Shape-aware Text-driven Layered Video Editing

Yao-Chih Lee Ji-Ze Genevieve Jang Yi-Ting Chen Elizabeth Qiu Jia-Bin Huang
University of Maryland, College Park

https://text-video-edit.github.io

Figure 1. **Shape-aware consistent video editing.** Our method enables consistent text-guided video editing with both *appearance* and *shape* changes. The top row shows the input frames. The second and third rows present editing results from two text prompts: "`running sports car`" and "`running minivan`", respectively. Note that text-driven editing involves *both* texture and structure editing on the foreground object. Our method performs consistent edits on sequential frames while preserving the object motion in the input video.

Abstract

Temporal consistency is essential for video editing applications. Existing work on layered representation of videos allows propagating edits consistently to each frame. These methods, however, can only edit object appearance *rather than object* shape *changes due to the limitation of using a fixed UV mapping field for texture atlas. We present a shape-aware, text-driven video editing method to tackle this challenge. To handle shape changes in video editing, we first propagate the deformation field between the input and edited keyframe to all frames. We then leverage a pre-trained text-conditioned diffusion model as guidance for refining shape distortion and completing unseen regions. The experimental results demonstrate that our method can achieve shape-aware consistent video editing and compare favorably with the state-of-the-art.*

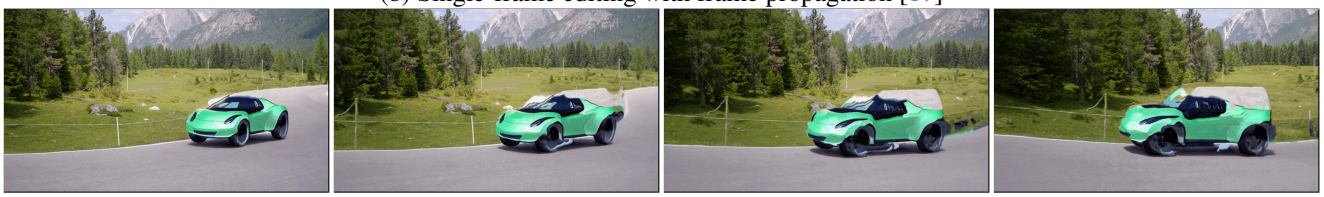
1. Introduction

Image editing. Recently, image editing [19, 20, 24, 34, 40, 44] has made tremendous progress, especially those using diffusion models [19, 20, 40, 44]. With free-form text prompts, users can obtain photo-realistic edited images without artistic skills or labor-intensive editing. However, unlike image editing, video editing is more challenging due to the requirement of temporal consistency. Independently editing individual frames leads to undesired inconsistent frames, as shown in Fig. 2a. A naïve way to deal with temporal consistency in video editing is to edit a single frame and then propagate the change to all the other frames. Nevertheless, artifacts are presented when there are unseen pixels from the edited frame in the other frames, as shown in Fig. 2b.

Video editing and their limitations. For consistent video editing, *Neural Layered Atlas* (NLA) [18] decomposes a video into unified appearance layers *atlas*. The layered decomposition helps consistently propagate the user edit to

(a) Multi-frame editing with frame interpolation [42]

(b) Single-frame editing with frame propagation [17]



(c) Text2LIVE [2] with prompt "`sports car`"

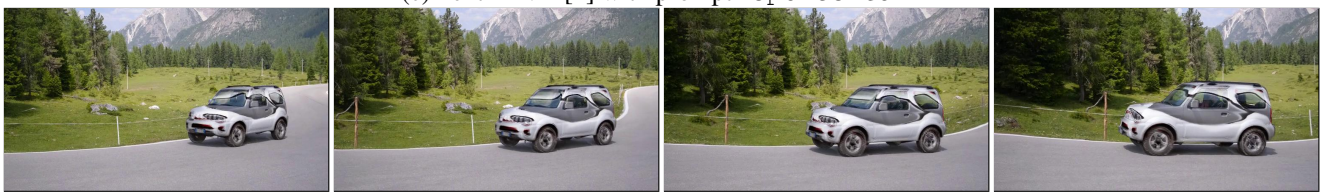


Figure 2. **Limitation of existing work.** Compare these results from baseline methods with our "`sports car`" result in Fig. 1. (a) Multiple frames are edited *independently* and interpolated by frame interpolation method [42]. Such an approach shows realistic per-frame results but suffers from temporal flickering. (b) Extracting a single keyframe for image editing, the edits are propagated to each frame via [17]. The propagated edits are temporally stable. However, it yields visible distortions due to the unseen pixels from the keyframe. (c) The SOTA Text2LIVE [2] results demonstrate temporally-consistent appearance editing but remain the source shape "`Jeep`" instead of the target prompt "`sports car`" by using the fixed UV mapping of NLA.

individual frames with per-frame UV sampling association. Based on NLA, Text2LIVE [2] performs text-driven editing on atlases with the guidance of the Vision-Language model, CLIP [39]. Although Text2LIVE [2] makes video editing easier with a text prompt, it can only achieve *appearance manipulation* due to the use of fixed-shape associated UV sampling. Since per-frame UV sampling gathers information on motion and shape transformation in each frame to learn the pixel mapping from the atlas, shape editing is not feasible, as shown in Fig. 2c.

Our work. In this paper, we propose a *shape-aware* text-guided video editing approach. The core idea in our work lies in a novel UV map deformation formulation. With a selected keyframe and target text prompt, we first generate an edited frame by image-based editing tool (*e.g.*, Stable Diffusion [44]). We then perform pixel-wise alignment between the input and edited keyframe pair through a semantic correspondence method [51]. The correspondence specifies the deformation between the input-edited pair at the keyframe. According to the correspondence, the shape and appearance change can then be mapped back to the atlas space. We can thus obtain per-frame deformation by sampling the deformation from the atlas to the original UV maps. While this method helps with shape-aware editing, it is insufficient due to unseen pixels in the edited keyframe. We tackle this by further optimizing the atlas texture and the deformation using a pretrained diffusion model by adopting the gradient update procedure described in DreamFusion [38]. Through the atlas optimization, we achieve consistent *shape* and *appearance* editing, even in challenging cases where the moving object undergoes 3D transformation (Fig. 1).

Our contributions.

- We extend the capability of existing video editing methods to enable shape-aware editing.
- We present a deformation formulation for frame-dependent shape deformation to handle target shape edits.
- We demonstrate the use of a pre-trained diffusion model for guiding atlas completion in layered video representation.

2. Related Work

Text-driven image synthesis and editing. Recent years have witnessed impressive progress in text-guided image synthesis and manipulation using GANs [24, 25, 27, 41, 43, 55, 56, 61]. On text-to-image *generation*, DALL-E [41] first demonstrates the benefits of training text-to-image models using a massive image-text dataset. Most recent text-to-image generators [6, 30] use a pre-trained CLIP [39] as the

guidance. On text-to-image *manipulation/editing*, recent methods also take advantage of the pretrained CLIP embedding for text-driven editing [9,36,58]. These methods either pretrain the model with CLIP embedding as inputs or use a test-time optimization approach [2,8,21].

Recently, diffusion models [7, 14, 50] have shown remarkable success in both text-guided image generation [1, 35, 44–46] and editing [12, 35, 44] tasks. Stable Diffusion [44] performs a denoising diffusion process in a latent space and achieves high-resolution text-to-image generation and image-to-image translation results. In particular, the release of the model trained on large-scale text-image pair dataset [47] facilitates various creative applications from artists and practitioners in the community. Our work leverages the state-of-the-art text-to-image model, Stable Diffusion [44], and extends its semantic image editing capability to consistent video editing.

Video generation. Building upon the success of photorealistic (text-driven) image generation, recent work has shown impressive results on video generation, with a focus on generating long video [5, 11, 49, 60] and videos from free-form text prompts [13, 48, 52]. Unlike video *generation* methods, our work differs in that we perform text-driven video *editing* for real videos.

Video editing. In contrast to the breakthrough of image editing, video editing methods are faced with two core challenges: 1) temporal consistency and 2) computational complexity of the additional dimension. To attain temporally consistent editing effects, EbSynth [17] utilizes keyframes and propagates the edits to the entire video with optical flows computed from consecutive frames. Such flow-based techniques have been applied in other tasks such as video synthesis [3], video completion [10,15,26], and blind video consistency [22, 23]. Several studies address temporal inconsistency in the latent space via GAN inversion [29,54,57]. However, current GAN-based models can only model datasets with limited diversity (e.g., portrait or animal faces). Another line of approaches [18, 28, 32, 33, 59] decomposes a video into unified layer representation for consistent editing. Neural Layered Atlas (NLA) [18] performs test-time optimization on a given input video to learn the canonical appearance layer and per-frame UV mapping using video reconstruction loss. With layer decomposition, one can use text-driven image editing techniques to the unified layers to consistently broadcast the edits to each frame. The work most relevant to ours is Text2LIVE [2] and Loeschcke *et al*. [31]. Both methods build upon NLA to perform text-driven editing on the learned atlases. A pre-trained CLIP is used for each input video to guide the atlas editing via a test-time optimization framework. Yet, limited by the formulation of NLA, they only allow *appearance* edits due to the fixed UV mapping from the atlas to frames. The mapping fields store the original shape information in each frame so that the fixed UV mapping restricts the freedom of *shape editing* in [2, 18, 31]. Our work also builds upon NLA for achieving temporally consistent video editing. In contrast to existing methods [2, 31], we extend the capability of text-driven editing to enable shape editing.

3. Method

Given an input video $\mathcal{I}^s_{1..N}$ and a text prompt, our proposed shape-aware video editing method produces a video $\mathcal{I}^t_{1..N}$ with appearance *and* shape changes while preserving the motion in the input video. For maintaining temporal consistency, our method uses the pre-trained video decomposition method, NLA [18], to acquire the canonical atlas layer $\mathcal{I}^s_A$ and the associated per-frame UV map $\mathcal{W}^s_{A\to 1..N}$ per motion group. For simplicity, we assume a single moving object in an input video so that there are two atlases $\mathcal{I}^{s,FG}_A$ and $\mathcal{I}^{s,BG}_A$ for foreground and background contents, respectively. The edits in $\mathcal{I}^{s,FG}_A$ can be consistently transferred to each frame with UV mapping. To render the image $\mathcal{I}^s_j$ back, we use the $\mathcal{W}^s_{A\to t}$ and an alpha map α^s_t to sample and blend:

$$\begin{aligned} \mathcal{I}^s_j &= \mathcal{I}^{s,FG}_j * \alpha^s_j + \mathcal{I}^{s,BG}_j * (1-\alpha^s_j), \\ \mathcal{I}^{s,g}_j &= \mathcal{W}^{s,g}_{A\to j} \otimes \mathcal{I}^{s,g}_A, g \in \{FG, BG\}, \end{aligned} \tag{1}$$

where $\otimes$ denotes the warping operation. Following our shape deformation introduction, we focus on the foreground atlas and will omit FG from $\mathcal{I}^{s,FG}$ for simplicity.

We first select a single source keyframe $\mathcal{I}^s_k$ to pass into a text-driven image editing tool (*e.g.*, Stable Diffusion [44]). The edits in target $\mathcal{I}^t_k$ will then be propagated to $\mathcal{I}^t_{1..N}$ through the atlas space with the mapping of $\mathcal{W}^s_{A\to 1..N}$. Yet, the UV mapping cannot work when the edits involve *shape changes* since $\mathcal{W}^s_{A\to 1..N}$ are specifically for reconstructing the original shapes in the input video. Hence, to associate the target shape correctly, we propose a UV deformation formulation (Sec. 3.2) to transform each $\mathcal{W}^s_{A\to j}$ into $\mathcal{W}^t_{A\to j}$ according to the deformation between $(\mathcal{I}^s_k, \mathcal{I}^t_k)$. In other words, the keyframe deformation $\mathcal{D}^{s\to t}_k$ between $(\mathcal{I}^s_k, \mathcal{I}^t_k)$ serves as the *bridge* between input and output videos for changing into the edited target shape while preserving the source motion in the input. Note that the edits and keyframe deformation $\mathcal{D}^{s\to t}_k$ alone are insufficient due to some unobserved areas from the viewpoint of image $\mathcal{I}^s_k$. Therefore, to acquire a complete and consistent editing result, we leverage a pre-trained diffusion model to optimize the editing appearance and deformation parameters in the atlas space in Sec. 3.3. The process produces the final edited video $\mathcal{I}^t_{1..N}$ with desired object shape and appearance changes.

3.1. Keyframe editing

With the given text prompt, we edit a representative keyframe $\mathcal{I}^s_k$ (*e.g.*, the middle frame of the video) by a

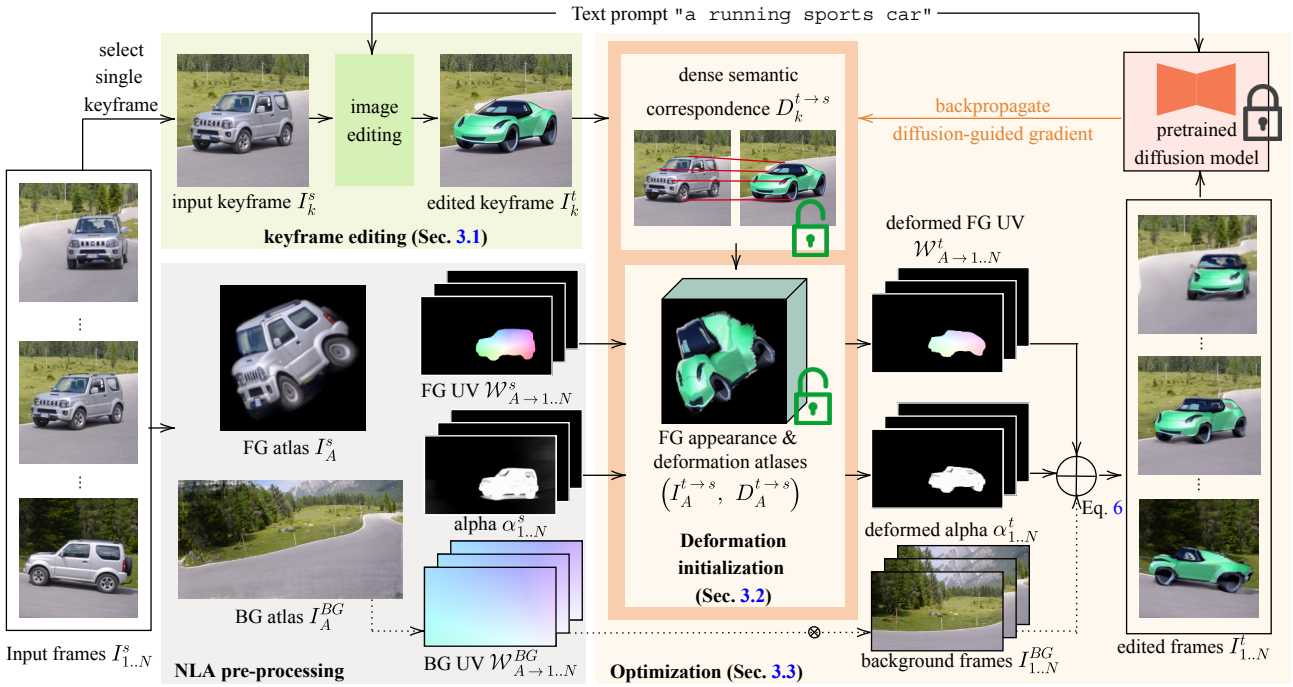


Figure 3. **Method overview.** Given an input video and a target edit text prompt, our method first bases on a pre-trained NLA [18] to decompose the video into unified atlases with the associated per-frame UV mapping. Aside from video decomposition, we use the text-to-image diffusion model to manipulate a single keyframe in the video (Sec. 3.1). Subsequently, we estimate the dense semantic correspondence between the input and edited keyframes for shape deformation. The shape deformation of the keyframe serves as the *bridge* between input and output videos for per-frame deformation through the UV mapping and atlas. Our deformation module (Sec. 3.2) transforms the UV map with the semantic correspondence to associate with the edits for each frame. To address the issues of unseen pixels from the single keyframe, we optimize the edited atlas and the deformation parameters guided by a pre-trained diffusion model with the input prompt (Sec. 3.3).

pre-trained Stable Diffusion [44] to obtain target edited keyframe $\mathcal{I}^t_k$. Afterward, we leverage a pre-trained semantic correspondence model [51] to associate the correspondence between two different objects. The pixel-level semantic correspondence is the deformation that transforms the target shape in $\mathcal{I}^t_k$ to the source shape in $\mathcal{I}^s_k$.

3.2. Deformation formulation

With the estimated semantic correspondence, we can obtain the pixel-level *shape deformation vectors*, $\mathcal{D}^{t\to s}_k \in \mathbb{R}^{H\times W\times 2}$. The target shape in $\mathcal{I}^t_k$ are then deformed into the source shapes in $\mathcal{I}^s_k$ via $\mathcal{D}^{t\to s}_k$:

$$\mathcal{I}^{t\to s}_k = \mathcal{D}^{t\to s}_k \otimes \mathcal{I}^t_k. \tag{2}$$

With the aid of $\mathcal{D}^{t\to s}_k$, the edited object can be back-projected to the atlas to form an edited atlas, $\mathcal{I}^{t\to s}_A$, by $\mathcal{W}^s_{k\to A}$. Since it maintains the original shape, we cannot directly map the edited $\mathcal{I}^t_k$ to the atlas with $\mathcal{W}^s_{k\to A}$.

Given the edited atlas $\mathcal{I}^{t\to s}_A$, the appearance edits can already be propagated to each frame with $\mathcal{W}^s_{A\to 1..N}$ in source shapes. However, this needs improvement since our goal is to generate a new video with the target shape. In addition to propagating the edited appearance via the atlas space, we spread the displacement vectors to each frame to obtain per-frame deformation by back projecting keyframe deformation $\mathcal{D}^{t\to s}_k$ into atlas space A with $\mathcal{W}^s_{k\to A}$ to get $\mathcal{D}^{t\to s}_A$. Yet, simply *warping* into the new image space is insufficient as the coordinate system also got transformed by the warping operation. Therefore, we formulate a *shape deformation vector transformation* matrix, $\boldsymbol{M}_{\mathcal{W}}$, to handle the deformation vectors w.r.t. the original coordinate system by a warp field $\mathcal{W}$:

$$\mathcal{D}'(x',y')^T = \boldsymbol{M}_{\mathcal{W}} D(x,y)^T, \tag{3}$$

where (x,y) and (x',y') represent the corresponding pixels in the source and target images, respectively, by the warping field, $\mathcal{W}$ (*i.e.*, $(x',y') = \mathcal{W}(x,y)$). For pixel-level deformation, we compute a per-pixel deformation vector $\boldsymbol{M}_{\mathcal{W}}$ for each pixel (x,y) by:

$$\boldsymbol{M}_{\mathcal{W}} = \begin{bmatrix} \mathcal{W}(x+\Delta x,y) - \mathcal{W}(x,y) \\ \mathcal{W}(x,y+\Delta y) - \mathcal{W}(x,y) \end{bmatrix}^T \begin{bmatrix} 1/\Delta x \\ 1/\Delta y \end{bmatrix}, \tag{4}$$

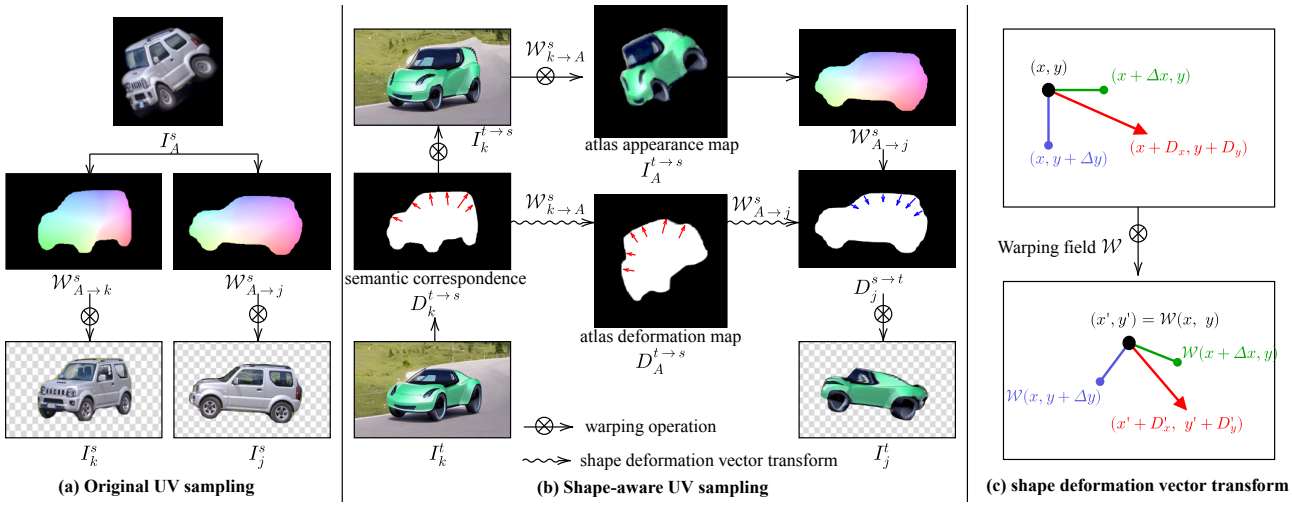


Figure 4. **Deformation formulation.** Given the semantic correspondence between the input and edited keyframes, we map the edits back to the atlas via the original UV map (in the shape of the original atlas). Meanwhile, we transform the per-pixel deformation vectors into the atlas space with the same UV mapping field by (c). Consequently, the UV map samples the color and the deformation vectors onto each frame to deform the original UV map respecting the edited shape.

where Δx and Δy denote small scalar shifts to form the local coordinate system in the source space. In practice, to avoid discrete sampling of warping, we use thin-plate spline [4] to approximate the warping field smoothly. We illustrate the transformation of the shape deformation vector in Fig. 4c. With the transformation for the vector, we can obtain the corresponding deformation in the target warped space with the warp function $\mathcal{W}$, which is the UV map in the atlas framework. Thus, the deformation map $\mathcal{D}_k^{t\to s}$ is propagated to each I_j^t by:

$$\begin{aligned}\mathcal{D}_A^{t\to s} &= \boldsymbol{M}_{\mathcal{W}_{k\to A}^s} \star (\mathcal{W}_{k\to A}^s \otimes \mathcal{D}_k^{t\to s}) \\ \mathcal{D}_j^{t\to s} &= \boldsymbol{M}_{\mathcal{W}_{A\to j}^s} \star (\mathcal{W}_{A\to j}^s \otimes \mathcal{D}_A^{t\to s}),\end{aligned} \tag{5}$$

where $\star$ denotes the per-pixel matrix multiplication for the deformation map. Hence, we can deform the UV map $\mathcal{W}_{A\to j}^s$ into $\mathcal{W}_{A\to j}^t$ by $\mathcal{W}_{A\to j}^t = \mathcal{D}_j^{s\to t} \otimes \mathcal{W}_{A\to j}^s$. Note that the alpha map for blending the target-shape object is also deformed in the same manner by $\alpha_j^t = \mathcal{D}_j^{s\to t} \otimes \alpha_j^s$. Finally, the edited $\mathcal{I}_j^t$ with initial deformation on the foreground object can be obtained by:

$$\mathcal{I}_j^t = \mathcal{W}_{A\to j}^t \otimes \mathcal{I}_A^{t\to s} * \alpha_j^t + \mathcal{I}_A^{BG} * (1-\alpha_j^t). \tag{6}$$

3.3. Atlas optimization

Through the deformation formulation in Sec. 3.2, we can already obtain an edited video with the corresponding shape changes if the semantic correspondence, *i.e.*, $D_k^{t\to s}$, is reliable. However, the estimated semantic correspondence is often inaccurate for shape deformation. As a result, it would yield distortions in some frames. Moreover, the edited atlas could be incomplete since it only acquires the editing pixels from the single edited keyframe so the unseen pixels from the keyframe are missing. Hence, these incomplete pixels produce visible artifacts in other frames.

To address these issues, we utilize an additional atlas network F_{θ_A} and semantic correspondence network $F_{\theta_{SC}}$ to fill the unseen pixels and refine the noisy semantic correspondence via an optimization. Here, the atlas network F_{θ_A} takes the initial appearance and deformation of the foreground atlas $(I_A^{t\to s}, D_A^{t\to s})$ as input and outputs the *refined* $(\tilde{\mathcal{I}}_A^{t\to s}, \tilde{\mathcal{D}}_A^{t\to s})$. Similarly, the semantic correspondence $\mathcal{D}_k^{t\to s}$ is approximated by a thin-plate spline. We feed the control points into the semantic correspondence network $F_{\theta_{SC}}$ to obtain the refined $\tilde{\mathcal{D}}_k^{t\to s}$.

We select several frames that capture different viewpoints for optimization. Our training of synthesizing the edited frames, $\mathcal{I}^t$, is guided by a pre-trained Vision-Language model with the target prompt. Inspired by DreamFusion [38], we leverage a pre-trained diffusion model [44] to provide pixel-level guidance by backpropagating the gradient of noise residual to the generated images (*without* backpropagating through the U-Net model). Adding a noise ε on $\mathcal{I}^t$ as the input, the pretrained diffusion UNet outputs a predicted noise $\hat{\varepsilon}$. The gradient of the noise residual $\hat{\varepsilon} - \varepsilon$ is backpropagated to update θ:

$$\nabla_\theta \mathcal{L}_{diff}(\mathcal{I}^t) \triangleq \mathbb{E}_{i,\varepsilon}[w(i)(\hat{\varepsilon}-\varepsilon)\frac{\partial \mathcal{I}^t}{\partial \theta}], \tag{7}$$

where i stands for the time step for the diffusion model and the parameter set $\theta = \{\theta_A, \theta_{SC}\}$. We update the unified information in the atlas space to maintain the temporal consistency of the editing appearance and deformation with only training on a few generated frames $\mathcal{I}^t$.

In addition to the guidance of the diffusion model on

multiple frames, we also apply several constraints to the learning of the refinement networks, F_{θ_A} and $F_{\theta_{SC}}$, to preserve the editing effects as in the target edited keyframe I_k^t. To ensure that the deformation through the atlas can successfully reconstruct the original edited $\mathcal{I}_k^t$, the keyframe loss, $\mathcal{L}_k$, measures the error between the original $\mathcal{I}_k^t$ and the reconstructed $\tilde{\mathcal{I}}_k^t$ by L1 loss:

$$\mathcal{L}_k = |\tilde{\mathcal{I}}_k^t - \mathcal{I}_k^t|. \tag{8}$$

Besides, we also apply a total variation loss to encourage the spatial smoothness of the refined appearance in the atlas. The atlas loss is as follows:

$$\mathcal{L}_A = \mathcal{L}_{tv}(\tilde{\mathcal{I}}_A^{t\to s}). \tag{9}$$

During the optimization, we also refine the semantic correspondence $\tilde{\mathcal{D}}_k^{t\to s}$ of the keyframe pair. An ideal semantic correspondence matches semantically-similar pixels and perfectly transforms the target shape into the source shape. Therefore, we compute the errors of the deformed target and the source object masks, $\mathcal{M}_k^t$ and $\mathcal{M}_k^s$:

$$\mathcal{L}_{SC} = |(\tilde{\mathcal{D}}_k^{t\to s} \otimes \mathcal{M}_k^t) - \mathcal{M}_k^s|. \tag{10}$$

The total loss function $\mathcal{L} = \mathcal{L}_{diff} + \lambda_k\mathcal{L}_k + \lambda_A\mathcal{L}_A + \lambda_{SC}\mathcal{L}_{SC}$, $\lambda_k, \lambda_A, \lambda_{SC} = 10^6, 10^3, 10^3$. The optimized parameters θ^* are then used to generate the final edited video $\mathcal{I}_{1..N}^{t*}$.

Implementation details.

We implement our method in PyTorch. We follow the video configuration in NLA with the resolution of 768×432. We use a thin-plate spline to inverse a warping field to prevent introducing holes by forward warping. The refinement networks, F_{θ_A} and $F_{\theta_{SC}}$ exploits the architecture of Text2LIVE [2] and TPS-STN [16], respectively. The optimization performs on 3 to 5 selected frames, including I_1^t, I_k^t, and I_N^t, for 600 to 1000 iterations. The optimization process takes 20 mins on a 24GB A5000 GPU. We further utilize an off-the-shelf super-resolution model [53] to obtain sharp details in the final edited atlases.

4. Experimental Results

Here we show sample editing results in the paper. We include additional video results in the supplementary material. We will make our source code and editing results publicly available to foster reproducibility.

4.1. Experimental Setup

Dataset. We select several videos from DAVIS [37]. Each video contains a moving object in 50 to 70 frames. We edit each video with a prompt that describes a target object with a different shape from the original one.

Compared methods. We compare our results with SOTA and several baseline methods. For fair comparisons, all the baseline methods use the same image editing method, Stable Diffusion [44].

• **Multi-frame baseline**: Multiple keyframes in a video are edited individually. The nearby edited keyframes temporally interpolate the remaining frames with FILM [42].

• **Single-frame baseline**: We extract a single keyframe from a video to be edited. The edited information is then propagated to each frame with EbSynth [17].

• **Text2LIVE** [2]: The SOTA text-driven editing method with NLA. Note that it utilizes a structure loss to preserve the original shape. We compare the official Text2LIVE in this section and show the comparison of removing structure loss in our supplementary material.

4.2. Visual Comparison

We show a visual comparison with the baseline methods and Text2LIVE in Fig. 5. In the first example with "`blackswan`→`duck`", the multi-frame baseline shows inconsistent editing in different frames. The single-frame baseline suffers from inaccurate frame motion and thus yields distortion during propagation. Text2LIVE shows a promising target appearance with temporal consistency but cannot change the shape that matches the target object. In contrast, our method provides the desired appearance *and* consistent shape editing. In the second example with "`boat`→`yacht`", the single-frame baseline shows an inconsistent shape since the frame propagation relies on the frame motion of the source shape. Consequently, it cannot propagate the edited pixels correctly in a different shape. In the third example with "`dog`→`cat`", the input video contains a non-rigid motion moving object. It poses further challenges for multi- and single-frame baselines. Again, Text2LIVE demonstrates plausible `cat` appearance while remaining in the source `dog` shape. Our shape-aware method maintains the object motion and manipulates the texture and shape corresponding to the desired editing.

4.3. Ablation Study

We conduct an ablation study in Fig. 6 to validate the effectiveness of the UV deformation and atlas optimization. With fixed NLA UV mapping, the shape edits in the keyframe cannot be adequately transformed through the atlas to each frame (Fig. 6a). Therefore, by adding a keyframe semantic correspondence to deform the target into the source shape, the fixed UV maps the edits correctly into the atlas but remains source shapes in the edited frames (Fig. 6b). To restore the target shape, our deformation module deforms the UV maps by the semantic correspondence (Fig. 6c). However, the unseen pixels and inaccurate correspondence yield artifacts in different views (*e.g.*, in the car's roof and back wheel). We refine the edited atlas and deformation with the atlas optimization (Fig. 6d).

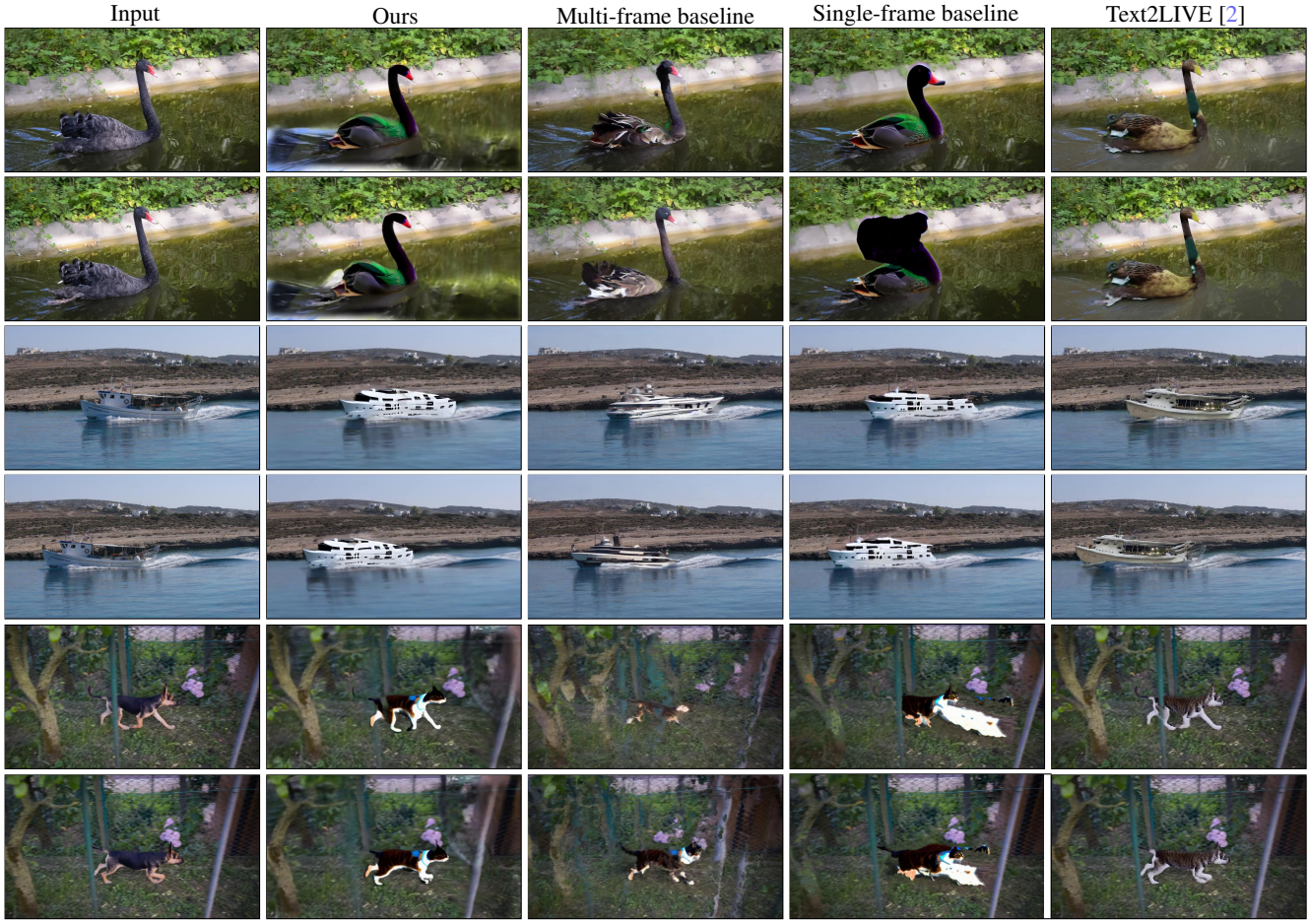


Figure 5. **Visual comparison with baselines and SOTA.** We show three examples with edits of "`blackswan` →`duck`", "`boat` →`yacht`", and "`dog` →`cat`". The multi-frame baseline shows inconsistency in the edited objects. The single-frame method suffers from the incomplete flow motion of the source object shape and thus could not propagate the edits properly. Text2LIVE demonstrates consistent appearance editing corresponding to the target edits. Nevertheless, the shape remains the same as the original object. In contrast, our proposed method outperforms the compared methods with consistent and plausible appearance and shape editing.

Figure 6. **Ablation study.** We study the effects of removing the deformation and optimization components. (a) Editing with fixed NLA UV mapping. (b) Using a semantic correspondence with fixed UV, the edits are mapped to the atlas properly but still remains the original shapes in results. (c) With deformation initialization (Sec. 3.2), the NLA UV maps are deformed to restore the target shape. (d) With further atlas optimization (Sec. 3.3), the incomplete pixels in edited atlas and distortion (in car's roof and back wheel) are refined.

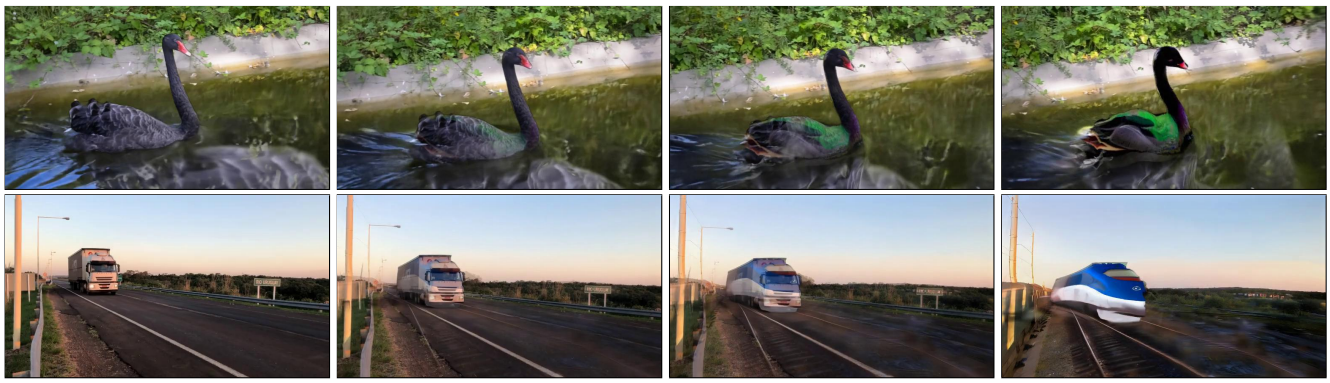


Figure 7. **Shape-aware interpolation.** Our methods allow interpolation between two shapes by simply interpolating the atlas deformation maps. The examples demonstrate the gradual changes from source objects to edited objects over the time.

Figure 8. **Limitations.** We visualize a failure example (`bear →lion`). The inaccurate NLA mapping in the motion of crossing hind legs yields distortion in the edited result.

4.4. Application

We present an application of shape-aware interpolation in Fig. 7. Through interpolating the deformation maps, the object shape can be easily interpolated *without* additional frame interpolation methods. Similarly, we can interpolate atlas textures. Note that we directly apply image editing on the background atlas since it can be treated as a natural panorama image (shown in Fig. 3). However, the foreground atlas is an unwrapped object texture, which is unnatural for general pre-trained editing models. Therefore, we edit the video frame and map it back to the atlas. This approach is more general and allows users to use their chosen images for video editing.

4.5. Limitations

Our method strictly relies on the *many-to-one* mapping from individual frames to a unified atlas. However, NLA may fail to get the ideal mapping in challenging scenarios with complex motions. Therefore, we observe artifacts in the erroneous mapping regions (*e.g.*, the motion of hind legs shown in Fig. 8). In addition, it remains difficult to build semantic correspondence between two different objects. While the atlas optimization can improve noisy correspondences, poor semantic correspondence initialization would hinder the optimization. We show that user manual correction (in Fig. 9) can lead to better video editing results.

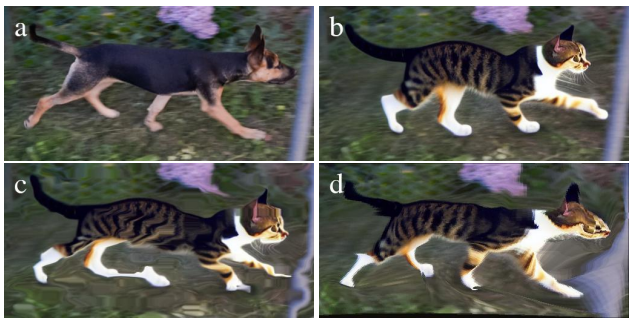


Figure 9. **User-guided correspondence.** Associating two different objects remains challenging even for the SOTA semantic correspondence methods. For a pair of source (a) and target (b), the severe false matching can be corrected by users' manual warping for better results.

5. Conclusions

We have presented a shape-aware text-driven video editing method. We tackle the limitation of appearance-only manipulation in existing methods. We propose a deformation formulation using layered video representation to transform the mapping field corresponding to the target shape edits. We further refine the unseen regions by utilizing the guidance from a pre-trained text-to-image diffusion model. Our method facilitates a variety of shape and texture editing applications.

Societal impacts. Our work proposes a tool for enabling creative video editing applications. Nevertheless, similar to many image/video synthesis applications, care should be taken to prevent misuse or malicious use of such techniques. We will release our code under a similar license as Stable Diffusion that focuses on ethical and legal use.[1]

[1] https://github.com/CompVis/stable-diffusion/blob/main/Stable_Diffusion_v1_Model_Card.md

References

[1] Yogesh Balaji, Seungjun Nah, Xun Huang, Arash Vahdat, Jiaming Song, Karsten Kreis, Miika Aittala, Timo Aila, Samuli Laine, Bryan Catanzaro, et al. ediffi: Text-to-image diffusion models with an ensemble of expert denoisers. *arXiv preprint arXiv:2211.01324*, 2022. 3

[2] Omer Bar-Tal, Dolev Ofri-Amar, Rafail Fridman, Yoni Kasten, and Tali Dekel. Text2live: Text-driven layered image and video editing. *ECCV*, 2022. 2, 3, 6, 7

[3] Kiran S Bhat, Steven M Seitz, Jessica K Hodgins, and Pradeep K Khosla. Flow-based video synthesis and editing. In *ACM SIGGRAPH*. 2004. 3

[4] Fred L. Bookstein. Principal warps: Thin-plate splines and the decomposition of deformations. *IEEE TPAMI*, 1989. 5

[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 *NeurIPS*, 2022. 3

[6] Katherine Crowson, Stella Biderman, Daniel Kornis, Dashiell Stander, Eric Hallahan, Louis Castricato, and Edward Raff. Vqgan-clip: Open domain image generation and editing with natural language guidance. *ECCV*, 2022. 2

[7] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *NeurIPS*, 2021. 3

[8] Kevin Frans, Lisa B Soros, and Olaf Witkowski. Clipdraw: Exploring text-to-drawing synthesis through language-image encoders. *arXiv preprint arXiv:2106.14843*, 2021. 3

[9] Rinon Gal, Or Patashnik, Haggai Maron, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. Stylegan-nada: Clip-guided domain adaptation of image generators. *ACM TOG*, 2022. 3

[10] Chen Gao, Ayush Saraf, Jia-Bin Huang, and Johannes Kopf. Flow-edge guided video completion. In *ECCV*, 2020. 3

[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. In *ECCV*, 2022. 3

[12] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. *arXiv preprint arXiv:2208.01626*, 2022. 3

[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. 3

[14] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *NeurIPS*, 2020. 3

[15] Jia-Bin Huang, Sing Bing Kang, Narendra Ahuja, and Johannes Kopf. Temporally coherent completion of dynamic video. *ACM Transactions on Graphics (TOG)*, 35(6):1–11, 2016. 3

[16] Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. Spatial transformer networks. *NeurIPS*, 2015. 6

[17] Ondřej Jamriška, Šárka Sochorová, Ondřej Texler, Michal Lukáč, Jakub Fišer, Jingwan Lu, Eli Shechtman, and Daniel Sỳkora. Stylizing video by example. *ACM TOG*, 2019. 2, 3, 6

[18] Yoni Kasten, Dolev Ofri, Oliver Wang, and Tali Dekel. Layered neural atlases for consistent video editing. *ACM TOG*, 2021. 1, 3, 4

[19] 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. 1

[20] Gwanghyun Kim, Taesung Kwon, and Jong Chul Ye. Diffusionclip: Text-guided diffusion models for robust image manipulation. In *CVPR*, 2022. 1

[21] Gihyun Kwon and Jong Chul Ye. Clipstyler: Image style transfer with a single text condition. In *CVPR*, 2022. 3

[22] Wei-Sheng Lai, Jia-Bin Huang, Oliver Wang, Eli Shechtman, Ersin Yumer, and Ming-Hsuan Yang. Learning blind video temporal consistency. In *ECCV*, 2018. 3

[23] Chenyang Lei, Yazhou Xing, Hao Ouyang, and Qifeng Chen. Deep video prior for video consistency and propagation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022. 3

[24] Bowen Li, Xiaojuan Qi, Thomas Lukasiewicz, and Philip HS Torr. Manigan: Text-guided image manipulation. In *CVPR*, 2020. 1, 2

[25] Wenbo Li, Pengchuan Zhang, Lei Zhang, Qiuyuan Huang, Xiaodong He, Siwei Lyu, and Jianfeng Gao. Object-driven text-to-image synthesis via adversarial training. In *CVPR*, 2019. 2

[26] Zhen Li, Cheng-Ze Lu, Jianhua Qin, Chun-Le Guo, and Ming-Ming Cheng. Towards an end-to-end framework for flow-guided video inpainting. In *CVPR*, 2022. 3

[27] Wentong Liao, Kai Hu, Michael Ying Yang, and Bodo Rosenhahn. Text to image generation with semantic-spatial aware gan. In *CVPR*, 2022. 2

[28] Sharon Lin, Matthew Fisher, Angela Dai, and Pat Hanrahan. Layerbuilder: Layer decomposition for interactive image and video color editing. *arXiv preprint arXiv:1701.03754*, 2017. 3

[29] Feng-Lin Liu, Shu-Yu Chen, Yukun Lai, Chunpeng Li, Yue-Ren Jiang, Hongbo Fu, and Lin Gao. Deepfacevideoediting: Sketch-based deep editing of face videos. *ACM TOG*, 2022. 3

[30] Xingchao Liu, Chengyue Gong, Lemeng Wu, Shujian Zhang, Hao Su, and Qiang Liu. Fusedream: Training-free text-to-image generation with improved clip+ gan space optimization. *arXiv preprint arXiv:2112.01573*, 2021. 2

[31] Sebastian Loeschcke, Serge Belongie, and Sagie Benaim. Text-driven stylization of video objects. *arXiv preprint arXiv:2206.12396*, 2022. 3

[32] 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. *ACM TOG*, 2020. 3

[33] Erika Lu, Forrester Cole, Tali Dekel, Andrew Zisserman, William T Freeman, and Michael Rubinstein. Omnimatte: Associating objects and their effects in video. In *CVPR*, 2021. 3

[34] Seonghyeon Nam, Yunji Kim, and Seon Joo Kim. Text-adaptive generative adversarial networks: manipulating images with natural language. *NeurIPS*, 2018. 1

[35] Alex 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 *ICML*, 2022. 3

[36] Or Patashnik, Zongze Wu, Eli Shechtman, Daniel Cohen-Or, and Dani Lischinski. Styleclip: Text-driven manipulation of stylegan imagery. In *ICCV*, 2021. 3

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

[38] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988*, 2022. 2, 5

[39] 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 *ICML*, 2021. 2

[40] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *ICML*. PMLR, 2021. 1

[41] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *ICML*, 2021. 2

[42] Fitsum Reda, Janne Kontkanen, Eric Tabellion, Deqing Sun, Caroline Pantofaru, and Brian Curless. Film: Frame interpolation for large motion. *ECCV*, 2022. 2, 6

[43] Scott Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. Generative adversarial text to image synthesis. In *ICML*. PMLR, 2016. 2

[44] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 1, 2, 3, 4, 5, 6

[45] 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. 3

[46] 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. 3

[47] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *arXiv preprint arXiv:2210.08402*, 2022. 3

[48] Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, et al. Make-a-video: Text-to-video generation without text-video data. *arXiv preprint arXiv:2209.14792*, 2022. 3

[49] Ivan Skorokhodov, Sergey Tulyakov, and Mohamed Elhoseiny. Stylegan-v: A continuous video generator with the price, image quality and perks of stylegan2. In *CVPR*, 2022. 3

[50] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 3

[51] Prune Truong, Martin Danelljan, Fisher Yu, and Luc Van Gool. Probabilistic warp consistency for weakly-supervised semantic correspondences. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8708–8718, 2022. 2, 4

[52] 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. *arXiv preprint arXiv:2210.02399*, 2022. 3

[53] Xintao Wang, Liangbin Xie, Chao Dong, and Ying Shan. Real-esrgan: Training real-world blind super-resolution with pure synthetic data. In *International Conference on Computer Vision Workshops (ICCVW)*. 6

[54] Weihao Xia, Yujiu Yang, and Jing-Hao Xue. Gan inversion for consistent video interpolation and manipulation. *arXiv preprint arXiv:2208.11197*, 2022. 3

[55] Weihao Xia, Yujiu Yang, Jing-Hao Xue, and Baoyuan Wu. Tedigan: Text-guided diverse face image generation and manipulation. In *CVPR*, 2021. 2

[56] Tao Xu, Pengchuan Zhang, Qiuyuan Huang, Han Zhang, Zhe Gan, Xiaolei Huang, and Xiaodong He. Attngan: Fine-grained text to image generation with attentional generative adversarial networks. In *CVPR*, 2018. 2

[57] Yiran Xu, Badour AlBahar, and Jia-Bin Huang. Temporally consistent semantic video editing. *arXiv preprint arXiv:2206.10590*, 2022. 3

[58] Zipeng Xu, Tianwei Lin, Hao Tang, Fu Li, Dongliang He, Nicu Sebe, Radu Timofte, Luc Van Gool, and Errui Ding. Predict, prevent, and evaluate: Disentangled text-driven image manipulation empowered by pre-trained vision-language model. In *CVPR*, 2022. 3

[59] Vickie Ye, Zhengqi Li, Richard Tucker, Angjoo Kanazawa, and Noah Snavely. Deformable sprites for unsupervised video decomposition. In *CVPR*, 2022. 3

[60] 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 *ICLR*, 2022. 3

[61] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris N Metaxas. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In *ICCV*, 2017. 2