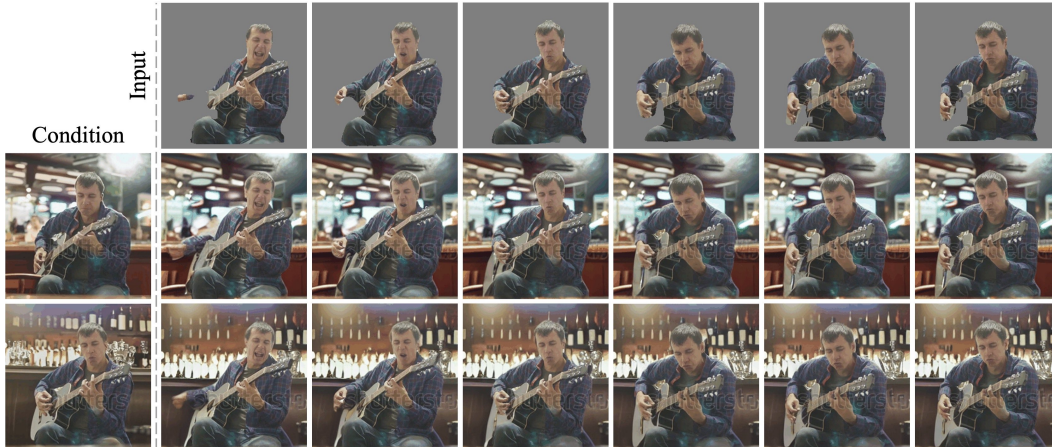


Figure 8: Our model is robust to inaccurate masks. We show one video sequence from HiC+ with two different condition frames, followed by six generated frames for each.

Videos from various domains. Our model generalizes to various domains such as gaming and animation clips, as well as videos with multiple moving subjects. In fig. 7, we show the condition frame with its segmentation, along with three generated frames for one example in each domain.

Robust to inaccurate masks. As stated in section 3.4, masks created or extracted in practice are often imperfect, being either incomplete or excessive. Here we show that our model trained in our designed procedure is robust to imperfect masks. In fig. 8 we showcase an example of this. Despite a large region of the guitar missing, our model is able to hallucinate them in a reasonable way by taking the global context into account.

Runtime. Generating one video on an NVIDIA A100 GPU takes about 8.5 seconds, enabling much faster idea iteration compared to traditional workflows.

7.2 Data Processing

Training. In Sec. 3.4 of the main manuscript, we described our data processing and augmentation strategies for training data. Specifically, to deal with incomplete segmentation, we apply random rectangular cut-outs to the segmentation and masks. We show two examples in fig. 9.

Evaluation. As mentioned in Sec. 1 of the main manuscript, at test time, the composited foreground-background frames used as condition can be created with various methods, such as photo editing

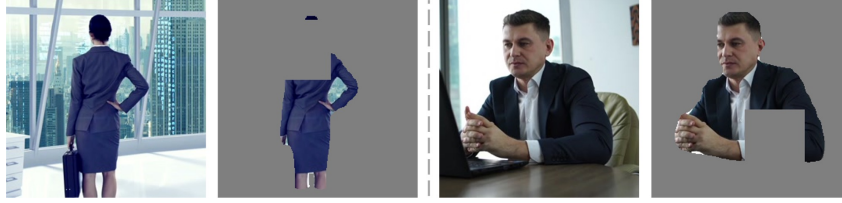


Figure 9: **Data augmentation.** We apply random cut-outs to the person segmentation during training. Here we show two examples of cut-outs with their corresponding original frames.



Figure 10: **Failure cases.** Foreground sequences from DAVIS.

tools (*e.g.* Adobe Photoshop [3]) or automated image outpainting methods (*e.g.* Dall-E [32]). In our experiments, we adopt ChatGPT 4 [29] and Adobe Firefly [2] to automatically synthesize these composited frames for use at test time. Specifically, we first sample the 0-th, 8-th, and 15-th frames from the input video, and ask ChatGPT the following question “Based on these three images, can you give me a description as prompt, less than 10 words, about one contextual scenario we can put this human in?”. We then use Firefly to synthesize the outpainted frames, given the foreground segmentation and the text prompt. We use the “gpt4-vision-preview” version of ChatGPT 4[†], and the official Firefly GenFill^{**}.

7.3 Limitations

We show two failure cases of our method in fig. 10. In the first example, the grass-like texture on the dress is excluded from the segmentation, hence the model mistakenly perceives it to be an independent object growing outside the dress. While in the second example, the Firefly-inpainted frame has the broomstick facing towards the wrong direction. Although the model tries to generate something reasonable, it fails to correct this mistake to produce a coherent video. Despite certain fault tolerance from our model, providing proper input segmentation and condition signal helps ensure high-quality generation results.

7.4 Data Ethics

The HiC+ dataset [26] includes public datasets HVU [12], Charades [37], Moments [28], and WebVid10M [4]. Readers are encouraged to refer to Section A.1 in the supplementary material of [7] for the license of these datasets. Moreover, same as [26], more internal videos are incorporated during training and evaluation. We have conducted the necessary legal review process, and can provide more details of this data upon request.

We present a method that can synthesize highly realistic videos that place human and non-human subjects into diverse background scenes. While our work has implications and values for a wide range of fields, we note that our model can be misused to generate malicious content. Similar to Firefly GenFill [2] and Gen1 [13], the generated content can be watermarked to prevent misuse. Moreover, our model inherits the demographic biases presented in the training data. We make our best effort to demonstrate impartiality in all the results shown.

[†]<https://platform.openai.com/docs/overview>
^{**}<https://firefly.adobe.com/generate/inpaint>

NeurIPS Paper Checklist

1. Claims

Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

Answer: [Yes]

Justification: The contributions and scope of the paper have been truthfully described throughout the abstract and the introduction, and concisely summarized at the end of the introduction. Each of the listed contributions is backed up by experimental results or literature review throughout the paper.

Guidelines:

- The answer NA means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. Limitations

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [Yes]

Justification: We included a separate section 7.3 to discuss the limitations of the work, including discussions on potential data bias in section 7.4.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. Theory Assumptions and Proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [NA]

Justification: The paper does not include theoretical results, while essential theoretical background with formulas is provided in section 3.1.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. Experimental Result Reproducibility

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: We included detailed descriptions of our model architecture in section 3.3, references to publicly available pre-trained weights, data processing procedures, and other training details in section 3.4 and further in section 7.2. We will also open-source our code and model checkpoints upon paper acceptance.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
 - (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
 - (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
 - (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

- (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. Open access to data and code

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: As stated in the response above, we provided detailed instructions on how to replicate our experiment results in the paper. We will release our code and models upon paper acceptance.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. Experimental Setting/Details

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: We provided detailed instructions on replicating the training and evaluation procedures in section 3.4, section 4, and section 7.2. We did not perform delicate tuning for the hyperparameters.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. Experiment Statistical Significance

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [No]

Justification: We did not compute error bars due to the lack of adequate computational resources. However, we tried our best to provide faithful and reliable quantitative evaluation results, including a user study.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. Experiments Compute Resources

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: We described our compute setup as well as the training time and inference runtime in section 3.4 and section 7.1, respectively.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. Code Of Ethics

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines?>

Answer: [Yes]

Justification: We included a separate section 7.4 to discuss the ethics-related concerns, including licensing and legal information of the dataset used, and potential societal impacts of our work with possible measures to combat them.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.

- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. Broader Impacts

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [\[Yes\]](#)

Justification: We included a separate section 7.4 that discussed potential societal impacts of our work with possible measures to combat them.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. Safeguards

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [\[Yes\]](#)

We mentioned in section 7.4 that the generated content from our model can be watermarked to prevent potential misuse.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. Licenses for existing assets

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: As noted in section 7.4, the licenses for the datasets used in this work can be found in the supplement of [7]. All the pre-trained models and tools we used are specified in the paper in section 3.4 and section 7.2 with URLs.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, paperswithcode.com/datasets has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. New Assets

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [NA]

Justification:

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. Crowdsourcing and Research with Human Subjects

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification:

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification:

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.