Space-Time Diffusion Features for Zero-Shot Text-Driven Motion Transfer

Danah Yatim[1*] Rafail Fridman[1*] Omer Bar Tal[1] Yoni Kasten[2] Tali Dekel[1]
Weizmann Institute of Science[1] NVIDIA Research[2]
*Indicates equal contribution.

Figure 1. Given an input video and a text prompt describing the target objects and scene, our method generates a new video in which the overall motion and scene layout of the input video are preserved, while allowing for notable structural and appearance changes.

Abstract

We present a new method for text-driven motion transfer – synthesizing a video that complies with an input text prompt describing the target objects and scene while maintaining an input video's motion and scene layout. Prior methods are confined to transferring motion across two subjects within the same or closely related object categories and are applicable for limited domains (e.g., humans). In this work, we consider a significantly more challenging setting in which the target and source objects differ drastically in shape and fine-grained motion characteristics (e.g., translating a jumping dog into a dolphin). To this end, we leverage a pre-trained and fixed text-to-video diffusion model, which provides us with generative and motion priors. The pillar of our method is a new space-time feature loss derived directly from the model. This loss guides the generation process to preserve the overall motion of the input video while complying with the target object in terms of shape and fine-grained motion traits. [1] *Project page:* https://diffusion-motion-transfer.github.io/

1. Introduction

Imagine transferring the car's motion shown in the video in Fig. 1 to a different object, such as a bicycle or a train. This task poses a crucial challenge – retaining key motion characteristics of the input video, such as the car's direction of motion, speed, and pose, while substantially altering the dynamic object's structure to convey the target's unique visual attributes. Conceptually, solving this task requires prior knowledge about objects' appearance, pose, shape, and motion under deformations and different viewpoints, as well as an understanding of their interaction with the scene (e.g., turning the car into a bicycle requires revealing unseen background content and synthesizing plausible scene effects, such as shadows).

Previous methods have been primarily focused on transferring motion across two similarly-looking videos depicting two subjects within the same object category. These methods are typically confined to object categories for which strong geometric priors exist, e.g., humans for which parametric models for robust shape and pose are available (e.g., [7, 11]). Other works attempt to learn such explicit mid-level representation in a self-supervised manner from the input videos (e.g., [39, 60])– an extremely challenging task by itself. In this work, we take the task of motion transfer to the realm of text-driven video editing, where

[1] Code will be made publicly available.

the target object and scene are specified via a text prompt. We aim at addressing a significantly more general setting, which involves transferring motion across different object categories under significant variations in shape and deformations across time (Fig. 1)

Our approach diverges from traditional motion transfer works by avoiding explicit mid-level modeling of pose and shape. Instead, we harness the *generative motion priors* learned from broad video data by a pre-trained and fixed *text-to-video model*. Specifically, we delve into the intermediate *space-time feature representation* learned by the video model and introduce a new loss that guides the generation process of the target video towards the preservation of the original video's overall scene layout and motion. Importantly, our method allows flexibility and deviations from the exact structure and shape of the source objects. This contrasts with prior works in text-driven image and video editing that manipulate *spatial features* of a pre-trained *text-to-image* model. These methods inherently lack the ability to perform consistent structural edits across frames since they rely solely on spatial image priors. To the best of our knowledge, our work is the first to empirically analyze and harness *space-time* features of a text-to-video model.

Our lightweight framework works in a zero-shot fashion, requiring no training data or fine-tuning. We demonstrate results on many videos and edits, encompassing various objects and scenes. We further suggest a new metric to measure motion similarity under shape deviation and quantitatively evaluate our method w.r.t. existing text-to-video methods, demonstrating state-of-the-art performance in terms of motion preservation and edit fidelity.

To summarize, our key contributions include:

- An effective zero-shot framework that harnesses the generative motion prior of a pre-trained text-to-video model for the task of motion transfer.
- New insights about the *space-time* intermediate features learned by a pre-trained text-to-video diffusion model.
- A new metric for evaluating motion fidelity under structural deviations between two videos.
- State-of-the-art results compared to competing methods, achieving a significantly better balance between motion preservation and fidelity to the target prompt.

2. Related works

Motion transfer. A related task to ours is motion transfer from a source to a target subject, where the subjects are of the same or closely related object categories (e.g., [1, 11, 29, 53, 54, 66, 67]). These methods take as input a driving video depicting the source motion, and an image or a video, depicting the target subject. A prevalent approach among these methods is to explicitly model the pose of the dynamic object via a parametric model (e.g. OpenPose [7]). Thus, these methods are largely restricted to domains for which robust parametric models exist (e.g., humans or faces) or to transferring motion across videos depicting similar motion and closely related object categories. In contrast, we are aimed at *text-driven motion transfer* across *distinct object categories*. That is, the target object and scene are specified through a text prompt, where the source and target objects can differ significantly in shape, appearance, and their natural fine-grained motion traits.

Text-to-video models. Early works on text-to-video generation utilized VAE or conditional GAN frameworks [32, 38, 43] trained on small-scale datasets of simple domains (e.g., moving digits). Recently, there have been substantial efforts in training large-scale text-to-video models on vast datasets with autoregressive Transformers (e.g., [24, 63, 68]) or Diffusion Models (e.g., [5, 22]). A prominent recent trend extends pre-trained text-to-image (T2I) diffusion models to text-to-video (T2V) generation by adding temporal layers on top of an image architecture [4, 18, 46, 56]. Make-A-Video [56] add temporal Convs and attention layers to a pre-trained T2I pixel-space diffusion model. Other works extend T2I diffusion models that operate on a latent space (e.g., StableDiffusion [50]) to the temporal domain [4, 18].

Several works [13, 17, 40, 65] train or fine-tune diffusion models for video-to-video translation tasks. Gen-1 [17] design a video architecture that is conditioned on structure/appearance representations, allowing text-driven appearance manipulation of a reference video. Control-A-Video [13] extends a conditional T2I to the temporal domain, allowing the generation of videos that preserve the per-frame layout of a reference video. Nevertheless, since these methods are conditioned on generic mid-level representations of the reference video, they do not allow the preservation of the motion of the reference video *while significantly deviating from the per-frame structural layout*. In this work, we utilize a pre-trained publicly available T2V diffusion model [8, 64] and show for the first time how it can be leveraged for motion transfer in a zero-shot manner.

Image & video editing via feature manipulation. There has been unprecedented progress in text-to-image generation using diffusion models [14, 15, 21, 42]. Following this success, a surge of works empirically analyzed the internal feature representation of prominent T2I diffusion models, e.g., StableDiffusion [50], and showed how to perform various editing tasks using simple operations in the T2I feature space.[6, 12, 20, 23, 35, 45, 62]. Prompt-to-Prompt [20] analyzed the cross-attention maps and showed how to manipulate them for controlling the spatial composition in generated images. Plug-and-Play Diffusion (PnP) [62] showed that the spatial features capture semantic information at high spatial granularity and utilized them for image-to-image translation.

A prominent line of works adopt a pre-trained T2I diffusion model for *video* editing [9, 19, 28, 48, 69]. For example, Tune-A-Video [69] fine-tunes a T2I model on a single input video and uses the fine-tuned model for stylizing the video or replacing object categories. TokenFlow [19] performs consistent video editing in a zero-shot manner by enforcing consistency on the diffusion features across frames. Nevertheless, these methods do not have access to

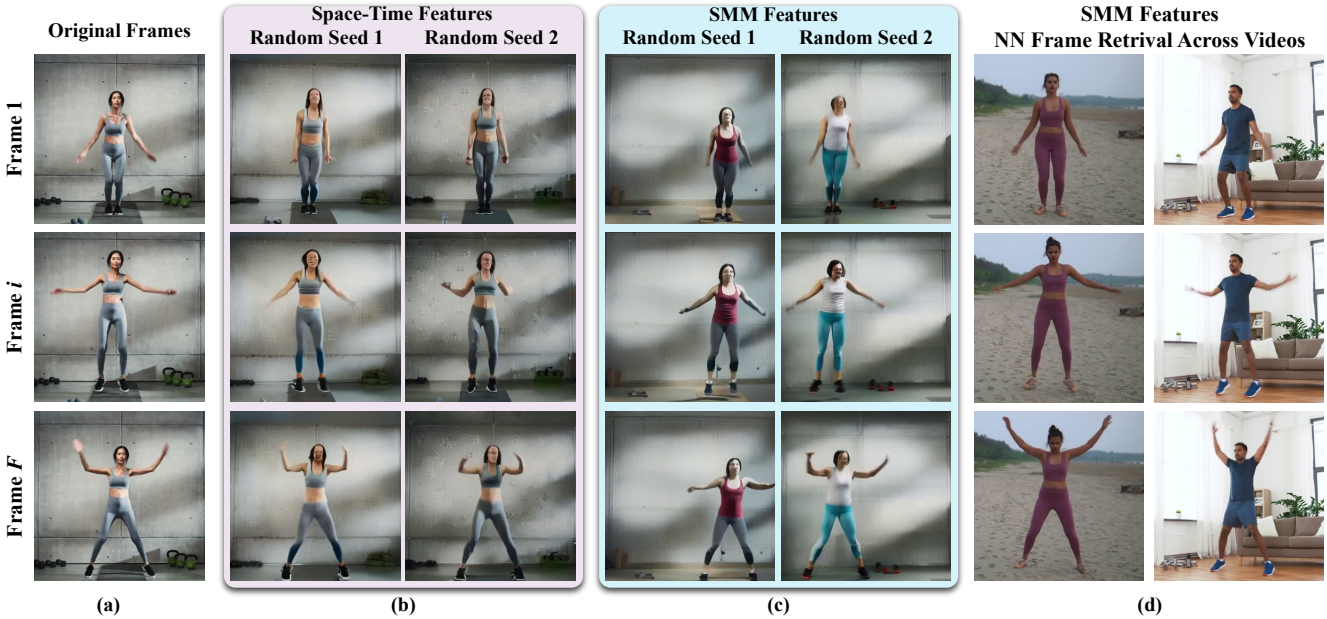


Figure 2. **Diffusion feature inversion via guided feature reconstruction.** We extract space-time features $\boldsymbol{f}$ from an input video (a) and steer the generation process of a random sample to produce the same feature $\boldsymbol{f}$, using feature reconstruction as guidance (b); the synthesized videos closely resemble the original video content in terms of appearance, shape, and pose. Replacing the full space-time features with their spatial marginal mean feature $\texttt{SMM}[\boldsymbol{f}]$ allows for more flexibility (c); the SMM feature inversion results capture the original object pose, general position, and scene layout yet are not restricted to the original content at the pixel-level. This is also demonstrated in the nearest neighbor frames retrieved from other videos depicting similar actions, according to similarly in $\texttt{SMM}[\boldsymbol{f}]$ features (c).

a T2V generative motion prior and are not designed for edits that require significant structural deviation from the original video. Unlike these works, we leverage the motion prior of a large-scale T2V model, which allows us to support edits that require shape deviation (e.g., car to bike in Fig. 1), utilizing a novel loss function that we derive from the model's feature space. To the best of our knowledge, our work is the first to investigate the internal feature representation of a T2V model and leverage it for an editing task.

Consistent video editing. Several recent methods have leveraged layered video representations for consistent text-driven video editing [2, 10, 25, 27, 31, 34, 71]. For example, Text2LIVE [2] proposed to use a pre-trained Layered Neural Atlases (NLA) [27] representation, which is edited using losses defined in CLIP text-image space [49]. However, since the structure and motion of the edited video are determined via the pre-trained NLA, such methods are restricted to only appearance changes. Recently, [31] generalized this approach to allow local structural changes in the edited videos. Nevertheless, NLA takes hours to train, and cannot faithfully represent complex videos due to the strong regularization of objects' motion. In addition, all these methods use a 2D generative prior, thus cannot support large structure deviation and adaption of motion.

3. Preliminary

Diffusion models. Diffusion models [15, 21, 46, 57] are generative models that aim to approximate a data distribution q by mapping an input noise $\boldsymbol{x}_T \sim \mathcal{N}(0, I)$ to a clean sample $\boldsymbol{x}_0 \sim q$, through an iterative denoising process. The DDIM sampler allows to denoise an initial noise $\boldsymbol{x}_T$ in a deterministic manner [58]. By applying DDIM inversion, a clean sample $\boldsymbol{x}_0$ can be mapped back to the sequence of noisy samples $\{\boldsymbol{x}_i\}_{t=1}^{T}$ used to generate it.

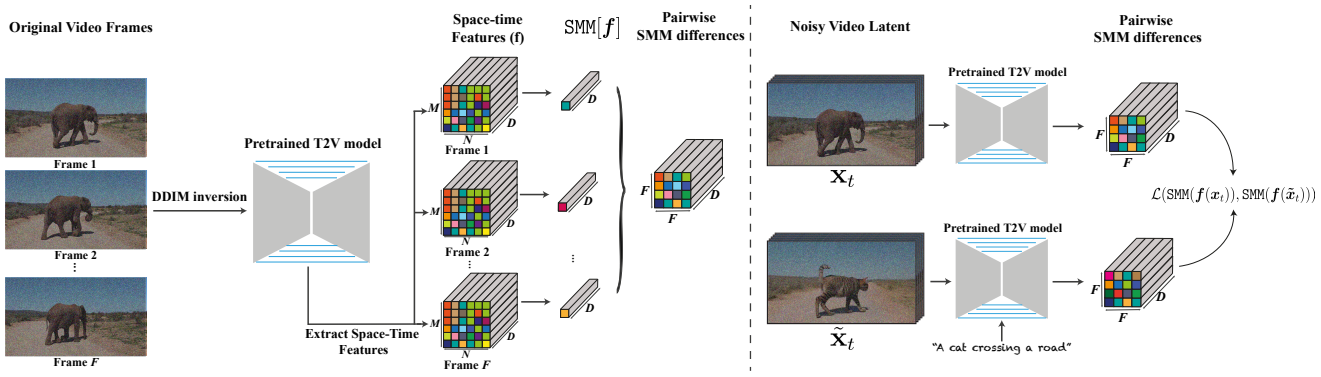
In latent text-to-image (T2I) diffusion models, e.g., StableDiffusion [50], a pre-trained encoder compresses an RGB image $\boldsymbol{I} \in \mathbb{R}^{H\times W\times 3}$ to a latent $\boldsymbol{x} \in \mathbb{R}^{H'\times W'\times 4}$, which can be decoded back to the high-resolution space. These T2I models comprise a UNet architecture [52], which consists of Convolutions and Attention modules.

Latent video diffusion models (e.g., [4, 64]) extend latent T2I models to text-to-video (T2V) by inflating the 2D architecture to the temporal domain, i.e., adding temporal convolutions and temporal attention layers, and fine-tuning on video datasets. In this case, the T2V model generates a latent video $\boldsymbol{x} \in \mathbb{R}^{F\times H'\times W'\times 4}$, which is then decoded to the output RGB video $\mathcal{V} \in \mathbb{R}^{F\times H\times W\times 3}$. In this work, we utilize the publicly available ZeroScope T2V [8] model, which inflates StableDiffusion [50].

4. Method

Given an input video $\mathcal{V}$ and a target text prompt P, our goal is to generate a new video $\mathcal{J}$ that preserves the overall motion and semantic layout of $\mathcal{V}$, while complying with P. We utilize ZeroScope – a pre-trained latent T2V model [8, 64].

The key component of our method, illustrated in Fig. 3,



(a) Spatial Marginal Mean (SMM) Feature (b) Motion-Guided Video Generation

Figure 3. **Pipeline.** (a) Given an input video, we apply DDIM inversion and extract space-time features $\boldsymbol{f} \in \mathbb{R}^{F\times M\times N\times D}$ from intermediate layer activations. We obtain our Spatial Marginal Mean (SMM) feature $\texttt{SMM}[\boldsymbol{f}] \in \mathbb{R}^{F\times D}$ by computing the mean over the spatial dimensions, and compute the pairwise differences between each pair of SMM features. (b) For editing, we guide the generation at each denoising step with our Pairwise SMM differences objective (b). See Sec. 4 for more details.

is a novel objective function that is used as guidance during the generation process of $\mathcal{J}$. We conceive this objective based on our empirical analysis that reveals new insights about space-time diffusion features extracted from the model. Specifically, we show that the first-order statistics of the features in the spatial dimensions, i.e., the spatial marginal mean of the features, can serve as a powerful per-frame global descriptor that (i) retains spatial information such as objects' position, pose, and the semantic layout of the scene, and (ii) robust to pixel-level variations in both appearance and shape.

4.1. Space-time analysis of diffusion features

We focus our analysis on features extracted from the intermediate layer activations of the video model. Recall that the video model is initialized from a text-to-image model, for which the semantic DIFT features [59], were shown to encode localized *semantic* information shared across objects from different domains [59, 62]. Here, we examine the *space-time* properties of the corresponding features in the *video model*. See Supplementary Material (SM) for details about the video model architecture and feature selection.

Given the input video $\mathcal{V}$, we apply DDIM inversion with an empty prompt [58], and obtain a sequence of latents $[\boldsymbol{x}_1, ..., \boldsymbol{x}_T]$, where $\boldsymbol{x}_t$ is the video latent at generation step t. We input the latent $\boldsymbol{x}_t$ to the network and extract the space-time features $\boldsymbol{f}(\boldsymbol{x}_t) \in \mathbb{R}^{F\times M\times N\times D}$, where F, M, N are the number of frames, height and width of the D dimensional feature activation, respectively.

Diffusion feature inversion. To gain a better understanding of what the features $\{\boldsymbol{f}(\boldsymbol{x}_t)\}_{t=1}^{T}$ encode, we adopt the concept of "feature inversion" [36, 55, 61]. Our goal is to optimize for a video $\mathcal{V}^*$, randomly initialized, that would produce these features when fed into the network. Specifically, this is achieved using feature reconstruction guidance [16, 41] during the sampling process of $\mathcal{V}^*$. Formally,

$$\hat{\boldsymbol{x}}_T \sim \mathcal{N}(0, \mathcal{I})$$
$$\hat{\boldsymbol{x}}_{t-1} = \Phi(\boldsymbol{x}_t^*, P_s), \text{where } \boldsymbol{x}_t^* = \text{argmin}_{\hat{\boldsymbol{x}}} \|\boldsymbol{f}(\boldsymbol{x_t}) - \boldsymbol{f}(\hat{\boldsymbol{x}}_\mathbf{t})\|^2$$

Here, Φ is the diffusion model, and P_s is a general text prompt describing the input video (e.g., "a car"). We minimize the feature reconstruction objective using gradient descent at each generation step. See SM for more details.

Figure 2 shows our inversion results for the space-time features extracted from an input video; we repeat the inversion process several times, each with different random initialization (i.e., different seeds). We observe that inverted videos nearly reconstruct the original frames (Fig. 2(b)).

Ultimately, we opt to find a feature descriptor that retains such information about objects' pose and the semantic layout of the scene yet is robust to variations in appearance and shape. To reduce the dependency on the pixel-level information, we introduce a new feature descriptor, dubbed *Spatial Marginal Mean (SMM)*, obtained by reducing the spatial dimensions. Formally,

$$\texttt{SMM}[\boldsymbol{f}(\boldsymbol{x}_t)] = \frac{1}{M \cdot N} \sum_{i=1}^{M} \sum_{j=1}^{N} \boldsymbol{f}(\boldsymbol{x}_t)_{i,j} \tag{1}$$

Where $\boldsymbol{f}(\boldsymbol{x}_t)_{i,j} \in \mathbb{R}^D$ is the entry at spatial location (i, j) in the space-time feature volume $\boldsymbol{f}(\boldsymbol{x}_t)$.

We repeat the inversion experiment (Eq. 1), with $\{\texttt{SMM}[\boldsymbol{f}(\boldsymbol{x}_t)]\}_{t=1}^{T}$ as the target features to reconstruct. Figure 2(c) shows the inversion results for the SMM features, for different initializations. Remarkably, although the spatial dimensions are collapsed in the SMM features, the inverted videos convey the correct pose and position of objects, while depicting larger structural and appearance variations than using the full space-time features.

We further demonstrate these properties by treating the spatial marginal mean associated with each frame as a global per-frame descriptor, and using it to retrieve nearest neighbour frames from other videos. As seen in Fig. 2(d), the retrieved nearest frames depict the same pose, under noticeable appearance and viewpoint changes.

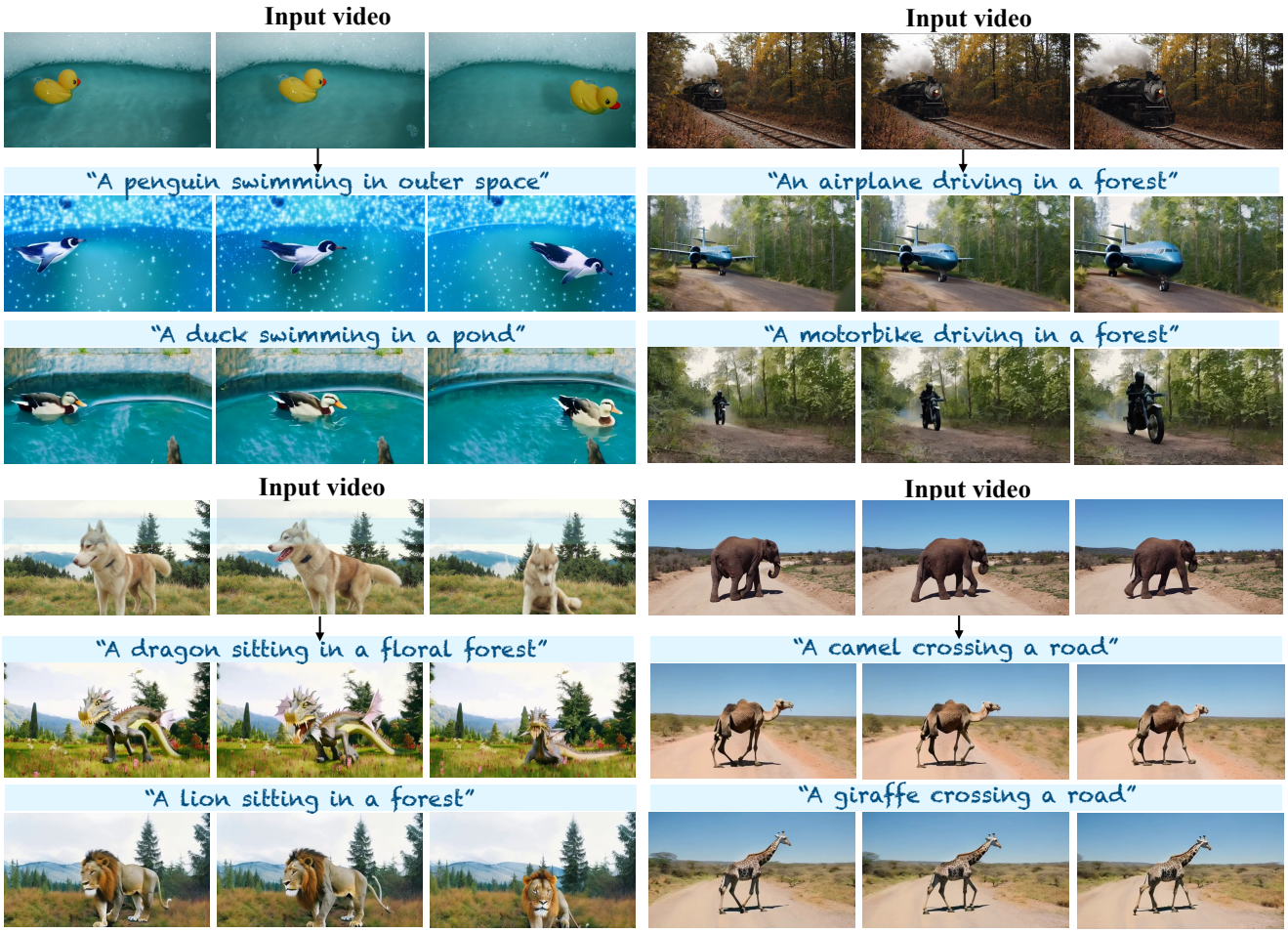


Figure 4. **Sample results of our method.** See SM for full video results.

4.2. Motion-guided video generation

Based on our findings, we now turn to the task of generating a new video $\mathcal{J}$ that complies with the input prompt P and preserves the motion characteristics of the driving video $\mathcal{V}$.

Our feature inversion analysis raises the question of whether the same approach can be used for editing, by simply replacing the source prompt P_s with an edit prompt P in Eq. 1.

Figure 7 shows these results for a couple of videos, which demonstrate two issues: (i) depending on the initialization, the optimization may converge to a local minima in which the accurate position of objects and their orientation may differ from the input, (ii) the SMM features still retain appearance information, which reduces the fidelity to the text prompt. We propose the following two components to resolve these issues.

Pairwise SMM differences. As seen in Fig. 7, directly optimizing for the SMM features often prevents us from deviating from the original appearance. To circumvent this problem, we propose an objective function that aims to preserve the pairwise differences of the SMM features, rather than their exact values. Formally, let $\phi_i^t, \tilde{\phi}_i^t \in \mathbb{R}^d$ be the SMM features for frame i and step t for the driving video and the generated video, respectively. For a generation step t, the pairwise SMM differences $\Delta^t, \tilde{\Delta}^t \in \mathbb{R}^{F\times F\times d}$ are defined as follows:

$$\Delta^t_{(i,j)} = \phi^t_i - \phi^t_j \qquad \tilde{\Delta}^t_{(i,j)} = \tilde{\phi}^t_i - \tilde{\phi}^t_j \tag{2}$$

for $i, j \in \{1, \dots, F\}$. Our loss for time step t is then:

$$\mathcal{L}(\texttt{SMM}(\boldsymbol{f}(\boldsymbol{x}_t)), \texttt{SMM}(\boldsymbol{f}(\tilde{\boldsymbol{x}}_t))) = \sum_i \sum_j ||\Delta^t_{(i,j)} - \tilde{\Delta}^t_{(i,j)}||_2^2 \tag{3}$$

Intuitively, this loss lets us preserve the relative changes in the features through time, while discarding the exact appearance information of the source video (Fig. 7).

Initialization. It is well-known that the diffusion denoising process is performed in a coarse-to-fine manner, thus, the initialization plays an important role in defining the low frequencies of the generated content [3, 37]. Initialization from a random point may often converge to an undesired local minimum, in which object position is not well-preserved. Note that the low-frequency information of the original video is readily available in the DDIM inverted

Algorithm 1 Motion-Guided Video Generation

Input:

$\mathcal{V}, \mathcal{P}$ ▷ Input video and target text prompt

$\{\boldsymbol{x}_t\}_{t=1}^{T} \leftarrow$ DDIM-Inv$[\mathbf{V}]$ $\quad \forall t \in [T]$

$\epsilon_0 \sim \mathcal{N}(0, \mathcal{I})$

$\tilde{\boldsymbol{x}}_T = LF_\xi(\boldsymbol{x}_T) + (\epsilon_0 - LF_\xi(\epsilon_0))$ ▷ Filtered noise (Eq. 4.2)

For $t = T, \ldots, 1$ **do**

$\boldsymbol{f}(\boldsymbol{x}_t), \boldsymbol{f}(\tilde{\boldsymbol{x}}_t) \leftarrow$ Extract space-time features

$\texttt{SMM}(\boldsymbol{f}(\boldsymbol{x}_t)), \texttt{SMM}(\boldsymbol{f}(\tilde{\boldsymbol{x}}_t)) \leftarrow$ Spatial marginal mean (Eq. 1)

$\boldsymbol{x}_t^* = \text{argmin}_{\tilde{\boldsymbol{x}}_t} \mathcal{L}(\texttt{SMM}(\boldsymbol{f}(\boldsymbol{x}_t)), \texttt{SMM}(\boldsymbol{f}(\tilde{\boldsymbol{x}}_t)))$

$\tilde{\boldsymbol{x}}_{t-1} = \Phi(\boldsymbol{x}_t^*, P)$ ▷ Apply a denoising step

Output: $\mathcal{J} \leftarrow \boldsymbol{x}_0$

noise $\boldsymbol{x}_T$. However, we empirically found that this initialization may often restrict edit-ability [44] (see SM for an example). We thus extract only the low frequencies from $\boldsymbol{x}_T$. Specifically, let $\boldsymbol{x} \in \mathbb{R}^{F \times M \times N}$ be a tensor representing F frames, with a spatial resolution of $M \times N$. We denote by $LF_\xi(\boldsymbol{x})$ the operation of spatially downsampling and upsampling $\boldsymbol{x}$ by a factor of ξ. Then, our initial latent $\tilde{\boldsymbol{x}}_T$ is given by:

$$\tilde{\boldsymbol{x}}_T = LF_\xi(\boldsymbol{x}_T) + (\epsilon_0 - LF_\xi(\epsilon_0)) \tag{4}$$

where $\epsilon_0 \sim \mathcal{N}(0, I)$ is a random noise. Intuitively, $\tilde{\boldsymbol{x}}_T$ preserves the low-frequencies of the DDIM noise where the higher frequencies are determined by ϵ_0.

To summarize, starting from the filtered latent $\tilde{\boldsymbol{x}}_T$, our method deploys the following guided generation process:

$$\boldsymbol{x}_t^* = \text{argmin}_{\tilde{\boldsymbol{x}}_t} \mathcal{L}(\texttt{SMM}(\boldsymbol{f}(\boldsymbol{x}_t)), \texttt{SMM}(\boldsymbol{f}(\tilde{\boldsymbol{x}}_t)))$$
$$\tilde{\boldsymbol{x}}_{t-1} = \Phi(\boldsymbol{x}_t^*, P)$$

Our full framework is summarized in Alg. 1.

5. Results

We evaluate our method on various scenes and object categories, most of which involve camera as well as object motion. The driving videos are taken from DAVIS dataset [47] and from the Web. Our video results and implementation details are available in the Supplementary Materials (SM).

Figures 1, 4 show sample results of our method. As seen, our method facilitates edits that involve notable changes to the shape and structure of deforming objects, while preserving camera and objects' motion. For instance, we preserve the 3D pose of the car when transferring its motion to a bike or a train (Fig. 1); and maintain the actions of non-rigidly moving objects, e.g., sitting dog or walking camel in Fig. 4.

Baselines We compare our method to the following text-driven video editing methods: (i) *Shape-Aware Layered Neural Atlases (SA-NLA)* [31] that utilizes a pre-trained layered video representation [27] and a pre-trained T2I model [50]. (ii) *TokenFlow* [19], a zero-shot method that works in the feature space of a pre-trained T2I model (iii) *GEN-1* [17] and (iv) *Control-A-Video* [13], both are video-to-video diffusion models that condition the generation on input depth maps, (v) *Tune-A-Video* [69] that inflates a T2I model and finetunes it on a single test video, and (iv) SDEdit [37] applied to the same T2V model as our method.

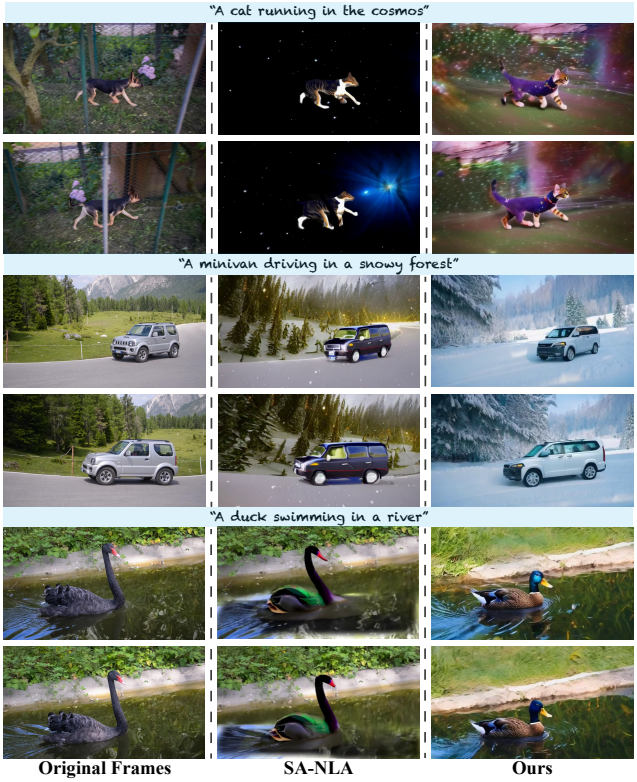


Figure 5. **Comparison to SA-NLA [31]**. See SM for video results.

5.1. Qualitative evaluation

Figure 5 shows a qualitative comparison to SA-NLA [31]. Note that SA-NLA utilizes a layered video representation [27], which requires foreground/background separation masks and takes ∼10 hours to train. Thus, we compare to their provided videos and edit prompts qualitatively. As seen in Fig. 5, both our method and SA-NLA exhibit high fidelity to the original motion. Nevertheless, our method allows for greater deviation in structure, (e.g., matching the structure of a duck in the swan example) and adaption of fine-grained motion traits, which are necessary for capturing the unique attributes of the target object. For example, adapting the shape and movement of a dog's tail to resemble a naturally-looking cat's tail.

Figure 6 provides comparisons to the additional baselines. As seen, none of these methods can both convey the original motion and adhere to the edit prompt. TokenFlow [19] is tailored to preserve structure of the input video. Gen-1 [17] and Control-A-Video [13] struggle to deviate from the input shapes as they condition the generation on the per-frame depth maps extracted from the input video. Tune-A-Video [69] manages to fulfill the target prompt, yet objects are not aligned in pose and motion (e.g., camel to bear). Our method significantly outperforms these baselines, by suc-

Figure 6. **Comparison.** Sample results comparing our method to TokenFlow [19], Gen-1 [17], Tune-A-Video [69], and Control-A-Video [13]. See SM for full video comparisons.

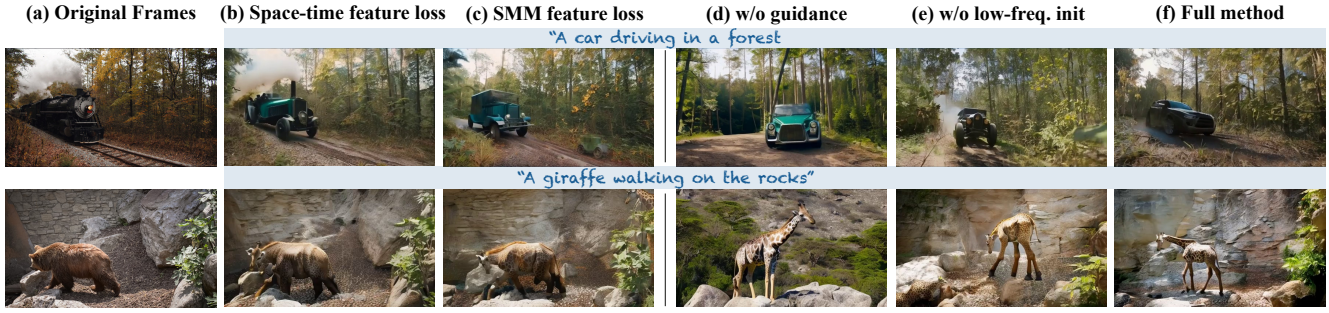


Figure 7. **Ablation.** (b-c) We compare alternative loss functions instead of our pairwise SMM differences loss (Eq. 4.2); (b) using full space-time features reconstruction prevents deviations in appearance and shape; and (c) SMM feature reconstruction allows for more flexibility yet retains appearance information. (d-f) Ablation of our key components: (d) directly sampling from the initial latent w/o optimization preserves only the coarse layout. (e) Starting our optimization from randomly sampled noise (w/o low frequency filtering Eq. 4.2) results in lower motion fidelity compared to our full method (f).

cessfully matching the desired edits, which require significant structural changes and may involve synthesizing dynamic scene elements (e.g., smoke behind the motorbike). We refer the reader to the SM for full video results.

Quantitative evaluation. We numerically evaluate the key aspects of our results using the following metrics:

(i) Edit fidelity. Following previous works (e.g., [17, 19]), we use CLIP [49] to measure the similarity between each frame and the target text and report the average score.

(ii) Motion fidelity. We aim to assess the fidelity of our results in preserving the original motion. Given our task involves structural deviations, there is no alignment between pixels in the original and output videos. Consequentially, traditional metrics such as comparing optical-flow fields are unsuitable for our use case. We thus introduce a new metric, based on similarity between *unaligned* long-trajectories. We expect that even under structural changes, the two sets of trajectories would exhibit shared characteristics.

Specifically, we use off-the-shelf tracking method [26]

to estimate $\mathcal{T} = \{\tau_1, \ldots, \tau_n\}, \tilde{\mathcal{T}} = \{\tilde{\tau}_1, \ldots, \tilde{\tau}_m\}$, two sets of tracklets in the input and output videos, respectively.

Inspired by the Chamfer distance, we define our *Motion-Fidelity-Score* as follows. For each tracklet $\tau_i \in \mathcal{T}$, we measure the similarity to its nearest neighbor in $\tilde{\tau}_i \in \tilde{\mathcal{T}}$, and vice versa.

$$\frac{1}{m}\sum_{\tilde{\tau}\in\tilde{\mathcal{T}}}\max_{\tau\in\mathcal{T}}\textbf{corr}(\tau,\tilde{\tau}) + \frac{1}{n}\sum_{\tau\in\mathcal{T}}\max_{\tilde{\tau}\in\tilde{\mathcal{T}}}\textbf{corr}(\tau,\tilde{\tau}) \qquad (5)$$

where the correlation between two tracklets $\textbf{corr}(\tau, \tilde{\tau})$ is computed as follows, similarly to [33]:

$$\textbf{corr}(\tau,\tilde{\tau}) = \frac{1}{F}\sum_{k=1}^{F}\frac{v_k^x\cdot\tilde{v}_k^x + v_k^y\cdot\tilde{v}_k^y}{\sqrt{(v_k^x)^2+(v_k^y)^2}\cdot\sqrt{(\tilde{v}_k^x)^2+(\tilde{v}_k^y)^2}}$$

where $(v_k^x, v_k^y), (\tilde{v}_k^x, \tilde{v}_k^y)$ are the k^{th} frame displacement of tracklets $\tau, \tilde{\tau}$ respectively.

Figure. 8 reports the metrics above for a set of 54 video-edit text pairs containing 21 unique videos. Our method outperforms the baselines by achieving both high fidelity to the target text prompt and the original motion. As expetcted, TokenFlow [19] achieves high motion fidelity score, yet a low edit fidelity score. Control-A-Video [13] exhibits a similar behaviour since it utilizes depth maps as a guidance signal to edit the video. Tune-A-Video [69] shows an inverse trend, i.e., satisfying the desired edit at the cost of motion fidelity. We further consider SDEdit [37] with different noise levels, none of which can resolve the motion-edit tradeoff. Note that Gen1's API outputs a different number of frames, thus we could not quantitatively evaluate their performance.

(iii) *User study.* We employ the Two-alternative Forced Choice (2AFC) protocol for text-driven video editing [2, 17, 19, 48]. Participants are presented with the input video, our results and a baseline, and are asked to determine which video better aligns with the text prompt while preserving the motion of the original video. We collected 7000 user judgments from 150 users. As seen in Table 1, our method is consistently preferred over all baselines.

5.2. Ablations

In Fig. 7, we ablate key design choices in our framework. First, we consider alternative loss functions by substituting the pairwise SMM differences in Eq.4.2 with: (i) full space-time features $\boldsymbol{f}$ and (ii) SMM features (Eq.1). Figure 7 (b) shows that space-time features restrict both shape and appearance variations; optimizing for SMM features (c) increases flexibility yet is insufficient for matching the edit.

We next ablate our guided sampling and latent initialization strategy. Sampling from the initial latent without optimization (d) converges to unrelated objects' pose, while initializing the optimization from randomly sampled noise (e) fails to retain the original motion characteristics.

6. Discussion and conclusions

We tackled the task of text-driven motion transfer, focusing on scenarios where the source and target objects differ in shape and fine-grained motion traits. We introduced a zero-shot method that utilizes a pre-trained text-to-video diffusion model, through a simple optimization framework. Our work is the first to analyze and reveal new insights about space-time T2V features, and the first to show how to harness their properties for text-driven motion transfer.

As for limitations, our performance relies on the generative priors learned by the T2V model. Thus, in some cases, the combination of target object and input video motion may be out-of-distribution for the T2V model. In this case, the motion fidelity of our results would be degraded or suffer from visual artifacts (Fig. 9). Furthermore, publicly available T2V models are still in infancy, in terms of

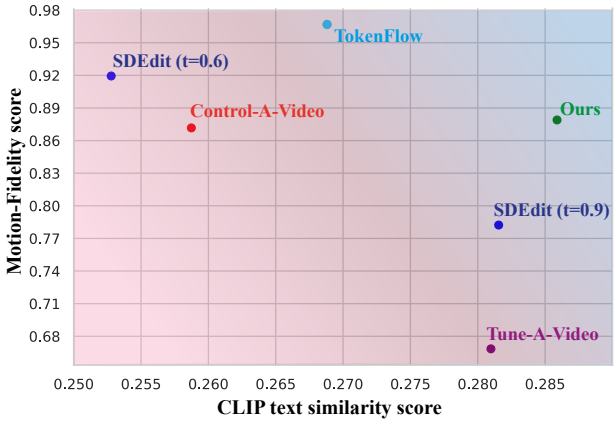


Figure 8. **Quantitative evaluation.** For each baseline, we measure CLIP text similarity (higher is better) and motion fidelity (Eq. 5; higher is better). Our method exhibits a better balance between these two metrics.

Figure 9. **Limitations.** Our method struggles to preserve the original motion since the combination of the original motion and the edit prompt may be out of distribution for the T2V model.

Method	Judgements in our favor (%)
TokenFlow	72.57%
Control-A-Video	84.50%
Tune-A-Video	77.80%

Table 1. **User Study.** We report the percentage of judgments in our favour w.r.t. each baseline.

quality, resolution, video length, and the scale of their training data compared to the vast distribution of natural videos. Despite the limitations of publicly available text-to-video models, our method achieves a significant improvement over prior state-of-the-art methods, demonstrating the potential of leveraging the priors and space-time feature space learned by these models.

7. Acknowledgement

This project received funding from the Israeli Science Foundation (grant 2303/20). We thank GEN-1 authors for their help in conducting comparisons.

References

[1] Guha Balakrishnan, Amy Zhao, Adrian V Dalca, Fredo Durand, and John Guttag. Synthesizing images of humans in unseen poses. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8340–8348, 2018. 2

[2] Omer Bar-Tal, Dolev Ofri-Amar, Rafail Fridman, Yoni Kasten, and Tali Dekel. Text2live: Text-driven layered image and video editing. In *European Conference on Computer Vision*, pages 707–723. Springer, 2022. 3, 8

[3] Omer Bar-Tal, Lior Yariv, Yaron Lipman, and Tali Dekel. Multidiffusion: Fusing diffusion paths for controlled image generation. *arXiv preprint arXiv:2302.08113*, 2023. 5

[4] Andreas Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. Align your latents: High-resolution video synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22563–22575, 2023. 2, 3

[5] Andreas Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. Align your latents: High-resolution video synthesis with latent diffusion models. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 2

[6] Mingdeng Cao, Xintao Wang, Zhongang Qi, Ying Shan, Xiaohu Qie, and Yinqiang Zheng. Masactrl: Tuning-free mutual self-attention control for consistent image synthesis and editing, 2023. 2

[7] Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7291–7299, 2017. 1, 2

[8] cerspense. https://huggingface.co/cerspense/zeroscope_v2_576w, 2023. 2, 3, 12

[9] Duygu Ceylan, Chun-Hao Paul Huang, and Niloy Jyoti Mitra. Pix2video: Video editing using image diffusion. *ArXiv*, abs/2303.12688, 2023. 2

[10] Wenhao Chai, Xun Guo, Gaoang Wang, and Yan Lu. Stablevideo: Text-driven consistency-aware diffusion video editing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 23040–23050, 2023. 3

[11] Caroline Chan, Shiry Ginosar, Tinghui Zhou, and Alexei A Efros. Everybody dance now. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5933–5942, 2019. 1, 2

[12] Hila Chefer, Yuval Alaluf, Yael Vinker, Lior Wolf, and Daniel Cohen-Or. Attend-and-excite: Attention-based semantic guidance for text-to-image diffusion models. *arXiv preprint arXiv:2301.13826*, 2023. 2

[13] Weifeng Chen, Jie Wu, Pan Xie, Hefeng Wu, Jiashi Li, Xin Xia, Xuefeng Xiao, and Liang Lin. Control-a-video: Controllable text-to-video generation with diffusion models, 2023. 2, 6, 7, 8, 12

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

[15] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 2021. 2, 3

[16] Dave Epstein, Allan Jabri, Ben Poole, Alexei A. Efros, and Aleksander Holynski. Diffusion self-guidance for controllable image generation. *arXiv preprint arXiv:2306.00986*, 2023. 4

[17] 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. 2, 6, 7, 8, 12

[18] Songwei Ge, Seungjun Nah, Guilin Liu, Tyler Poon, Andrew Tao, Bryan Catanzaro, David Jacobs, Jia-Bin Huang, Ming-Yu Liu, and Yogesh Balaji. Preserve your own correlation: A noise prior for video diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22930–22941, 2023. 2

[19] Michal Geyer, Omer Bar-Tal, Shai Bagon, and Tali Dekel. Tokenflow: Consistent diffusion features for consistent video editing. *arXiv preprint arxiv:2307.10373*, 2023. 2, 6, 7, 8, 12

[20] 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. 2

[21] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 2020. 2, 3

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

[23] Susung Hong, Gyuseong Lee, Wooseok Jang, and Seungryong Kim. Improving sample quality of diffusion models using self-attention guidance. *arXiv preprint arXiv:2210.00939*, 2022. 2

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

[25] Yasamin Jafarian, Tuanfeng Y Wang, Duygu Ceylan, Jimei Yang, Nathan Carr, Yi Zhou, and Hyun Soo Park. Normal-guided garment uv prediction for human re-texturing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4627–4636, 2023. 3

[26] Nikita Karaev, Ignacio Rocco, Benjamin Graham, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. Cotracker: It is better to track together. *arXiv:2307.07635*, 2023. 7

[27] Yoni Kasten, Dolev Ofri, Oliver Wang, and Tali Dekel. Layered neural atlases for consistent video editing. *ACM Transactions on Graphics (TOG)*, 2021. 3, 6

[28] Levon Khachatryan, Andranik Movsisyan, Vahram Tadevosyan, Roberto Henschel, Zhangyang Wang, Shant Navasardyan, and Humphrey Shi. Text2video-zero: Text-to-image diffusion models are zero-shot video generators. *arXiv preprint arXiv:2303.13439*, 2023. 2

[29] Hyeongwoo Kim, Pablo Garrido, Ayush Tewari, Weipeng Xu, Justus Thies, Matthias Niessner, Patrick Pérez, Christian Richardt, Michael Zollhöfer, and Christian Theobalt. Deep video portraits. *ACM transactions on graphics (TOG)*, 2018. 2

[30] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014. 12

[31] Yao-Chih Lee, Ji-Ze Genevieve Jang, Yi-Ting Chen, Elizabeth Qiu, and Jia-Bin Huang. Shape-aware text-driven layered video editing. *arXiv preprint arXiv:2301.13173*, 2023. 3, 6, 12

[32] Yitong Li, Martin Min, Dinghan Shen, David Carlson, and Lawrence Carin. Video generation from text. In *Proceedings of the AAAI conference on artificial intelligence*, 2018. 2

[33] Ce Liu, Antonio Torralba, William T Freeman, Frédo Durand, and Edward H Adelson. Motion magnification. *ACM transactions on graphics (TOG)*, 24(3):519–526, 2005. 8

[34] Sebastian Loeschcke, Serge Belongie, and Sagie Benaim. Text-driven stylization of video objects. In *European Conference on Computer Vision*, pages 594–609. Springer, 2022. 3

[35] Wan-Duo Kurt Ma, JP Lewis, W Bastiaan Kleijn, and Thomas Leung. Directed diffusion: Direct control of object placement through attention guidance. *arXiv preprint arXiv:2302.13153*, 2023. 2

[36] Aravindh Mahendran and Andrea Vedaldi. Understanding deep image representations by inverting them. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5188–5196, 2015. 4

[37] 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*, 2022. 5, 6, 8

[38] Gaurav Mittal, Tanya Marwah, and Vineeth N Balasubramanian. Sync-draw: Automatic video generation using deep recurrent attentive architectures. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 1096–1104, 2017. 2

[39] Ron Mokady, Rotem Tzaban, Sagie Benaim, Amit H Bermano, and Daniel Cohen-Or. Jokr: Joint keypoint representation for unsupervised cross-domain motion retargeting. *arXiv preprint arXiv:2106.09679*, 2021. 1

[40] 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 preprint arXiv:2302.01329*, 2023. 2

[41] Chong Mou, Xintao Wang, Jiechong Song, Ying Shan, and Jian Zhang. Dragondiffusion: Enabling drag-style manipulation on diffusion models. *arXiv preprint arXiv:2307.02421*, 2023. 4

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

[43] Yingwei Pan, Zhaofan Qiu, Ting Yao, Houqiang Li, and Tao Mei. To create what you tell: Generating videos from captions. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 1789–1798, 2017. 2

[44] Gaurav Parmar, Krishna Kumar Singh, Richard Zhang, Yijun Li, Jingwan Lu, and Jun-Yan Zhu. Zero-shot image-to-image translation. In *ACM SIGGRAPH 2023 Conference Proceedings*, pages 1–11, 2023. 6

[45] Or Patashnik, Daniel Garibi, Idan Azuri, Hadar Averbuch-Elor, and Daniel Cohen-Or. Localizing object-level shape variations with text-to-image diffusion models. *arXiv preprint arXiv:2303.11306*, 2023. 2

[46] Ryan Po, Wang Yifan, Vladislav Golyanik, Kfir Aberman, Jonathan T Barron, Amit H Bermano, Eric Ryan Chan, Tali Dekel, Aleksander Holynski, Angjoo Kanazawa, et al. State of the art on diffusion models for visual computing. *arXiv preprint arXiv:2310.07204*, 2023. 2, 3

[47] 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

[48] Chenyang Qi, Xiaodong Cun, Yong Zhang, Chenyang Lei, Xintao Wang, Ying Shan, and Qifeng Chen. Fatezero: Fusing attentions for zero-shot text-based video editing. *arXiv:2303.09535*, 2023. 2, 8

[49] 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*. PMLR, 2021. 3, 7

[50] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022. 2, 3, 6

[51] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022. 12

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

[53] Aliaksandr Siarohin, Stéphane Lathuilière, Sergey Tulyakov, Elisa Ricci, and Nicu Sebe. Animating arbitrary objects via deep motion transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2377–2386, 2019. 2

[54] Aliaksandr Siarohin, Stéphane Lathuilière, Sergey Tulyakov, Elisa Ricci, and Nicu Sebe. First order motion model for image animation. *Advances in neural information processing systems*, 32, 2019. 2

[55] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013. 4

[56] 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, 2022. 2

[57] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International Conference on Machine Learning*, pages 2256–2265. PMLR, 2015. 3

[58] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2020. 3, 4, 12

[59] Luming Tang, Menglin Jia, Qianqian Wang, Cheng Perng Phoo, and Bharath Hariharan. Emergent correspondence from image diffusion. *arXiv preprint arXiv:2306.03881*, 2023. 4, 12

[60] Jiale Tao, Biao Wang, Borun Xu, Tiezheng Ge, Yuning Jiang, Wen Li, and Lixin Duan. Structure-aware motion transfer with deformable anchor model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3637–3646, 2022. 1

[61] Narek Tumanyan, Omer Bar-Tal, Shai Bagon, and Tali Dekel. Splicing vit features for semantic appearance transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10748–10757, 2022. 4

[62] Narek Tumanyan, Michal Geyer, Shai Bagon, and Tali Dekel. Plug-and-play diffusion features for text-driven image-to image translation. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1921–1930, 2023. 2, 4, 12

[63] 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. 2

[64] Jiuniu Wang, Hangjie Yuan, Dayou Chen, Yingya Zhang, Xiang Wang, and Shiwei Zhang. Modelscope text-to-video technical report. *arXiv preprint arXiv:2308.06571*, 2023. 2, 3, 12

[65] Xiang Wang, Hangjie Yuan, Shiwei Zhang, Dayou Chen, Jiuniu Wang, Yingya Zhang, Yujun Shen, Deli Zhao, and Jingren Zhou. Videocomposer: Compositional video synthesis with motion controllability. *arXiv preprint arXiv:2306.02018*, 2023. 2

[66] Dongxu Wei, Xiaowei Xu, Haibin Shen, and Kejie Huang. C2f-fwn: Coarse-to-fine flow warping network for spatial-temporal consistent motion transfer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 2852–2860, 2021. 2

[67] Olivia Wiles, A Koepke, and Andrew Zisserman. X2face: A network for controlling face generation using images, audio, and pose codes. In *Proceedings of the European conference on computer vision (ECCV)*, pages 670–686, 2018. 2

[68] Chenfei Wu, Jian Liang, Lei Ji, Fan Yang, Yuejian Fang, Daxin Jiang, and Nan Duan. Nüwa: Visual synthesis pre-training for neural visual world creation. In *European conference on computer vision*, pages 720–736. Springer, 2022. 2

[69] 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, 6, 7, 8, 12

[70] Yilun Xu, Mingyang Deng, Xiang Cheng, Yonglong Tian, Ziming Liu, and Tommi S. Jaakkola. Restart sampling for improving generative processes. *CoRR*, abs/2306.14878, 2023. 12

[71] Vickie Ye, Zhengqi Li, Richard Tucker, Angjoo Kanazawa, and Noah Snavely. Deformable sprites for unsupervised video decomposition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2657–2666, 2022. 3

A. Text-to-Video Model Architecture and Feature Selection

Text-to-Video Model. We use ZeroScope [8] text-to-video model, which is claimed to be fine-tuned from a Modelscope model [64] on video clips of the length of 24 frames and 576x320 resolution. Our generated results are in the same resolution with a length of 24 frames. The model was inflated from the StableDiffusion model [51] by introducing temporal layers within each building block of the UNet.

Feature Selection. The decoder of the UNet in ZeroScope comprises four blocks, each with a different resolution. We performed our analysis on coarse features, extracted from the 2nd decoder block of the UNet. We noticed that different coarse features in this block performed similarly for our task. Specifically, we tested intermediate features extracted from the spatial/temporal convolution models, output tokens from the spatial/temporal attention models, as well as features taken directly after the Upsampling block (a.k.a semantic DIFT features[59]). We empirically found that features extracted after the Upsampling block produce more visually appealing edit results.

B. Implementation Details

Feature Extraction. To obtain intermediate latents, we follow [62] and use DDIM inversion (applying DDIM sampling in reverse order) with a classifier-free guidance scale of 1 and 1000 forward steps, using a video-specific inversion prompt. We use these intermediate latents for initialization and extracting diffusion features.

Initialization and Sampling. In our experiments, we use 50 denoising steps using Restart Sampling [70] combined with DDIM sampling [58], with a classifier-free guidance scale of 10. To obtain the initial noise, we apply the downsampling/upsampling operation LF_ξ, described in Eq. 4 with a factor $\xi = 4$.

Optimization details. We apply the optimization described in Sec. 4.2 for the initial 20 denoising steps. In most of our experiments, we are using the Adam optimizer [30] with a learning rate of 0.01 for 30 optimization steps, but in cases where the edit required a bigger deviation from the original structure, we used a linear learning rate decay from 0.005 to 0.002 for 10 optimization steps.

Runtime. The runtime of our method mainly consists of two parts - DDIM inversion, which takes ~10 minutes, and sampling with optimization, which takes ~7 minutes for 10 optimization steps per denoising step and ~15 minutes for 30 optimization steps per denoising step, depending on the configuration.

C. Baseline Comparison Details

For comparing with Tune-A-Video [69], TokenFlow [19] and Control-A-Video [13] we used the official repositories. For visual comparison with Gen-1 [17], we used the publicly available web platform. Since this platform outputs videos of different lengths with some frames being duplicated, we excluded Gen-1 from numerical comparisons. Since SA-NLA [31] takes 10 hours to train, we compare to their provided videos and edit prompts qualitatively.