Pix2Video: Video Editing using Image Diffusion

Duygu Ceylan[1*] Chun-Hao P. Huang[1*] Niloy J. Mitra[1,2]
[1]Adobe Research [2]University College London

Abstract

Image diffusion models, trained on massive image collections, have emerged as the most versatile image generator model in terms of quality and diversity. They support inverting real images and conditional (e.g., text) generation, making them attractive for high-quality image editing applications. We investigate how to use such pre-trained image models for text-guided video editing. The critical challenge is to achieve the target edits while still preserving the content of the source video. Our method works in two simple steps: first, we use a pre-trained structure-guided (e.g., depth) image diffusion model to perform text-guided edits on an anchor frame; then, in the key step, we progressively propagate the changes to the future frames via self-attention feature injection to adapt the core denoising step of the diffusion model. We then consolidate the changes by adjusting the latent code for the frame before continuing the process. Our approach is training-free and generalizes to a wide range of edits. We demonstrate the effectiveness of the approach by extensive experimentation and compare it against four different prior and parallel efforts (on ArXiv). We demonstrate that realistic text-guided video edits are possible, without any compute-intensive preprocessing or video-specific finetuning. https://duyguceylan.github.io/pix2video.github.io/.

1. Introduction

Diffusion-based algorithms [8, 18, 48] have emerged as the generative model of choice for image creation. They are stable to train (even over huge image collections), produce high-quality results, and support conditional sampling. Additionally, one can invert [31, 48] a given image into a pre-trained diffusion model and subsequently edit using only textual guidance [15]. Such a generic workflow, to handle real images and interact using semantic text prompts, is an exciting development and opens the door for many downstream content creation tasks.

However, the same workflow is barely available for videos where the development of video diffusion models is still in its infancy [32, 46, 61]. Not surprisingly, naively applying an image-based workflow to each video frame produces inconsistent results (see Figure 1). Alternately, while it is possible to use a single frame for style guidance and employ video stylization propagation [20], the challenge lies in stylizing new content revealed under changing occlusions across frames.

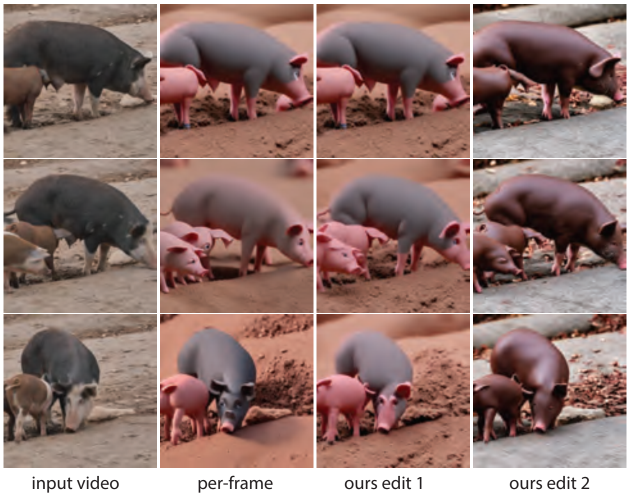


Figure 1: There has been exciting advancements in large scale image generation models [41]. When applied independently to a sequence of images ('per-frame'), however, such methods produce inconsistent results across frames. Our method uses a pre-trained and fixed image generation model to consistently edit a video clip based on a target text prompt. We show examples of two different edits ('ours').

In this paper, we explore the feasibility of *editing a video clip using a pre-trained image diffusion model and text instructions with no additional training*. We start by inverting the input video clip and expecting the user to edit, using textual prompts, one of the video frames. The goal is then to *consistently* propagate the edit across the rest of the video. The challenge is to balance between respecting the user edit and maintaining the plausibility and temporal coherency of the output video. Image generation models already generate images faithful to the edit prompt. Hence, what remains

[*] These authors contributed equally to this work

challenging is to propagate the edit in a temporally coherent manner.

Temporal coherency requires preserving the appearance across neighboring frames while respecting the motion dynamics. Leveraging the fact that the spatial features of the self attention layers are influential in determining both the structure and the appearance of the generated images, we propose to inject features obtained from the previously edited frames into the self attention layer of the current frame. This feature injection notably adapts the self attention layers to perform cross-frame attention and enables the generation of images with coherent appearance characteristics. To further improve consistency, we adopt a *guided diffusion* strategy in which we update the intermediate latent codes to enforce similarity to the previous frame before we continue the diffusion process. While the image generation model cannot reason about motion dynamics explicitly, recent work has shown that generation can be conditioned on static structural cues such as depth or segmentation maps [63]. Being disentangled from the appearance, such structural cues provide a path to reason about the motion dynamics. Hence, we utilize a depth-conditioned image generation model and use the predicted depth from each frame as additional input.

We term our method *Pix2Video* and evaluate it on various real video clips demonstrating both local (e.g., changing the attribute of a foreground object) and global (e.g., changing the style) edits. We compare with several state-of-the-art approaches including diffusion-based image editing methods applied per frame [30], patch-based video based stylization methods [20], neural layered representations that facilitate consistent video editing [3], and concurrent diffusion based video editing methods [58]. We show that Pix2Video is on par with or better than the baselines *without* requiring any compute-intensive preprocessing [3] or any video-specific finetuning [58]. This can be seen in Figure 1 "ours" columns where the appearance of the foreground objects are more consistent across frames than per-frame editing.

In summary, we present a *training free* approach that utilizes pre-trained large scale image generation models for video editing. Our method does not require pre-processing and does not incur any additional overhead during inference stage. This ability to use an existing image generation model paves the way to bring exciting advancements in controlled image editing to videos at no cost. *Source code is available at the project page.*

2. Related Work

2.1. Image generation and editing

While many deep generative models, e.g., GAN [12], have demonstrated the ability to generate realistic images [4, 22], recently, diffusion models have become the choice of models due to the high quality output they achieve on large scale datasets [9]. Denoising Diffusion Probabilistic Model (DDPM) [18] and its variant Denoising Diffusion Implicit Model (DDIM) [48] have been widely used for unconditional text-to-image generation. Several large scale text-to-image generation models [34, 39, 44], which operate on the pixel space have been presented, achieving very high quality results. Rombach et al. [41] have proposed to work in a latent space which has lead to the widely adopted open source Stable Diffusion model. We refer the readers to a recent survey [8] and the extensive study [21] for a detailed discussion on diffusion models.

In the presence of high quality text conditioned image generation models, several recent works have focused on utilizing additional control signals for generation or editing existing images. Palette [43] has shown various image-to-image translation applications using a diffusion model including colorization, inpainting, and uncropping. Several methods have focused on providing additional control signals such as sketches, segmentation maps, lines, or depth maps by adapting a pretrained image generation model. These methods work by either finetuning [55] an existing model, introducing adapter layers [33] or other trainable modules [54, 63], or utilizing an ensemble of denoising networks [2]. Since our model uses a pretrained image diffusion model, it can potentially use any model that accepts such additional control signals. In another line of work, methods have focused on editing images while preserving structures via attention layer manipulation [15, 52], spatial guidance [7], or per-instance finetuning [24]. Our method also performs attention layer manipulation, specifically in the self attention layers, along with a latent update at each diffusion step. Unlike single image based editing work, however, we utilize previous frames when performing these steps. We would also like to emphasize that the edit of the anchor frame for our method can potentially be performed with any such method that utilize the same underlying image generation model.

2.2. Video generation and editing

Until recently, GANs have been the method of choice for video generation, with many works designed towards unconditional generation [5, 13, 45, 47, 51, 62, 64]. In terms of conditional generation, several methods have utilized guidance channels such as segmentation masks or keypoints [56, 57]. However, most of these methods are trained on specific domains. One particular domain where very powerful image generators such as StyleGAN [22] exist is faces. Hence, several works have explored generating videos by exploring the latent space of such an image based generator [37, 59]. While we also exploit an image generation model, we are not focused on a specific domain.

With the success of text-to-image generation models, there has been recent attempts in text-to-video generation models using architectures such as transformers [19, 53, 60] or diffusion models [14, 17, 46, 61]. However, such models are still in their infancy compared to images, both due to the complexity of temporal generation as well as large scale annotated video datasets not being comparable in size to images. Concurrent works [10, 32] explore mixed image and video based training to address this limitation. In another concurrent work, Wu et al. [58] inflate an image diffusion model and finetune on a specific input video to enable editing tasks. In our work, we use the pretrained image diffusion model as it is with no additional training.

Layered neural representations [29] have recently been introduced, providing another direction for editing videos. Layered neural atlases [23] are such representations that map the foreground and background of a video to a canonical space. Text2Live [3] combines such a representation with text guidance to show compelling video editing results. While impressive, the computation of such neural representations includes extensive per-video training (7-10 hours), which limits their applicability in practice. Loeschcke et al. [28] also utilizes layered neural atlases with a CLIP based optimization to stylize a foreground object in a video.

Finally, video stylization is a specific type of editing task where the style of an example frame is propagated to the video. While some methods utilize neural feature representations to perform this task [42], Jamriska et al. [20] consider a patch-based synthesis approach using optical flow. In a follow-up work [50], they provide a per-video fast patch-based training setup to replace traditional optical flow. Both methods achieve high quality results but are limited when the input video shows regions that are not visible in the provided style keyframes. They rely on having access to multiple stylized keyframes in such cases. However, generating consistent multiple keyframes itself is a challenge. Our method can also be perceived as orthogonal since the (subset of) frames generated by our method can subsequently be used as keyframe inputs to these models.

3. Method

Given a sequence of frames of a video clip, $\mathcal{I} := \{I_1, \dots, I_n\}$, we would like to generate a new set of images $\mathcal{I}' := \{I'_1, \dots, I'_n\}$ that reflects an edit denoted by a target text prompt $\mathcal{P}'$. For example, given a video of a car, the user may want to generate an edited video where attributes of the car, such as its color, are edited. We aim to exploit the power of a pretrained and fixed large-scale image diffusion model to perform such manipulations as coherently as possible, without the need for any example-specific finetuning or extensive training. We achieve this goal by manipulating the internal features of the diffusion model (Section 3.1) along with additional guidance constraints (Section 3.2).

Given that the fixed image generation model is trained with only single images, it cannot reason about dynamics and geometric changes that happen in an input video. In light of the recent progress in conditioning image generation models with various structure cues [2, 11, 54], we observe that this additional structure channel is effective in capturing the motion dynamics. Hence, we build our method on a depth-conditioned Stable Diffusion model [1]. Given $\mathcal{I}$, we perform per-frame depth prediction [40] and utilize it as additional input to the model.

3.1. Self-attention Feature Injection

In the context of static images, a large-scale image generation diffusion model typically consists of a U-Net architecture composed of residual, self-attention, and cross-attention blocks. While the cross-attention blocks are effective in terms of achieving faithfulness to the text prompt, self-attention layers are effective in determining the overall structure and the appearance of the image. At each diffusion step t, the input features f_t^l to the self-attention module at layer l, are projected into *queries*, *keys*, and *values* by matrices W^Q, W^K, and W^V, respectively to obtain queries Q^l, keys K^l, and values V^l. The output of the attention block is then computed as:

$$Q^l = W^Q f_t^l; K^l = W^K f_t^l; V^l = W^v f_t^l$$
$$\hat{f}_t^l = \text{softmax}(Q^l (K^l)^\top)(V^l),$$

where d_k denotes the dimension of Q and K. In other words, for each location in the current spatial feature map f_t^l, a weighted summation of every other spatial features is computed to capture global information. Extending to the context of videos, our method captures the interaction across the input image sequence by manipulating the input features to the self-attention module. Specifically, we inject the features obtained from the previous frames. A straightforward approach is to attend to the features $f_t^{j,l}$ of an earlier frame j while generating the features $f_t^{i,l}$ for frame i as,

$$Q^{i,l} = W^Q f_t^{i,l}; K^{i,l} = W^K f_t^{j,l}; V^{i,l} = W^v f_t^{j,l}.$$

With such feature injection, the current frame is able to utilize the context of the previous frames and hence preserve the appearance changes. A natural question is whether an explicit, potentially recurrent, module can be employed to fuse and represent the state of the previous frame features without explicitly attending to a specific frame. However, the design and training of such a module is not trivial. Instead, we rely on the pre-trained image generation model to perform such fusion implicitly. For each frame i, we inject the features obtained from frame $i-1$. Since the editing is performed in a frame-by-frame manner, the features of

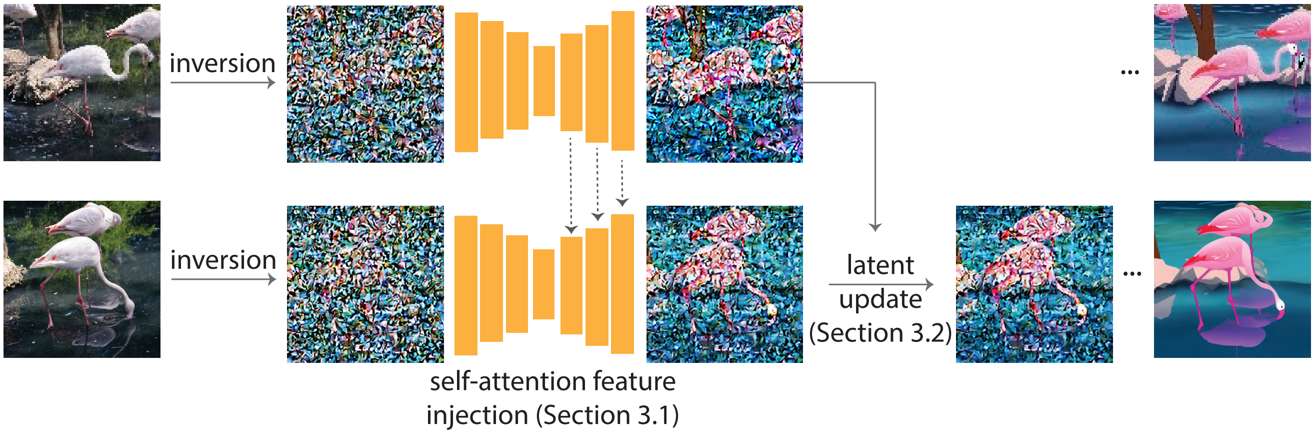

inversion
inversion
...
...
latent update (Section 3.2)
self-attention feature injection (Section 3.1)

Figure 2: **Method pipeline**. Pix2Video first inverts each frame with DDIM-inversion and consider it as the initial noise x_T for the denoising process. To edit each frame $i > 1$ (lower row), we select a reference frame (upper row), inject its self-attention features to the UNet. At each diffusion step, we also update the latent of the current frame guided by the latent of the reference frame. In practice, we consider both $i-1$ (previous) and $i=1$ (anchor) frames as reference for feature injection, while we use only the previous frame for the guided latent update.

$i-1$ are computed by attending to frame $i-2$. Consequently, we have an implicit way of aggregating the feature states. In Section 4, we demonstrate that while attending to the previous frame helps to preserve the appearance, in longer sequences it shows the limitation of diminishing the edit. Attending to an additional anchor frame avoids this forgetful behavior by providing a global constraint on the appearance. Hence, in each self-attention block, we concatenate features from frames a and $i-1$ to compute the key and value pairs. In our experiments, we set $a=1$, i.e., the first frame.

$$Q^{i,l} = W^Q f_t^{i,l};$$
$$K^{i,l} = W^K[f_t^{a,l}, f_t^{i-1,l}]; V^{i,l} = W^v[f_t^{a,l}, f_t^{i-1,l}].$$

We perform the above feature injection in the decoder layers of the UNet, which we find effective in maintaining appearance consistency. As shown in the ablation study, and also reported by the concurrent work of Tumanyan et al. [52], the deeper layers of the decoder capture high resolution and appearance-related information and already result in generated frames with similar appearance but small structural changes. Performing the feature injection in earlier layers of the decoder enables us to avoid such high-frequency structural changes. We do not observe further significant benefit when injecting features in the encoder of the UNet and observe slight artifacts in some examples.

3.2. Guided Latent Update

While self-attention feature injection effectively generates frames that have coherent appearance, it can still suffer from temporal flickering. In order to improve the temporal stability, we exploit additional guidance to update the latent variable at each diffusion step along the lines of classifier guidance [34]. To perform such an update, we first formulate an energy function that enforces consistency.

Stable Diffusion [1, 14], like many other large-scale image diffusion models, is a denoising diffusion implicit model (DDIM) where at each diffusion step, given a noisy sample x_t, a prediction of the noise-free sample $\hat{x}_0$, along with a direction that points to x_t, is computed. Formally, the final prediction of x_{t-1} is obtained by:

$$x_{t-1} = \sqrt{\alpha_{t-1}} \underbrace{\hat{x}_0^t}_{\text{predicted `}x_0\text{'}} + \underbrace{\sqrt{1-\alpha_{t-1}-\sigma_t^2\epsilon_\theta(x_t,t)}}_{\text{direction pointing to }x_t} + \underbrace{\sigma_t\epsilon_t}_{\text{random noise}},$$
$$\hat{x}_0^t = \frac{x_t - \sqrt{1-\alpha_t}\epsilon_\theta^t(x_t)}{\sqrt{\alpha_t}},$$

where α_t and σ_t are the parameters of the scheduler and ϵ_θ is the noise predicted by the UNet at the current step t. The estimate $\hat{x}_0^t$ is computed as a function of x_t and indicates the final generated image. Since our goal is to generate similar consecutive frames eventually, we define an L2 loss function $g(\hat{x}_0^{i,t}, \hat{x}_0^{i-1,t}) = \|\hat{x}_0^{i,t} - \hat{x}_0^{i-1,t}\|_2^2$ that compares the predicted clean images at each diffusion step t between frames $i-1$ and i. We update x_{t-1}^i, the current noise sample of a frame i at diffusion step t, along the direction that minimizes g:

$$x_{t-1}^i \leftarrow x_{t-1}^i - \delta_{t-1}\nabla_{x_t^i} g(\hat{x}_0^{t,i-1}, \hat{x}_0^{t,i}), \qquad (1)$$

Algorithm 1

```
Input $\mathcal{I} = \{I_1, \dots I_n\}$, text prompt $\mathcal{P}'$, $T = 50$ diffusion steps, a pretrained diffusion model SD
Output $\mathcal{I}' = \{I'_1, \dots I'_n\}$
$\mathcal{X}^{\mathcal{T}} \leftarrow \{x_T^1, ..., x_T^n\}$ by DDIM inversion
$\mathcal{I}' = \emptyset, \mathcal{F}^{anchor} = \emptyset, \mathcal{F}^{prev} = \emptyset, \hat{\mathcal{X}}_0^{prev} = \emptyset$
for f $\in$ [1,$n$]: do
    $\hat{\mathcal{X}}_0^{tmp} = [], \hat{\mathcal{F}}^{tmp} = []$
    $x_t = x_T^f$
    $\delta_{t-1} = 100$ if $t-1 < 25$ else $\delta_{t-1} = 0$
    for t $\in$ [1,$T$]: do
        if $f = 1$ then
            $f^{anchor} = \emptyset, f^{prev} = \emptyset, \hat{x}_0^{prev} = \emptyset$
        else
            $f^{anchor} = \mathcal{F}^{anchor}[t]$
            $f^{prev} = \mathcal{F}^{prev}[t]$
            $\hat{x}_0^{prev} = \hat{\mathcal{X}}_0^{prev}[t]$
        end if
        $x_{t-1}, \hat{x}_0^t, f^t = SD(x_t, t, \mathcal{P}, f^{anchor}, f^{prev})$
        if $f = 1$ then
            $\mathcal{F}^{anchor} \leftarrow \mathcal{F}^{anchor} \cup \{f^t\}$
        else
            $x_{t-1} \leftarrow x_{t-1} - \delta_{t-1} \nabla_{x_t} \|\hat{x}_0^{prev} - \hat{x}_0^t\|_2^2$
        end if
        $\mathcal{F}^{tmp} \leftarrow \mathcal{F}^{tmp} \cup \{f^t\}$
        $\hat{\mathcal{X}}_0^{tmp} \leftarrow \hat{\mathcal{X}}_0^{tmp} \cup \{\hat{x}_0^t\}$
    end for
    $\hat{\mathcal{X}}_0^{prev} = \hat{\mathcal{X}}_0^{tmp}$
    $\mathcal{F}^{prev} = \mathcal{F}^{tmp}$
    $\mathcal{I}' \leftarrow \mathcal{I}' \cup \{I'_f\}$
end for
```

where δ_{t-1} is a scalar that determines the step size of the update. We empirically observe that performing one gradient update at each diffusion step is sufficient and we set $\delta_{t-1} = 100$ in our experiments. We perform this update process for the early denoising steps, namely the first 25 steps among the total number of 50 steps, as the overall structure of the generated image is already determined in the earlier diffusion steps [25]. Performing the latent update in the remaining steps often results in lower-quality images.

Finally, the initial noise used to edit each frame also significantly affects the temporal coherency of the generated results. We use an inversion mechanism, DDIM inversion [48], while other inversion methods aiming to preserve the editability of an image can be used [31] as well. To get a source prompt for inversion, we generate a caption for the first frame of the video using a captioning model [26]. We provide the overall steps of our method in Algorithm 1.

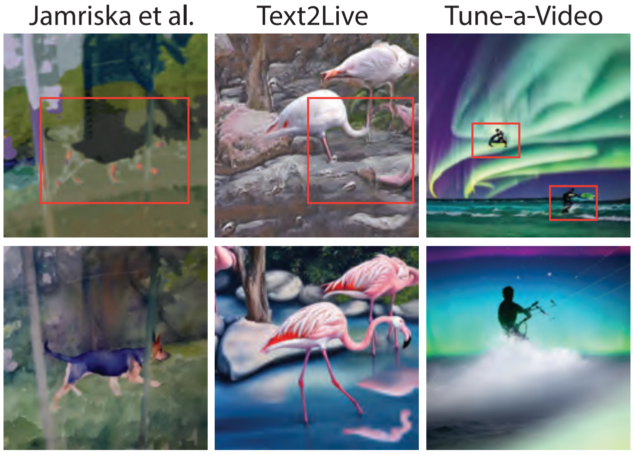


Figure 3: **Comparisons**. Top: results from baselines; bottom: our results. The method of Jamriska et al. relies on optical flow to propagate edits in a temporally coherent manner but fails as new content becomes visible. Text2Live suffers when multiple foreground objects are present and a neural atlas cannot be computed robustly. Without an explicit notion of structure, Tune-a-Video is not able to preserve the structure in the input video for some edits.

4. Evaluation

Dataset. Following [3, 10, 58], we evaluate Pix2Video on videos obtained from the DAVIS dataset [36]. For videos that have been used in previous or concurrent work, we use editing prompts provided by such work. For other videos, we generate editing prompts by consulting a few users. The length of these videos ranges from 50 to 82 frames.

Baselines. We compare Pix2Video with both state-of-the-art image and video editing approaches. (i) The method of Jamriska et al. [20] propagates the style of a set of given frames to the input video clip. We use the edited anchor frame as a keyframe. (ii) We compare to a recent text-guided video-editing method, Text2Live [3]. We note that this method first requires the computation of a neural atlas [23] for the foreground and background layers of a video which takes approximately 7-8 hours per video. Given the neural atlas, the method further finetunes the text-to-

Table 1: Compared to other video-based baselines, our method does not require a heavy pre-processing stage nor a per-video or per-edit finetuning strategy.

	pre-processing	finetuning	layering
Jamriska et al. [20]	no	no	no
Text2Live [3]	7-8 hours	30 min	yes
Tune-a-Video [58]	75sec	10 min	no
ours	125sec	no	no

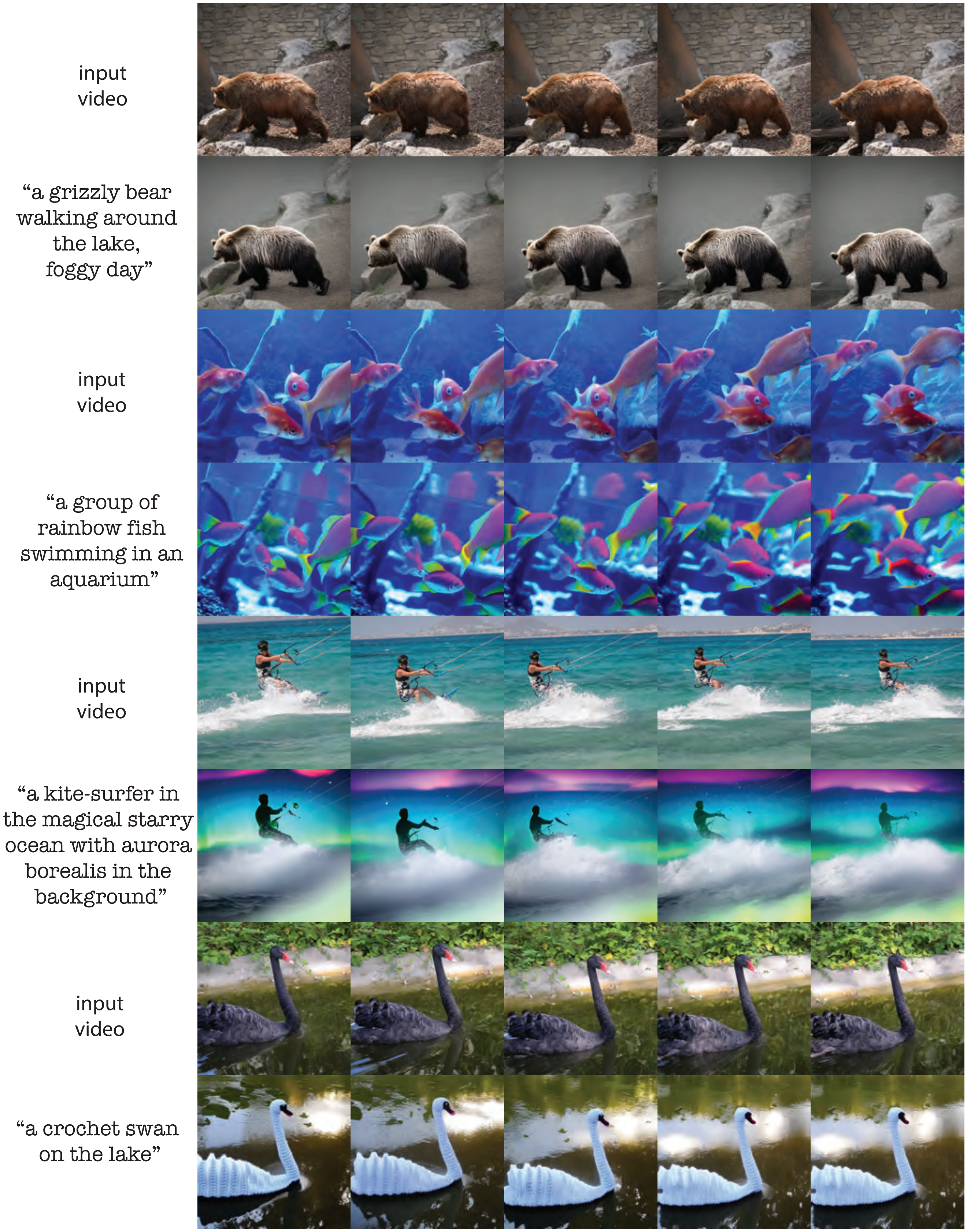

input video
“a grizzly bear walking around the lake, foggy day”
input video
“a group of rainbow fish swimming in an aquarium”
input video
“a kite-surfer in the magical starry ocean with aurora borealis in the background”
input video
“a crochet swan on the lake”

Figure 4: **Text-guided video edits.** For each example, we show frames from the input video at the top and the corresponding edited frames, and the edit prompt at the bottom. Please refer to the project webpage for the videos.

image generation model which takes another 30 minutes. (iii) We also compare against SDEdit [30] where we add noise to each input frame and denoise conditioned on the edit prompt. We experiment with different strengths of noise added and use the depth-conditioned Stable Diffusion [1] as in our backbone diffusion model. (iv) Finally, we also consider the concurrent Tune-a-Video [58] method, which performs a video-specific finetuning of a pretrained image model. Since this method generates only a limited number of frames, we generate 24 frames by sampling every other frame in the input video following the setup provided by the authors. Note that this method is not conditioned on any structure cues like depth. We summarize the characteristics of each baseline in Table 1.

Metrics. We expect a successful video edit to faithfully reflect the edited prompt and be temporally coherent. To capture the *faithfulness*, we follow [17, 58] and report CLIP score [16, 35], which is the cosine similarity between the CLIP embedding [38] of the edit prompt and the embedding of each frame in the edited video. We refer to this metric as "CLIP-Text". To measure the *temporal coherency*, we measure the average CLIP similarity between the image embeddings of consecutive frames in the edited video ("CLIP-Image"). We observe that CLIP image embeddings encode more global appearance than local details. Hence, we also compute the optical flow between consecutive frames [49], and warp each frame in the edited video to the next using the flow. We compute the average mean-squared pixel error between each warped frame and its corresponding target frame as "Pixel-MSE". We note that this metric is favorable for Text2Live [3] and the method of Jamriska et al. [20], which explicitly utilize optical flow information. Since our method also uses the coarse depth structure guidance, we have included it in our evaluations.

4.1. Results

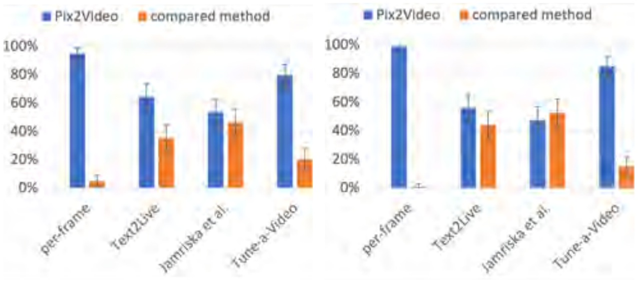
Qualitative results. We provide a set of example edits our method achieves in Figure 4. For each example, we show several randomly sampled frames both from the input and edited video, along with the edit prompts. As seen in the figure, Pix2Video can handle videos with a clear foreground object (e.g., bear) as well as multiple foreground objects (e.g., fish). We can perform *localized* edits where a prompt specifies the attribute of an object (e.g., swan) as well as *global* edits which change the overall style (e.g., kite surfer). Please note that, unlike Text2Live, we do *not* use any explicit mask information to specify which regions should be edited. This enables us to handle reflections automatically, as in the swan example. Since our method utilizes the depth information as a coarse structure guidance, it is more effective for video stylization tasks. Since the depth information is coarse, especially for the background, we observe that edits can also make structural changes in these regions. We refer to the supplementary material for more examples.

Table 2: **Quantitative comparison.** Our method attains the highest CLIP-Text score (faithfulness) and fairly good CLIP-Image and Pixel-MSE (temporal coherency).

	CLIP-Text ↑	CLIP-Image ↑	Pixel-MSE ↓
Jamriska et al. [20]	0.2684	0.9838	44.62
Text2Live [3]	0.2679	0.9806	72.57
Tune-a-Video [58]	0.2691	0.9674	1190.62
SDEdit [30]	0.2775	0.8731	2324.29
img2img, null inv. [31]	0.2802	0.8882	1261.60
prompt2prompt [15]	0.2618	0.9243	296.51
ours w/o update	0.2893	0.9740	371.18
ours (warped $\hat{x}_0^{i-1,t}$)	0.2892	0.9760	216.60
ours	0.2891	0.9767	228.62

Quantitative results and comparisons. We provide quantitative comparisons to the baseline methods in Table 2 and also refer to Figure 3. Among the baseline methods, we observe that Jamriska et al. [20] achieve good temporal coherency as it explicitly utilizes flow information to warp and propagate edits from the provided keyframes. However, as new content appears in the video or when flow information is unreliable, it fails to hallucinate details resulting in less faithful edits (Fig. 3 top left). Tex2Live [3] constructs a neural atlas for the input video, which by construction ensures temporal consistency by mapping edits in the atlas to each frame. It performs well when a clear foreground and background separation exists, and an accurate neural atlas can be synthesized. Since this method edits the atlas itself it is temporally coherent by construction. However, when there are multiple foreground objects, e.g., Fig. 3 top middle, a reliable neural atlas cannot be computed, and the method fails. The edited results have "ghost shadow" that consequently deteriorates CLIP-Text scores.

We also observe that, unlike our method, it is not straightforward to perform global style edits on *both* the foreground and background consistently with Text2Live. While attaining high CLIP-Text scores, i.e., generating frames faithful to the edit prompt, SDEdit [30] results in worst temporal coherency as it generates each frame independently. This is confirmed by the lowest CLIP-Image score and highest Pixel-MSE in Table 2. The concurrent Tune-a-Video method [58] achieves a nice balance between edit propagation and temporal coherency. However, subsampling a fixed number of frames in the video inevitably hurts the temporal scores. We also observe that for some edits it cannot preserve the structure of the objects in the input video (Fig. 3 top right). Some edits result in very similar outputs, which could be attributed to per-example finetuning that might cause overfitting (see the pig and fish examples in the project webpage). In contrast, by using the ad-



(a) Faithfulness (b) Preference

Figure 5: **User evaluation.** Our user study shows that Pix2Video not only better reflects the edits but is also perceptually preferred over other methods.

ditional depth conditioning, Pix2Video better preserves the structure of the input video and strikes a good balance between respecting the edit as well as keeping temporal consistency without requiring any training.

User study. We further evaluate Pix2Video against the baselines with a user study. Given 10 videos with 2 different edits each, we ask 37 participants to compare our result with one of the baselines shown in random order. Each edited video is ranked, by pairwise comparison, by 11 users on average. We ask two questions to the user: (i) Which video better represents the provided editing caption? (ii) Which edited video do you prefer? The first question evaluates "faithfulness" while the second indicates overall video quality via "preference". Please see supplemental matetrial for more details on our perceptual study.

We plot the results of user voting in Fig. 5, where the error bars represent 99% confidence intervals. In Fig. 5a, the majority of the participants agreed that our results reflect the edits more faithfully than others, in accordance with the higher CLIP-Text score in Table 2. Fig. 5b shows that our results are also preferred over other baselines when viewed side-by-side. Note that temporal smoothness plays a crucial role in the perceptual quality of a video. Despite Jamriska et al. [20] losing edits as pointed out in Fig. 3, it is on par with our method in terms of overall preference which we attribute to high temporal coherency (see CLIP-Image and Pixel-MSE in Table 2). In summary, the user study confirms that we achieve a good balance between ensuring edits and maintaining temporal consistency.

Comparison to controllable text based image editing methods. We are witnessing a growing interest in adapting large scale text to image generation methods for controllable image editing applications. Hence, we also make comparisons to such methods applied in a per-frame manner. Specifically, we compare to null text inversion [31] and Prompt-to-Prompt [15]. As show in Table 2, these methods are inferior especially with respect to temporal scores (CLIP-Image and Pixel-MSE). Prompt-to-Prompt supports localized edits by cross frame attention control and performs better in such cases as opposed to more global edits. However, even in such cases, in terms of temporal consistency it is not on par with our method (see Figure 6). We note that our method is orthogonal and can adapt the Prompt-to-Prompt framework as the underlying text-to-image model as shown by the concurrent work [27].

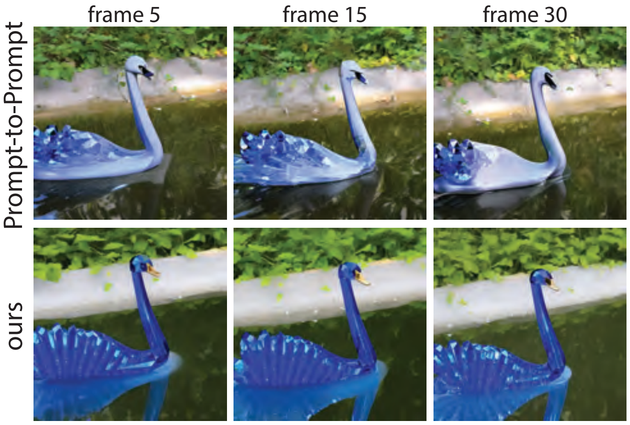


Figure 6: **Comparison to Prompt-to-Prompt.** While Prompt-to-Prompt can achieve localized edits based on the original and target prompts `a black swan on the lake` → `a Swarovski blue crystal swan on the lake` respectively, applying it on a per-frame basis results in more temporal flickering compared to our method.

Ablations. We perform ablation studies to validate several design choices. First, we evaluate different choices of previous frames to use for self attention feature injection. In Figure 8, we compare scenarios where we always attend to (i) a fixed anchor frame (first frame in our experiments), (ii) the previous frame only, (iii) an anchor frame and a randomly selected previous frame, and (iv) an anchor frame and a previous frame as in our method. In cases where no previous frame information is used or a random previous frame is chosen, we observe artifacts, especially for sequences that contain more rotational motion, e.g., structure of the car not being preserved as the car rotates. This confirms our intuition that attending to the previous frame implicitly represents the state of the edit in a recurrent manner. Without an anchor frame, we observe more temporal

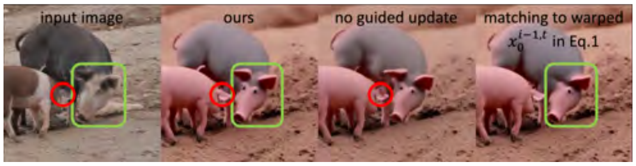


Figure 7: **Visual comparison of different guided latent update strategies.** See text for more discussion.

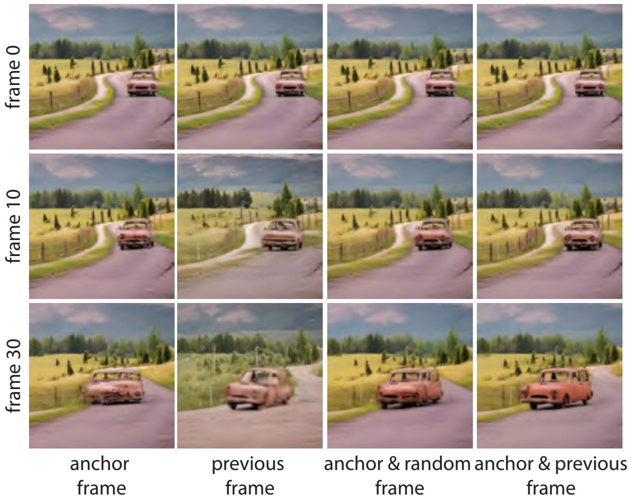


Figure 8: **Different self-attention feature injection schemes.** Using a fixed anchor frame results in structural artifacts as the distance between the anchor and the edited frame increases. Attending only to the previous frame or a randomly selected previous frame results in temporal and structural artifacts. We obtain the best results by using a fixed anchor and the previous frame.

flickering and the edit diminishes as the video progresses. By combining the previous frame with an anchor frame we strike a good balance.

Pix2Video consists of two main steps of feature injection and guided latent update. In Table 2, we report the results without the latent update step. As shown in the metrics, this results in worse CLIP-Image scores and Pixel-MSE errors confirming that this guidance is effective in enforcing temporal coherency and preserving the edit. We also experimented with warping $\hat{x}_0^{i-1,t}$ in Equation 1 to the current frame using optical flow (RAFT). Table 2 shows that this leads to on-par results. We hypothesize that this is due to the low resolution of the latent images. We expect the benefit of warping targets to more pronounced when operating at higher resolution. Fig. 7 provides visual comparison with the two aforementioned baselines. We refer to the supplementary material for more ablation studies.

Implementation details. We use the publicly available Stable Diffusion depth conditioned model [1] as our image generation model. We obtain depth estimates for the input videos using the MIDAS depth estimator [40]. For each video, we perform the inversion once save it. Each editing operation then amounts to generating each frame given the target prompt. We use 50 DDIM steps both for inversion and editing. We have experimented with per-frame null-text inverted noise [31] and observed similar quality. Hence, we decided to use DDIM inversion due to efficiency (2.5 seconds vs 40-60 seconds per frame). We generate results at 512 × 512 resolution. The temporal error Pixel-MSE in Table 2 is therefore computed on a 512 × 512 image domain. Our method does not incur any additional significant cost on the image inference step. Without any optimization and frame-by-frame processing, we invert and edit each frame in 2.5 and 5 seconds respectively using a batch size of 1 on an A100 GPU. When implemented with batching and AITemplate framework*, the time complexity reduces to ∼0.5 seconds/frame for inversion and 1 second/frame for editing.

5. Conclusion and Future Work

We present Pix2Video, a method that utilizes a pretrained and fixed text-to-image generation model for editing video clips. We demonstrate the power of our method on various inputs and editing tasks. We provide detailed comparisons to baselines along with an extensive user study. We show that Pix2Video is on par or superior to baselines while not requiring additional pre-processing or finetuning.

However, our method also has limitations that we would like to address in future work. We believe that there is still room for improvement in terms of temporal coherency. Exploiting other energy terms, e.g., patch-based similarity [20] and CLIP similarity, during the latent update stage, is a promising direction. As we utilize an anchor frame for feature injection, handling longer videos where the distance from the anchor increases can cause quality degradation. Additional conditioning (e.g., image embedding conditioning [6]) and a smart anchor update mechanism is a potential direction. Our method aims to make the diffusion network more temporally aware. It is a promising future direction to empower the decoder (or the upsampler) to more faithfully capture high frequency details and further enforce consistency. We use a per-frame depth prediction method which lacks full temporal coherency. Our method can benefit from advances in temporally coherent depth estimation. We also believe that how strongly the diffusion model adheres to the input depth conditioning is worth investigating.

Finally, given that our method does not require any finetuning, it has the advantage of being applied to parallel efforts that aim to introduce additional control to the image generation model, which we would like to exploit.

Acknowledgments.

We would like to thank the users who have participated in our user study. We also thank Jakub Fiser for helping to run the comparisons to the work *Stylizing Video by Example* [20].

References

[1] *Stable Diffusion v2*, 2022. https://huggingface.co/stabilityai/stable-diffusion-2-depth.

*https://github.com/facebookincubator/AITemplate

3, 4, 7, 9

[2] 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 an ensemble of expert denoisers. *arXiv preprint arXiv:2211.01324*, 2022. 2, 3

[3] 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 (ECCV)*, pages 707–723. Springer, 2022. 2, 3, 5, 7

[4] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale GAN training for high fidelity natural image synthesis. In *International Conference on Learning Representations (ICLR)*, 2019. 2

[5] 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. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2022. 2

[6] Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. *arXiv preprint arXiv:2211.09800*, 2022. 9

[7] Jooyoung Choi, Sungwon Kim, Yonghyun Jeong, Youngjune Gwon, and Sungroh Yoon. ILVR: Conditioning method for denoising diffusion probabilistic models. In *International Conference on Computer Vision (ICCV)*, pages 14367–14376, October 2021. 2

[8] Florinel-Alin Croitoru, Vlad Hondru, Radu Tudor Ionescu, and Mubarak Shah. Diffusion models in vision: A survey. *arXiv preprint arXiv:2209.04747*, 2022. 1, 2

[9] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Conference on Neural Information Processing Systems (NeurIPS)*, volume 34, pages 8780–8794, 2021. 2

[10] Patrick Esser, Johnathan Chiu, Parmida Atighehchian, Jonathan Granskog, and Anastasis Germanidis. Structure and content-guided video synthesis with diffusion models. *arXiv preprint arXiv:2302.03011*, 2023. 3, 5

[11] Oran Gafni, Adam Polyak, Oron Ashual, Shelly Sheynin, Devi Parikh, and Yaniv Taigman. Make-A-Scene: Scene-based text-to-image generation with human priors. *arXiv*, 2022. 3

[12] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Conference on Neural Information Processing Systems (NeurIPS)*, volume 27, 2014. 2

[13] Sonam Gupta, Arti Keshari, and Sukhendu Das. RV-GAN: Recurrent gan for unconditional video generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 2024–2033, June 2022. 2

[14] Yingqing He, Tianyu Yang, Yong Zhang, Ying Shan, and Qifeng Chen. Latent video diffusion models for high-fidelity video generation with arbitrary lengths. *arXiv preprint arXiv:2211.13221*, 2022. 3, 4

[15] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. In *International Conference on Learning Representations (ICLR)*, 2023. 1, 2, 7, 8

[16] Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. Clipscore: A reference-free evaluation metric for image captioning. *arXiv preprint arXiv:2104.08718*, 2021. 7

[17] Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P. Kingma, Ben Poole, Mohammad Norouzi, David J. Fleet, and Tim Salimans. Imagen video: High definition video generation with diffusion models. *arXiv*, 2022. 3, 7

[18] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Conference on Neural Information Processing Systems (NeurIPS)*, volume 33, pages 6840–6851, 2020. 1, 2

[19] 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. 3

[20] Ondřej Jamriška, Šárka Sochorová, Ondřej Texler, Michal Lukáč, Jakub Fišer, Jingwan Lu, Eli Shechtman, and Daniel Sýkora. Stylizing video by example. *ACM Transactions on Graphics*, 38(4), 2019. 1, 2, 3, 5, 7, 8, 9

[21] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2022. 2

[22] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Computer Vision and Pattern Recognition (CVPR)*, pages 4401–4410, 2019. 2

[23] 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. 3, 5

[24] Bahjat Kawar, Shiran Zada, Oran Lang, Omer Tov, Huiwen Chang, Tali Dekel, Inbar Mosseri, and Michal Irani. Imagic: Text-based real image editing with diffusion models. *arXiv preprint arXiv:2210.09276*, 2022. 2

[25] Mingi Kwon, Jaeseok Jeong, and Youngjung Uh. Diffusion models already have a semantic latent space. In *International Conference on Learning Representations (ICLR)*, 2023. 5

[26] 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. 5

[27] Shaoteng Liu, Yuechen Zhang, Wenbo Li, Zhe Lin, and Jiaya Jia. Video-p2p: Video editing with cross-attention control. *arXiv:2303.04761*, 2023. 8

[28] Sebastian Loeschcke, Serge Belongie, and Sagie Benaim. Text-driven stylization of video objects. In *ECCV 2022 Workshops*, 2022. 3

[29] Erika Lu, Forrester Cole, Tali Dekel, Weidi Xie, Andrew Zisserman, David Salesin, William T Freeman, and Michael Rubinstein. Layered neural rendering for retiming people in video. In *SIGGRAPH Asia*, 2020. 3

[30] Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. SDEdit: Guided

image synthesis and editing with stochastic differential equations. In *International Conference on Learning Representations (ICLR)*, 2021. 2, 7

[31] Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Null-text inversion for editing real images using guided diffusion models. *arXiv preprint arXiv:2211.09794*, 2022. 1, 5, 7, 8, 9

[32] Eyal Molad, Eliahu Horwitz, Dani Valevski, Alex Rav Acha, Yossi Matias, Yael Pritch, Yaniv Leviathan, and Yedid Hoshen. Dreamix: Video diffusion models are general video editors. *arXiv*, 2023. 1, 3

[33] Chong Mou, Xintao Wang, Liangbin Xie, Jian Zhang, Zhongang Qi, Ying Shan, and Xiaohu Qie. T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models. *arXiv*, 2023. 2

[34] 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 *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162, pages 16784–16804, 2022. 2, 4

[35] Dong Huk Park, Samaneh Azadi, Xihui Liu, Trevor Darrell, and Anna Rohrbach. Benchmark for compositional text-to-image synthesis. In *Neural Information Processing Systems Datasets and Benchmarks Track*, 2021. 7

[36] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. Van Gool, M. Gross, and A. Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *Computer Vision and Pattern Recognition*, 2016. 5

[37] Haonan Qiu, Yuming Jiang, Hang Zhou, Wayne Wu, and Ziwei Liu. Stylefacev: Face video generation via decomposing and recomposing pretrained stylegan3. *arXiv*, 2022. 2

[38] 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 (ICML)*, pages 8748–8763, 2021. 7

[39] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022. 2

[40] René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 44(3), 2022. 3, 9

[41] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Computer Vision and Pattern Recognition (CVPR)*, pages 10684–10695, 2022. 1, 2

[42] Manuel Ruder, Alexey Dosovitskiy, and Thomas Brox. Artistic style transfer for videos. In *German Conference on Pattern Recognition*, pages 26–36, 2016. 3

[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. In *International Conference on Computer Graphics and Interactive Techniques (SIGGRAPH)*, 2022. 2

[44] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Raphael Gontijo-Lopes, Burcu Karagol Ayan, Tim Salimans, Jonathan Ho, David J. Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2022. 2

[45] Masaki Saito, Eiichi Matsumoto, and Shunta Saito. Temporal generative adversarial nets with singular value clipping. In *ICCV*, 2017. 2

[46] Uriel Singer, Adam Polyak, Thomas Hayes, Xi 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. In *International Conference on Learning Representations (ICLR)*, 2023. 1, 3

[47] Ivan Skorokhodov, Sergey Tulyakov, and Mohamed Elhoseiny. Stylegan-v: A continuous video generator with the price, image quality and perks of stylegan2. In *Computer Vision and Pattern Recognition (CVPR)*, pages 3626–3636, 2022. 2

[48] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations (ICLR)*, 2021. 1, 2, 5

[49] Zachary Teed and Jia Deng. RAFT: Recurrent all-pairs field transforms for optical flow. In *European Conference on Computer Vision (ECCV)*, 2020. 7

[50] Ondřej Texler, David Futschik, Michal Kučera, Ondřej Jamriška, Šárka Sochorová, Meglei Chai, Sergey Tulyakov, and Daniel Sýkora. Interactive video stylization using few-shot patch-based training. *ACM Transactions on Graphics*, 39(4):73, 2020. 3

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

[52] Narek Tumanyan, Michal Geyer, Shai Bagon, and Tali Dekel. Plug-and-play diffusion features for text-driven image-to-image translation. *arXiv preprint arXiv:2211.12572*, 2022. 2, 4

[53] 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 descriptions. In *International Conference on Learning Representations (ICLR)*, 2023. 3

[54] Andrey Voynov, Kfir Abernan, and Daniel Cohen-Or. Sketch-guided text-to-image diffusion models. *arXiv preprint arXiv:2211.13752*, 2022. 2, 3

[55] Tengfei Wang, Ting Zhang, Bo Zhang, Hao Ouyang, Dong Chen, Qifeng Chen, and Fang Wen. Pretraining is all you need for image-to-image translation. *arXiv*, 2022. 2

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

[57] 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

[58] 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, 3, 5, 7

[59] Yiran Xu, Badour AlBahar, and Jia-Bin Huang. Temporally consistent semantic video editing. *arXiv preprint arXiv: 2206.10590*, 2022. 2

[60] Lijun Yu, Yong Cheng, Kihyuk Sohn, José Lezama, Han Zhang, Huiwen Chang, Alexander G. Hauptmann, Ming-Hsuan Yang, Yuan Hao, Irfan Essa, and Lu Jiang. MAGVIT: Masked generative video transformer. *arXiv*, 2022. 3

[61] Sihyun Yu, Kihyuk Sohn, Subin Kim, and Jinwoo Shin. Video probabilistic diffusion models in projected latent space. In *Computer Vision and Pattern Recognition (CVPR)*, 2023. 1, 3

[62] Sihyun Yu, Jihoon Tack, Sangwoo Mo, Hyunsu Kim, Junho Kim, Jung-Woo Ha, and Jinwoo Shin. Generating videos with dynamics-aware implicit generative adversarial networks. In *International Conference on Learning Representations (ICLR)*, 2022. 2

[63] Lvmin Zhang and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. *arXiv preprint arXiv:2302.05543*, 2023. 2

[64] Qihang Zhang, Ceyuan Yang, Yujun Shen, Yinghao Xu, and Bolei Zhou. Towards smooth video composition. *International Conference on Learning Representations (ICLR)*, 2023. 2