Structure and Content-Guided Video Synthesis with Diffusion Models

Patrick Esser Johnathan Chiu Parmida Atighehchian
Jonathan Granskog Anastasis Germanidis
Runway
https://research.runwayml.com/gen1

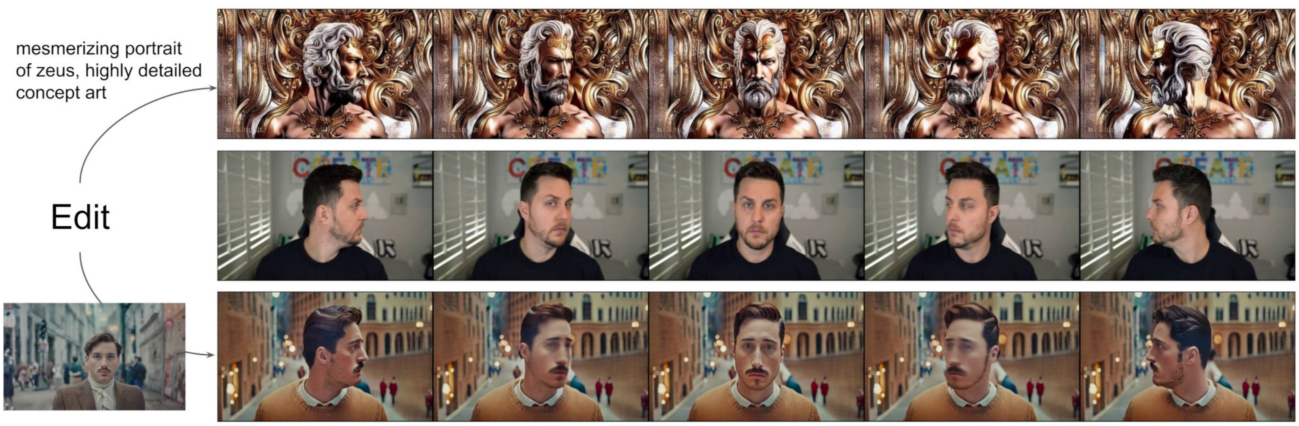


Figure 1. **Guided Video Synthesis** We present an approach based on latent video diffusion models that synthesizes videos (top and bottom) guided by content described through text (top) or images (bottom) while keeping the structure of an input video (middle).

Abstract

Text-guided generative diffusion models unlock powerful image creation and editing tools. Recent approaches that edit the content of footage while retaining structure require expensive re-training for every input or rely on error-prone propagation of image edits across frames.

In this work, we present a structure and content-guided video diffusion model that edits videos based on descriptions of the desired output. Conflicts between user-provided content edits and structure representations occur due to insufficient disentanglement between the two aspects. As a solution, we show that training on monocular depth estimates with varying levels of detail provides control over structure and content fidelity. A novel guidance method, enabled by joint video and image training, exposes explicit control over temporal consistency. Our experiments demonstrate a wide variety of successes; fine-grained control over output characteristics, customization based on a few reference images, and a strong user preference towards results by our model.

1. Introduction

Demand for more intuitive and performant video editing tools has increased as video-centric platforms have been popularized. But editing in the format is still complex and time-consuming due the temporal nature of video data. State-of-the-art machine learning models have shown great promise in improving editing workflows.

Generative approaches for image synthesis recently experienced a rapid surge in quality and popularity due to the introduction of powerful diffusion models trained on large-scale datasets. Text-conditioned models, such as DALL-E 2 [34] and Stable Diffusion [38], enable novice users to generate detailed imagery given only a text prompt as input. Latent diffusion models especially enable efficient methods for producing imagery via synthesis in a perceptually compressed space.

Motivated by this progress, we investigate generative models suited for interactive applications in video editing.

Current methods repurpose existing image models by either propagating edits with approaches that compute explicit correspondences [5] or by finetuning on each individual video [63]. We aim to circumvent expensive per-video training and correspondence calculation to achieve fast inference for arbitrary videos.

We propose a controllable structure and content-aware latent video diffusion model trained on a large-scale dataset of uncaptioned videos and images. We opt to represent structure with monocular depth estimates, and content with embeddings predicted by a pre-trained neural network. Our approach offers several powerful modes of control. First, we train our model such that the content of inferred videos, *e.g.* their appearance or style, match user-provided images or text prompts (Fig. 1). Second, we vary the fidelity of the structure representation during training to allow selecting the strength of the structure preservation at test-time. Finally, we also adjust the inference process via a custom guidance method, inspired by classifier-free guidance, to enable control over temporal consistency.

In summary, we present the following contributions:

- We extend latent diffusion models to video generation by introducing temporal layers into a pre-trained image model and by joint training on images and videos.
- We present a structure and content-aware model that edits videos given example images or text. Our method does not require per-video training or pre-processing.
- We demonstrate full control over temporal, content and structure consistency. We show for the first time that joint image-video training enables control over temporal stability. And, training on varying levels of detail in the structure representation allows choosing the desired level of preservation during inference.
- We show that our approach is preferred over several other approaches in a user study. We further improve the accuracy of previously unseen content by finetuning on a small set of images of the desired subject.

2. Related Work

Controllable video editing and media synthesis is an active area of research. In this section, we review prior work in related areas and connect our method to these approaches.

Unconditional video generation Generative adversarial networks (GANs) [12] can learn to synthesize videos based on specific training data [59, 45, 1, 56]. These methods often struggle with stability during optimization, and produce fixed-length videos [59, 45] or longer videos where artifacts accumulate over time [50]. [6] synthesize longer videos using a GAN with a better encoding of the time axis. Autoregressive transformers have also been proposed for unconditional video generation [11, 64]. Our focus is on providing user control over the synthesis process whereas these approaches are limited to sampling random content resembling their training distribution.

Diffusion models for image synthesis Diffusion models (DMs) [51, 53] can synthesize detailed media in many formats, such as images [34, 38], 3d shapes [66] and animations [54]. Many works improve diffusion-based image synthesis by changing the parameterization [14, 27, 46], introducing advanced sampling methods [52, 24, 22, 47, 20], designing more powerful architectures [3, 15, 57, 30], or conditioning on additional information [25]. Text-conditioning, based on embeddings from CLIP [32] or T5 [33], has become a particularly powerful approach for providing artistic control over model output [44, 28, 34, 3, 65, 10]. Latent diffusion models (LDMs) [38] perform diffusion in a compressed latent space reducing memory requirements and runtime. Our video model is an LDM trained simultaneously on videos and images.

Diffusion models for video synthesis Recently, diffusion models, masked generative models and autoregressive models have been applied to text-conditioned video synthesis [17, 13, 58, 67, 18, 49]. Similar to [17] and [49], we extend image synthesis diffusion models to video generation by introducing temporal connections into a pre-existing image model. Our model edits videos rather than synthesizing them from scratch. We demonstrate through a user study that our model with explicit conditioning over structure is preferred over other related approaches.

Video translation and propagation Image-to-image translation models, such as pix2pix [19, 62], can modify each frame in a video individually. This produces temporal inconsistencies as the time axis is ignored. Accounting for temporal or geometric information, such as flow, can increase consistency across frames when repurposing image synthesis models [42, 9]. We can extract such structural information to aid our spatio-temporal LDM in text- and image-guided video synthesis. Many generative adversarial methods, such as vid2vid [61, 60], leverage this type of input to guide synthesis.

Video style transfer takes a reference style image and statistically applies its style to an input video [40, 8, 55]. In contrast, our method edits both style and content while preserving the structure of a video instead of matching feature statistics only. Text2Live [5] allows editing input videos using text prompts by decomposing a video into neural layers [21]. Once available, a layered video representation [37] provides consistent propagation across frames. SinFusion [29] and Tune-a-Video [63] use diffusion models to edit videos but require per-video training. This limits the practicality of the approaches in creative tools. We opt to instead train our model on a large-scale dataset permitting inference on any video without individual training.

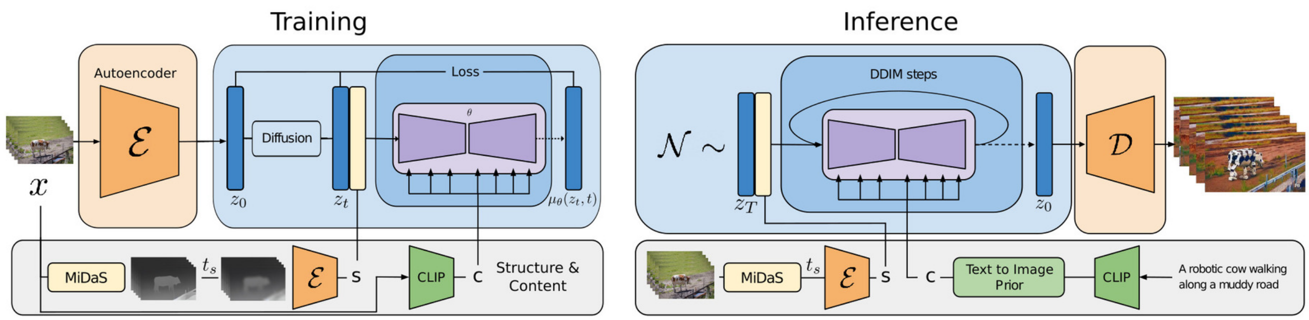


Figure 2. **Overview:** During training (left), input videos x are encoded to z_0 with a fixed encoder $\mathcal{E}$ and diffused to z_t. We extract a structure representation s by encoding depth maps obtained with MiDaS, and a content representation c by encoding one of the frames with CLIP. The model then learns to reverse the diffusion process in the latent space, with the help of s, which gets concatenated to z_t, as well as c, which is provided via cross-attention blocks. During inference (right), the structure s of an input video is provided in the same manner. To specify content via text, we convert CLIP text embeddings to image embeddings via a prior.

3. Method

For our purposes, it will be helpful to think of a video in terms of its *content* and *structure*. By structure, we refer to characteristics describing geometry and dynamics, *e.g.* shapes and locations of subjects as well as their temporal changes. We define content as features describing the appearance and semantics of the video, such as the colors and styles of objects and the lighting. The goal of our model is to edit the content of a video while retaining its structure.

To achieve this, we learn a generative model $p(x|s, c)$ of videos x conditioned on representations of structure s and content c. We infer the shape representation s from an input video, and modify it based on a text prompt c describing the edit. First, we describe our realization of the generative model as a conditional latent video diffusion model and, then, we describe our choices for shape and content representations. Finally, we discuss the optimization process of our model. See Fig. 2 for an overview.

3.1. Latent diffusion models

Diffusion models Diffusion models [51] learn to reverse a fixed forward diffusion process, which is defined as

$$q(x_t|x_{t-1}) \coloneqq \mathcal{N}(x_t, \sqrt{1-\beta_t}x_{t-1}, \beta_t\mathcal{I}) \ . \tag{1}$$

Normally-distributed noise is slowly added to each sample x_{t-1} to obtain x_t. The forward process models a fixed Markov chain and the noise is dependent on a variance schedule β_t where $t \in \{1, \dots, T\}$, with T being the total number of steps in our diffusion chain, and $x_0 \coloneqq x$.

Learning to Denoise The reverse process is defined according to the following equation with parameters θ

$$p_\theta(x_0) \coloneqq \int p_\theta(x_{0:T})dx_{1:T} \tag{2}$$

$$p_\theta(x_{0:T}) = p(x_T)\prod_{t=1}^{T} p_\theta(x_{t-1}|x_t), \tag{3}$$

$$p_\theta(x_{t-1}|x_t) \coloneqq \mathcal{N}(x_{t-1}, \mu_\theta(x_t,t), \Sigma_\theta(x_t,t)) \ . \tag{4}$$

Using a fixed variance $\Sigma_\theta(x_t, t)$, we are left learning the means of the reverse process $\mu_\theta(x_t, t)$. Training is typically performed via a reweighted variational bound on the maximum likelihood objective, resulting in a loss

$$L \coloneqq \mathbb{E}_{t,q}\lambda_t\|\mu_t(x_t,x_0) - \mu_\theta(x_t,t)\|^2 \ , \tag{5}$$

where $\mu_t(x_t, x_0)$ is the mean of the forward process posterior $q(x_{t-1}|x_t, x_0)$, which is available in closed form [14].

Parameterization The mean $\mu_\theta(x_t, t)$ is then predicted by a UNet architecture [39] that receives the noisy input x_t and the diffusion timestep t as inputs. Other parameterizations and weightings, such as ϵ [14] and v-parameterizations [46], can significantly improve sample quality compared to directly predicting the mean. Similar to [13], we found that v-parameterization improves color consistency thus all our experiments use it (see supp. material for more details).

Latent diffusion Latent diffusion models [38] (LDMs) take the diffusion process into the latent space. This provides improved separation between compressive and generative learning phases of the model. Specifically, LDMs use an autoencoder where an encoder $\mathcal{E}$ maps input data x to a lower dimensional latent code according to $z = \mathcal{E}(x)$ while a decoder $\mathcal{D}$ converts latent codes back to the input space such that perceptually $x \approx \mathcal{D}(\mathcal{E}(x))$.

Our encoder downsamples RGB-images $x \in \mathbb{R}^{3\times H\times W}$ by a factor of eight and outputs four channels, resulting in a latent code $z \in \mathbb{R}^{4\times H/8\times W/8}$. Thus, the diffusion UNet operates on a much smaller representation which significantly

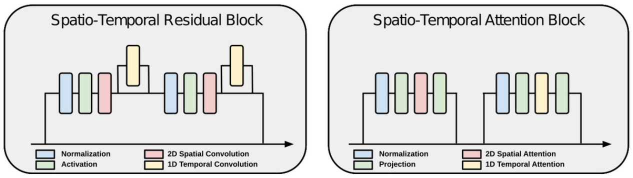


Figure 3. **Temporal Extension:** We extend an image-based UNet architecture to videos, by adding temporal layers in its building blocks. We add a 1D temporal convolution after each 2D spatial convolution in its residual blocks (left), and we add a 1D temporal attention block after each of its 2D spatial attention blocks (right).

improves runtime and memory efficiency. The latter is crucial for video modeling where the additional time axis increases memory costs.

3.2. Spatio-temporal Latent Diffusion

To correctly model a distribution over video frames, the architecture must account for temporal relationships. We also want to jointly learn an image model with shared parameters to benefit from better generalization obtained by training on large-scale image datasets.

To achieve this, we extend an image architecture by introducing temporal layers, which are only active for video inputs. All other layers are shared between the image and video model. The autoencoder remains fixed and processes each frame in a video independently.

The UNet consists of two main building blocks: Residual blocks and transformer blocks (see Fig. 3). Similar to [17, 49], we extend them to videos by adding both 1D convolutions across time and 1D self-attentions across time. In each residual block, we introduce one temporal convolution after each 2D convolution. Similarly, after each spatial 2D transformer block, we also include one temporal 1D transformer block, which mimics its spatial counterpart along the time axis. We also input learnable positional encodings of the frame index into temporal transformer blocks.

3.3. Representing Content and Structure

Conditional Diffusion Models Diffusion models are well-suited to modeling conditional distributions such as $p(x|s, c)$. The forward process q remains unchanged while the conditioning variables s, c become additional inputs to the model.

Our goal is to edit a video based on a text prompt describing the desired output. We choose to train on uncaptioned video data due to the lack of large-scale paired video-text datasets of similar quality as image datasets like [48]. Therefore, during training, we must derive structure and content representations from the training video x itself, *i.e.* $s = s(x)$ and $c = c(x)$, resulting in a per-example loss of

$$\lambda_t \|\mu_t(\mathcal{E}(x)_t, \mathcal{E}(x)_0) - \mu_\theta(\mathcal{E}(x)_t, t, s(x), c(x))\|^2 \ . \quad (6)$$

In contrast, during inference, structure s and content c are derived from an input video y and from a text prompt t respectively. An edited version x of y is obtained by sampling the generative model conditioned on $s(y)$ and $c(t)$:

$$z \sim p_\theta(z|s(y), c(t)), \quad x = \mathcal{D}(z) \ . \quad (7)$$

Content Representation We utilize CLIP [32] to infer a content representation from both text inputs t and video inputs x similar to previous works [35, 3]. CLIP embeddings are a promising content representation as they are more sensitive to semantic and stylistic properties while being more invariant towards geometric attributes [34]. During training, we encode a random frame in each input video with CLIP. To support text-based editing at inference, we train a prior model that allows sampling image embeddings from text embeddings [35, 49].

Structure Representation We need a representation that provides adequate separation between structure and content. We find that depth estimates extracted from input video frames provide the desired properties as they encode significantly less content information compared to simpler structure representations, such as edge filters which also encode textural properties. Still, depth maps reveal the silhouttes of objects which can prevent content edits involving changes in object shape.

To offer control over the amount of structure to preserve, we propose to train a model on structure representations with varying amounts of information. In particular, we blur depth estimates given a parameter t_s. During training, t_s is randomly sampled between 0 and T_s. The parameter can then be controlled at inference to achieve different editing effects (see Fig. 10).

While depths map work well for our usecase, our approach generalizes to other geometric guidance features or combinations of features that might be more helpful for other specific applications. For example, models focusing on human video synthesis might benefit from estimated poses or face landmarks.

Conditioning Mechanisms We account for the different characteristics of our content and structure representations with two different conditioning mechanisms. Since structure represents a significant portion of the spatial information of video frames, we use concatenation for conditioning to make effective use of this information. In contrast, attributes described by the content representation are not tied to particular locations. Hence, we leverage cross-attention which can effectively transport this information to any position.

We use the spatial transformer blocks of the UNet architecture for cross-attention conditioning. Each contains two attention operations, where the first one perform a spatial self-attention and the second one a cross attention with keys and values computed from the CLIP image embedding.

Figure 4. **Temporal Control:** By training image and video models jointly, we obtain explicit control over the temporal consistency of edited videos via a temporal guidance scale ω_t. On the left, frame consistency measured via CLIP cosine similarity of consecutive frames increases monotonically with ω_t, while mean squared error between frames warped with optical flow decreases monotonically. On the right, lower scales (0.5 in the middle row) achieve edits with a "hand-drawn" look, whereas higher scales (1.5 in the bottom row) result in smoother results. Top row shows the original input video, the two edits use the prompt "pencil sketch of a man looking at the camera".

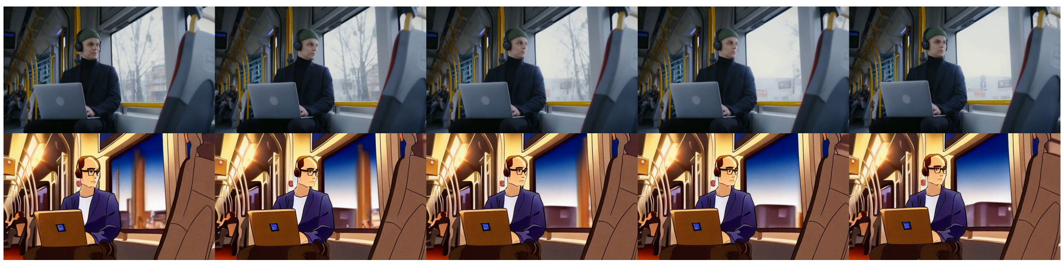
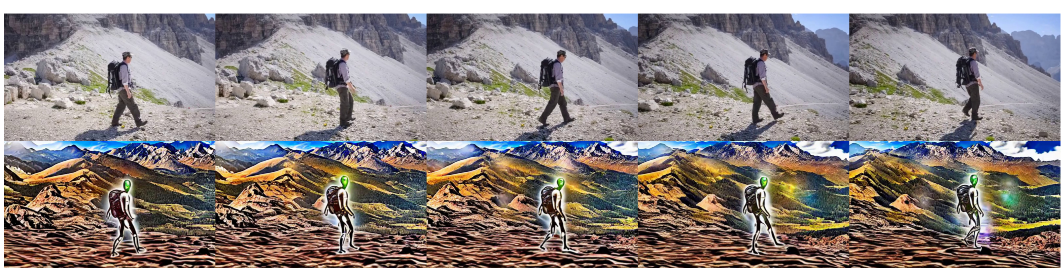
Prompt	Driving Video (top) and Result (bottom)
a man using a laptop inside a train, anime style	
a woman and man take selfies while walking down the street, claymation	
kite-surfer in the ocean at sunset	
alien explorer hiking in the mountains	

Figure 5. Our approach enables a wide range of video edits, including changes to animation styles such as anime or claymation, changes of environment such as time of day, and changing characters such as humans to aliens.

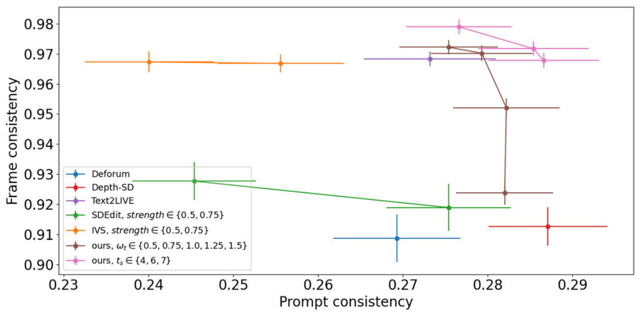


Figure 6. **Prompt-vs-frame consistency:** Image models such as SD-Depth achieve good prompt consistency but fail to produce consistent edits across frames. Propagation based approaches such as IVS and Text2Live increase frame consistency but fail to provide edits reflecting the prompt accurately. Our method achieves the best combination of frame and prompt consistency.

To condition on structure, we first estimate depth maps for all input frames using the MiDaS DPT-Large model [36]. We then apply t_s iterations of blurring and downsampling to the depth maps, where t_s controls the amount of structure to preserve. We resample the perturbed depth map to the resolution of the RGB-frames and encode it using $\mathcal{E}$. This latent representation of structure is concatenated with the input z_t given to the UNet. We also input four channels containing a sinusoidal embedding of t_s.

Sampling While Eq. (2) provides a direct way to sample from the trained model, many other sampling methods [52, 24, 22] require only a fraction of the number of diffusion timesteps to achieve good sample quality. We use DDIM [52] throughout our experiments. Furthermore, classifier-free diffusion guidance [16] significantly improves sample quality. For a conditional model $\mu_\theta(x_t, t, c)$, this is achieved by training the model to also perform unconditional predictions $\mu_\theta(x_t, t, \emptyset)$ and then adjusting predictions during sampling according to

$$\tilde{\mu}_\theta(x_t, t, c) = \mu_\theta(x_t, t, \emptyset) + \omega(\mu_\theta(x_t, t, c) - \mu_\theta(x_t, t, \emptyset))$$

where ω is the guidance scale that controls the strength. Based on the intuition that ω extrapolates the direction between an unconditional and a conditional model, we apply this idea to control temporal consistency of our model. Specifically, since we are training both an image and a video model with shared parameters, we can consider predictions by both models for the same input. Let $\mu_\theta(z_t, t, c, s)$ denote the prediction of our video model, and let $\mu^\pi_\theta(z_t, t, c, s)$ denote the prediction of the image model applied to each frame individually. Taking classifier-free guidance for c into account, we then adjust our prediction according to

$$\begin{aligned}\tilde{\mu}_\theta(z_t, t, c, s) = {} & \mu^\pi_\theta(z_t, t, \emptyset, s) \\ & + \omega_t(\mu_\theta(x_t, t, \emptyset, s) - \mu^\pi_\theta(x_t, t, \emptyset, s)) \\ & + \omega(\mu_\theta(x_t, t, c, s) - \mu_\theta(x_t, t, \emptyset, s))\end{aligned} \tag{8}$$

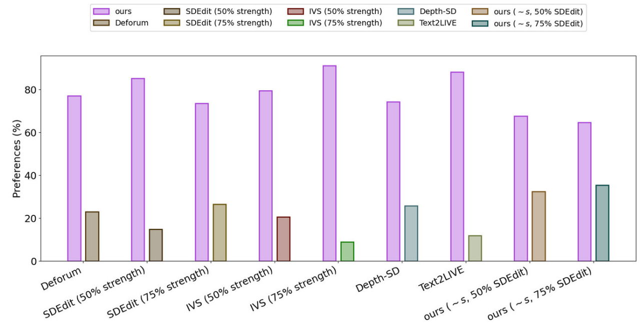


Figure 7. **User Preferences:** Based on our user study, the results from our model are preferred over the baseline models.

Our experiments demonstrate that this approach controls temporal consistency in the outputs, see Fig. 4.

3.4. Optimization

We train on an internal dataset of 240M images and 6.4M video clips. We use image batches of size 9216 with resolutions of 320×320, 384×320 and 448×256, as well as the same resolutions with flipped aspect ratios. We sample image batches with a probabilty of 12.5%. For videos, we use batch size 1152 and 8 frames from each video sampled four frames apart with a resolution of 448×256.

We train our model in multiple stages. First, we initialize model weights based on a pretrained text-conditional latent diffusion model [38][1]. We change the conditioning from CLIP text embeddings to CLIP image embeddings and fine-tune for 15k steps on images only. Afterwards, we introduce temporal connections as described in Sec. 3.2 and train jointly on images and videos for 75k steps. We then add conditioning on structure s with $t_s \equiv 0$ fixed and train for 25k steps. Finally, we resume training with t_s sampled uniformly between 0 and 7 for another 10k steps.

4. Results

To evaluate our approach, we use videos from DAVIS [31] and various stock footage. To automatically create edit prompts, we first run a captioning model [23] to obtain a description of the original video content. We then use GPT-3 [7] to generate edited prompts.

4.1. Qualitative Results

We demonstrate that our approach performs well on a number of diverse inputs (see Fig. 5). Our method handles a large variety of footage, such as landscapes and close-ups, and diverse camera motion without any explicit tracking of the input. Our depth-based structure representation combined with large-scale image-video joint training enable strong generalization and powerful editing capabili-

[1] https://github.com/runwayml/stable-diffusion

Figure 8. **Background Editing:** Masking the denoising process allows us to restrict edits to backgrounds for more control over results.

ties. For example, we can produce various animation styles, changes in time of day, and more complex changes of subject, such as turning a hiker into an alien (see Fig. 5). Please see the supplementary material for more results.

Using CLIP image embeddings as the content representation allows users to specify content through images. As an example application, we demonstrate character replacement in Fig. 9. For every video in a set of six videos, we re-synthesize it five times, each time providing a single content image taken from another video in the set. We can retain content characteristics with $t_s = 3$ despite large differences in their pose and shape.

Lastly, we illustrate the use of masked video editing in Fig. 8, where the model predicts everything outside the masked area(s) while retaining the original content inside the masked area. Notably, this technique resembles inpainting with diffusion models [43, 25].

4.2. User Study

We benchmark against Text2Live [5], a recent approach for text-guided video editing that employs layered neural atlases [21]. As a baseline, we compare against SDEdit [26] in two ways; per-frame generated results and a first-frame result propagated by a few-shot video stylization method [55] (IVS). We also include two depth-based versions of Stable Diffusion; one trained with depth-conditioning [2] and one that retains past results based on depth estimates [9]. We also include an ablation: applying SDEdit to our video model trained without conditioning on a structure representation (ours, $\sim s$).

We judge the success of our method qualitatively based on a user study. We run the user study using Amazon Mechanical Turk (AMT) on an evaluation set of 35 representative video editing prompts. For each example, we ask 5 annotators to compare faithfulness to the video editing prompt (”Which video better represents the provided edited caption?”) between a baseline and our method, presented in random order, and use a majority vote for the final result.

The results can be found in Fig. 7. Across all compared methods, results from our approach are preferred roughly 3 out of 4 times. A visual comparison among the methods can be found in the supplementary. We observe that SDEdit is sensitive to the editing strength. Low values fail to achieve the desired editing effect whereas high values change the structure of the input. Even with a fixed seed, both style and structure can change in unnatural ways between frames as their relationship is ignored by image-based approaches. Propagation of SDEdit outputs (IVS) leads to more consistent results but often introduces propagation artifacts especially with large motion. Depth-SD produces accurate, structure-preserving edits for individual frames but frames are inconsistent across time. The outputs of Text2Live tend to be temporally smooth due to its reliance on Layered Neural Atlases [21], but it often produces edits that represent the edit prompt inaccurately. A direct comparison with Text2Live is difficult as it requires input masks and separate edit prompts for foreground and background. In addition, computing a neural atlas takes about 10 hours whereas our approach requires approximately a minute.

4.3. Quantitative Evaluation

We quantify trade-offs between frame consistency and prompt consistency with the following two metrics.

Frame consistency We compute CLIP image embeddings on all frames of output videos and report the average cosine similarity between all pairs of consecutive frames.

Prompt consistency We compute CLIP image embeddings on all frames of output videos and the CLIP text embedding of the edit prompt. We report average cosine similarity between text and image embedding over all frames.

Fig. 6 shows the results of each model using our frame consistency and prompt consistency metrics. Our model tends to outperform the baseline models in both aspects (placed higher in the upper-right quadrant of the graph). We also notice a slight tradeoff with increasing the strength parameters in the baseline models: larger strength scales implies higher prompt consistency at the cost of lower frame consistency. Increasing the temporal scale (ω_t) of our model

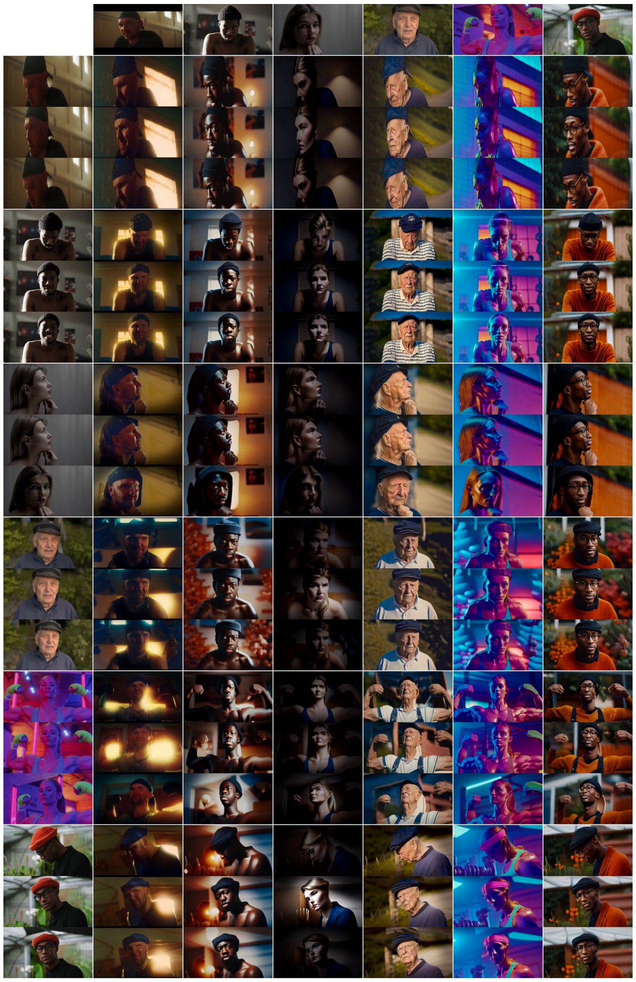


Figure 9. **Image Prompting:** We combine the structure of a driving video (first column) with content from other videos (first row).

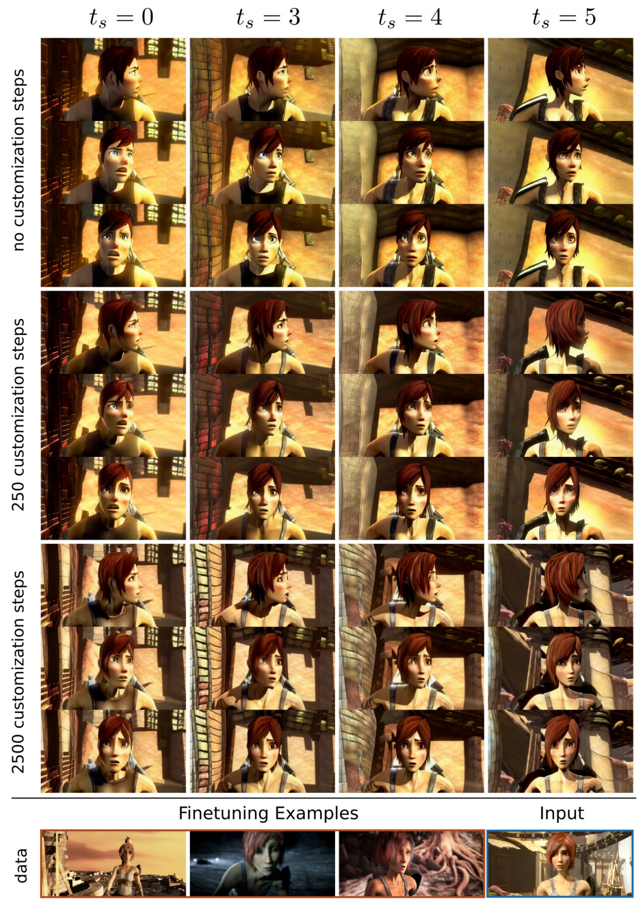


Figure 10. **Controlling Fidelity:** We obtain control over structure and appearance-fidelity. Each cell shows three frames produced with decreasing structure-fidelity t_s (left-to-right) and increasing number of customization training steps (top-to-bottom). The bottom shows examples of images used for customization (red border) and the input image (blue border). Same driving video as in Fig. 1.

results in higher frame consistency but lower prompt consistency. We also observe that an increased structure scale (t_s) results in higher prompt consistency as the content becomes less determined by the input structure.

4.4. Customization

Customization of image models enables generation of previously unseen content, such as specific people or styles, based on a small dataset used for finetuning [41]. We finetune our depth-conditioned latent video diffusion model on a set of 15-30 images and produce videos containing the desired subject. Half of the batch elements are of the subject and the other half belong to the original training dataset.

Fig. 10 shows an example with different numbers of customization steps as well as different levels of structure preservation t_s. Customization improves fidelity to the style and appearance of the character. In combination with higher t_s values, accurate animations are possible despite using a driving video of a person with different characteristics.

5. Conclusion

Our latent video diffusion model synthesizes new videos given structure and content information. We ensure structural consistency by conditioning on depth estimates while content is controlled with images or natural language. Temporal layers and joint image-video training achieve stable results across frames. A novel guidance method, inspired by classifier-free guidance, allows for control over temporal consistency. By training on depth maps with varying degrees of detail, we can adjust the level of structure preservation. This, together with model customization, improves content fidelity. Our quantitative evaluation and user study show that our method is preferred over related approaches. Future works should investigate other conditioning data, such as facial landmarks and pose estimates, and additional 3d-priors to improve generated results. Our model is intended for creative applications in content creation, but we realize the risks of dual-use and hope that further work will be aimed at combating abuse of generative models.

References

[1] Dinesh Acharya, Zhiwu Huang, Danda Pani Paudel, and Luc Van Gool. Towards high resolution video generation with progressive growing of sliced wasserstein gans. *arXiv preprint arXiv:1810.02419*, 2018. 2

[2] Stability AI. Stable diffusion depth. https://github.com/Stability-AI/stablediffusion, 2022. 7

[3] Yogesh Balaji, Seungjun Nah, Xun Huang, Arash Vahdat, Jiaming Song, Karsten Kreis, Miika Aittala, Timo Aila, Samuli Laine, Bryan Catanzaro, Tero Karras, and Ming-Yu Liu. ediff-i: Text-to-image diffusion models with ensemble of expert denoisers. *arXiv preprint arXiv:2211.01324*, 2022. 2, 4

[4] Arpit Bansal, Eitan Borgnia, Hong-Min Chu, Jie S. Li, Hamid Kazemi, Furong Huang, Micah Goldblum, Jonas Geiping, and Tom Goldstein. Cold diffusion: Inverting arbitrary image transforms without noise, 2023.

[5] Omer Bar-Tal, Dolev Ofri-Amar, Rafail Fridman, Yoni Kasten, and Tali Dekel. Text2live: Text-driven layered image and video editing. In *European Conference on Computer Vision*, pages 707–723. Springer, 2022. 2, 7

[6] Tim Brooks, Janne Hellsten, Miika Aittala, Ting-Chun Wang, Timo Aila, Jaakko Lehtinen, Ming-Yu Liu, Alexei A Efros, and Tero Karras. Generating long videos of dynamic scenes. 2022. 2

[7] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc., 2020. 6

[8] Dongdong Chen, Jing Liao, Lu Yuan, Nenghai Yu, and Gang Hua. Coherent online video style transfer. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1105–1114, 2017. 2

[9] deforum. Deforum stable diffusion. https://github.com/deforum/stable-diffusion, 2022. 2, 7

[10] Ming Ding, Zhuoyi Yang, Wenyi Hong, Wendi Zheng, Chang Zhou, Da Yin, Junyang Lin, Xu Zou, Zhou Shao, Hongxia Yang, and Jie Tang. Cogview: Mastering text-to-image generation via transformers. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 19822–19835. Curran Associates, Inc., 2021. 2

[11] Songwei Ge, Thomas Hayes, Harry Yang, Xi Yin, Guan Pang, David Jacobs, Jia-Bin Huang, and Devi Parikh. Long video generation with time-agnostic vqgan and time-sensitive transformer. *arXiv preprint arXiv:2204.03638*, 2022. 2

[12] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014. 2

[13] Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P Kingma, Ben Poole, Mohammad Norouzi, David J Fleet, et al. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*, 2022. 2, 3

[14] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 6840–6851. Curran Associates, Inc., 2020. 2, 3

[15] Jonathan Ho, Chitwan Saharia, William Chan, David J Fleet, Mohammad Norouzi, and Tim Salimans. Cascaded diffusion models for high fidelity image generation. *J. Mach. Learn. Res.*, 23:47–1, 2022. 2

[16] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance, 2022. 6

[17] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *arXiv:2204.03458*, 2022. 2, 4

[18] Wenyi Hong, Ming Ding, Wendi Zheng, Xinghan Liu, and Jie Tang. Cogvideo: Large-scale pretraining for text-to-video generation via transformers. *arXiv preprint arXiv:2205.15868*, 2022. 2

[19] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017. 2

[20] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. In *Proc. NeurIPS*, 2022. 2

[21] Yoni Kasten, Dolev Ofri, Oliver Wang, and Tali Dekel. Layered neural atlases for consistent video editing. *ACM Transactions on Graphics (TOG)*, 40(6):1–12, 2021. 2, 7

[22] Zhifeng Kong and Wei Ping. On fast sampling of diffusion probabilistic models. *arXiv preprint arXiv:2106.00132*, 2021. 2, 6

[23] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *ICML*, 2022. 6

[24] Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. DPM-solver: A fast ODE solver for diffusion probabilistic model sampling in around 10 steps. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022. 2, 6

[25] Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting

using denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11461–11471, 2022. 2, 7

[26] Chenlin Meng, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. Sdedit: Image synthesis and editing with stochastic differential equations. *CoRR*, abs/2108.01073, 2021. 7

[27] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning*, pages 8162–8171. PMLR, 2021. 2

[28] Alexander Quinn Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob Mcgrew, Ilya Sutskever, and Mark Chen. GLIDE: Towards photorealistic image generation and editing with text-guided diffusion models. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato, editors, *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 16784–16804. PMLR, 17–23 Jul 2022. 2

[29] Yaniv Nikankin, Niv Haim, and Michal Irani. Sinfusion: Training diffusion models on a single image or video. *arXiv preprint arXiv:2211.11743*, 2022. 2

[30] William Peebles and Saining Xie. Scalable diffusion models with transformers, 2022. 2

[31] Jordi Pont-Tuset, Federico Perazzi, Sergi Caelles, Pablo Arbeláez, Alexander Sorkine-Hornung, and Luc Van Gool. The 2017 davis challenge on video object segmentation. *arXiv:1704.00675*, 2017. 6

[32] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. 2, 4

[33] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(1), jun 2022. 2

[34] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents, 2022. 1, 2, 4

[35] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8821–8831. PMLR, 18–24 Jul 2021. 4

[36] René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44:1623–1637, 2019. 6

[37] Alex Rav-Acha, Pushmeet Kohli, Carsten Rother, and Andrew William Fitzgibbon. Unwrap mosaics: a new representation for video editing. *ACM SIGGRAPH 2008 papers*, 2008. 2

[38] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2021. 1, 2, 3, 6

[39] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 3

[40] Manuel Ruder, Alexey Dosovitskiy, and Thomas Brox. Artistic style transfer for videos. In Bodo Rosenhahn and Bjoern Andres, editors, *Pattern Recognition*, pages 26–36, Cham, 2016. Springer International Publishing. 2

[41] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. *arXiv preprint arXiv:2208.12242*, 2022. 8

[42] Alexander S. Disco diffusion v5.2 - warp fusion. https://github.com/Sxela/DiscoDiffusion-Warp, 2022. 2

[43] Chitwan Saharia, William Chan, Huiwen Chang, Chris A. Lee, Jonathan Ho, Tim Salimans, David J. Fleet, and Mohammad Norouzi. Palette: Image-to-image diffusion models, 2021. 7

[44] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S Sara Mahdavi, Rapha Gontijo Lopes, et al. Photorealistic text-to-image diffusion models with deep language understanding. *arXiv preprint arXiv:2205.11487*, 2022. 2

[45] Masaki Saito, Eiichi Matsumoto, and Shunta Saito. Temporal generative adversarial nets with singular value clipping. In *Proceedings of the IEEE international conference on computer vision*, pages 2830–2839, 2017. 2

[46] Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. In *International Conference on Learning Representations*, 2022. 2, 3

[47] Robin San-Roman, Eliya Nachmani, and Lior Wolf. Noise estimation for generative diffusion models. *arXiv preprint arXiv:2104.02600*, 2021. 2

[48] Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. *arXiv preprint arXiv:2111.02114*, 2021. 4

[49] Uriel Singer, Adam Polyak, Thomas Hayes, Xiaoyue Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, Devi Parikh, Sonal Gupta, and Yaniv Taigman. Make-a-video: Text-to-video generation without text-video data. *ArXiv*, abs/2209.14792, 2022. 2, 4

[50] Ivan Skorokhodov, Sergey Tulyakov, and Mohamed Elhoseiny. Stylegan-v: A continuous video generator with the price, image quality and perks of stylegan2, 2021. 2

[51] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In Francis Bach and David Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings*

of Machine Learning Research, pages 2256–2265, Lille, France, 07–09 Jul 2015. PMLR. 2, 3

[52] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021. 2, 6

[53] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020. 2

[54] Guy Tevet, Sigal Raab, Brian Gordon, Yonatan Shafir, Amit H Bermano, and Daniel Cohen-Or. Human motion diffusion model. *arXiv preprint arXiv:2209.14916*, 2022. 2

[55] Ondřej Texler, David Futschik, Michal Kučera, Ondřej Jamriška, Šárka Sochorová, Menglei Chai, Sergey Tulyakov, and Daniel Sýkora. Interactive video stylization using few-shot patch-based training. *ACM Transactions on Graphics*, 39(4):73, 2020. 2, 7

[56] Sergey Tulyakov, Ming-Yu Liu, Xiaodong Yang, and Jan Kautz. MoCoGAN: Decomposing motion and content for video generation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1526–1535, 2018. 2

[57] Arash Vahdat, Karsten Kreis, and Jan Kautz. Score-based generative modeling in latent space. *Advances in Neural Information Processing Systems*, 34:11287–11302, 2021. 2

[58] Ruben Villegas, Mohammad Babaeizadeh, Pieter-Jan Kindermans, Hernan Moraldo, Han Zhang, Mohammad Taghi Saffar, Santiago Castro, Julius Kunze, and Dumitru Erhan. Phenaki: Variable length video generation from open domain textual description, 2022. 2

[59] Carl Vondrick, Hamed Pirsiavash, and Antonio Torralba. Generating videos with scene dynamics. *Advances in neural information processing systems*, 29, 2016. 2

[60] Ting-Chun Wang, Ming-Yu Liu, Andrew Tao, Guilin Liu, Jan Kautz, and Bryan Catanzaro. Few-shot video-to-video synthesis. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019. 2

[61] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Guilin Liu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. Video-to-video synthesis. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2018. 2

[62] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 2

[63] Jay Zhangjie Wu, Yixiao Ge, Xintao Wang, Stan Weixian Lei, Yuchao Gu, Wynne Hsu, Ying Shan, Xiaohu Qie, and Mike Zheng Shou. Tune-a-video: One-shot tuning of image diffusion models for text-to-video generation. *arXiv preprint arXiv:2212.11565*, 2022. 2

[64] Wilson Yan, Yunzhi Zhang, Pieter Abbeel, and Aravind Srinivas. Videogpt: Video generation using vq-vae and transformers. *arXiv preprint arXiv:2104.10157*, 2021. 2

[65] Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, Yinfei Yang, Burcu Karagol Ayan, Ben Hutchinson, Wei Han, Zarana Parekh, Xin Li, Han Zhang, Jason Baldridge, and Yonghui Wu. Scaling autoregressive models for content-rich text-to-image generation, 2022. 2

[66] Xiaohui Zeng, Arash Vahdat, Francis Williams, Zan Gojcic, Or Litany, Sanja Fidler, and Karsten Kreis. Lion: Latent point diffusion models for 3d shape generation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. 2

[67] Daquan Zhou, Weimin Wang, Hanshu Yan, Weiwei Lv, Yizhe Zhu, and Jiashi Feng. Magicvideo: Efficient video generation with latent diffusion models, 2022. 2