This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

VideoSwap: Customized Video Subject Swapping with Interactive Semantic Point Correspondence

Yuchao Gu[1,2], Yipin Zhou[2], Bichen Wu[2], Licheng Yu[2], Jia-Wei Liu[1], Rui Zhao[1], Jay Zhangjie Wu[1], David Junhao Zhang[1], Mike Zheng Shou[1]*, Kevin Tang[2]

[1]Show Lab, National University of Singapore [2]GenAI, Meta
https://videoswap.github.io/

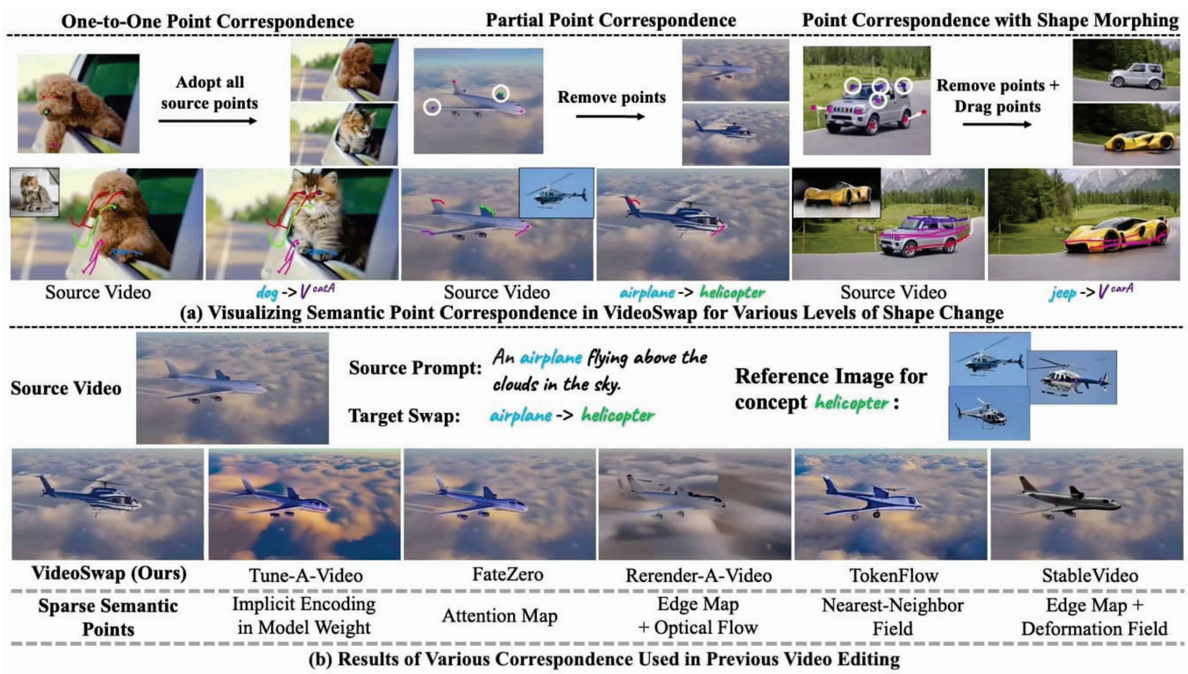

Figure 1. Customized video subject swapping results with VideoSwap. VideoSwap supports shape change in the swapped results while aligning with the source motion trajectory. The swapped target can be either a predefined concept from a pretrained model (*e.g.*, *helicopter*) or a customized concept (denoted by V^*). Previous methods based on implicit motion encoding and dense correspondence do not perform well in subject swapping with shape changes. We encourage readers to click and play the video clips in this figure using Adobe Acrobat.

Abstract

*Current diffusion-based video editing primarily focuses on structure-preserved editing by utilizing various dense correspondences to ensure temporal consistency and motion alignment. However, these approaches are often ineffective when the target edit involves a shape change. To embark on video editing with shape change, we explore customized video subject swapping in this work, where we aim to replace the main subject in a source video with a target subject having a distinct identity and potentially different shape. In contrast to previous methods that rely on dense correspondences, we introduce the VideoSwap framework that exploits semantic point correspondences, inspired by our observation that only a small number of semantic points are necessary to align the subject's motion trajectory and modify its shape. We also introduce various user-point interactions (*e.g.*, *removing points and dragging points) to address various semantic point correspondence. Extensive experiments demonstrate state-of-the-art video subject swapping results across a variety of real-world videos.*

*Corresponding Author

1. Introduction

Diffusion-based video editing [5, 10, 22, 25, 31, 43, 44, 49] is an emerging field that harnesses the capabilities of pre-

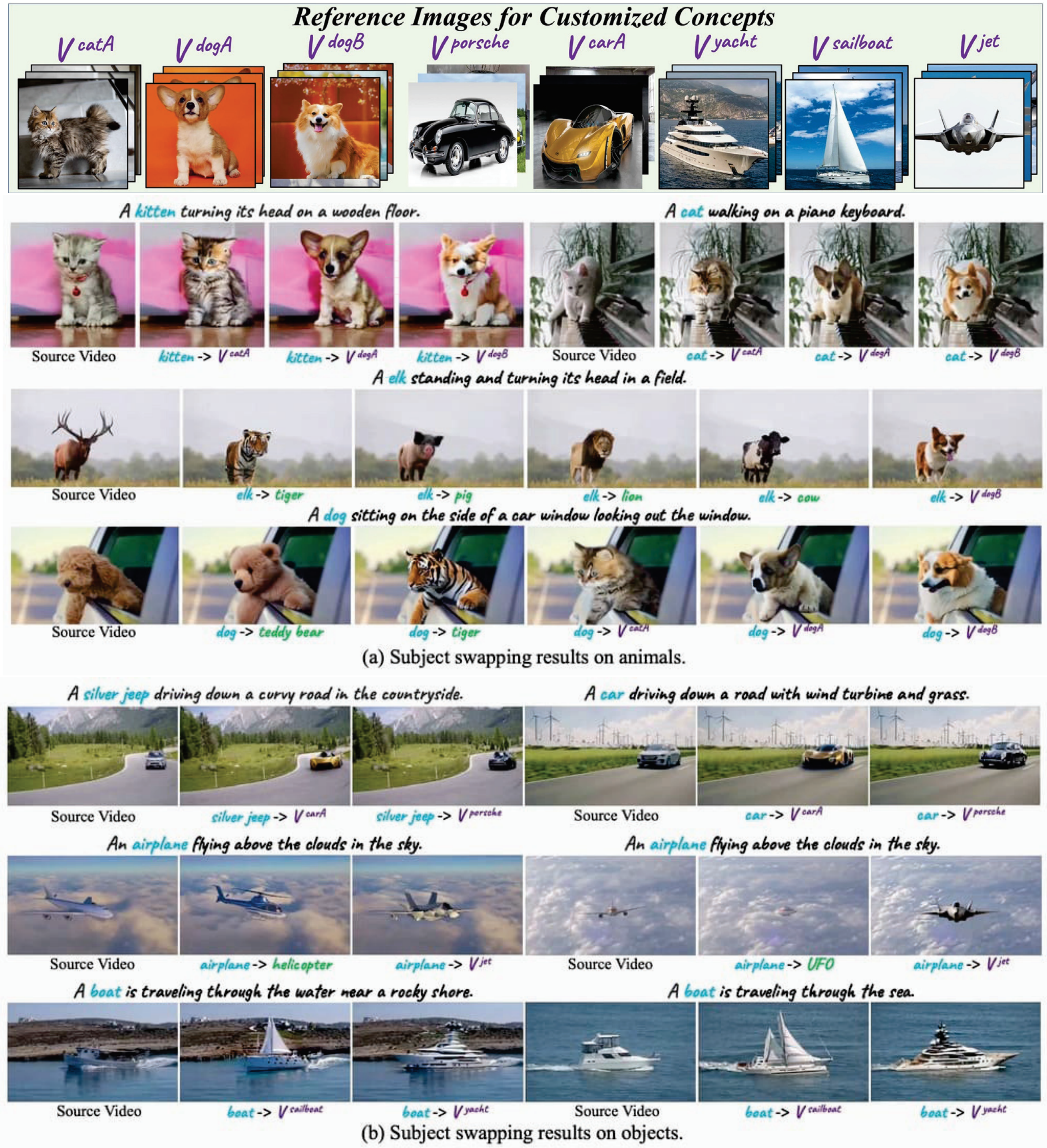

Figure 2. Customized video subject swapping results of VideoSwap on various concepts. The swapping target can either be a predefined concept in the pretrained model (*e.g.*, *helicopter*) or a customized concept created by ED-LoRA [13] (denoted as V^*). We encourage readers to click and play the video clips in this figure using Adobe Acrobat. **For legal issues, we cannot display the human swap results.**

trained text-to-image/video diffusion models [14, 15, 32, 37] to facilitate a range of video editing tasks, including style change and subject/background swapping. The main challenge in video editing lies in how to extract motion from the source video and transfer it to the edited video while ensuring temporal consistency. Pioneer Tune-A-Video [43] implicitly encodes source motion in the diffusion model weights by tuning from the source video. While it demonstrates versatile applications for video editing, the temporal consistency is far from satisfactory. Subsequent works make use of various dense correspondences extracted from the source video, including attention maps [23, 31], edge/depth maps [10, 22, 46, 49], optical flows [46], and deformation fields [5, 6] for video editing. While achieving better temporal consistency, dense correspondence imposes strict shape constraints on the target edit, which makes it ineffective for video editing with shape changes.

To embark on video editing with shape change, we delve into a challenging task: customized video subject swapping. Unlike the conventional video subject swapping addressed in previous works [5, 46, 49], where the swapped subject conforms to the shape of source subject, customized subjects have a clearly defined identity in terms of both appearance and shape. These distinctive characteristics should be preserved in the target edit. Therefore, previous structure-preserved video editing methods are often ineffective for this problem, as shown in Fig. 1(b).

To address customized video subject swapping, our primary insight is that ***the subject's motion trajectory can be effectively described using a small number of semantic points***. As shown in Fig. 1(a), the motion trajectory of an airplane can be depicted by semantic points located at its wings, nose, and tail. This insight naturally leads us to the following question: Can we utilize these semantic points as correspondences to align the motion trajectory while relaxing the shape constraints in video editing? To answer this question, we conduct a toy experiments and observe that it is possible to learn semantic point correspondence for a specific video subject using just a small number of source video frames. Users can interact with learned semantic points to generate unseen poses or modify the shape of the video subject. These observations suggest the potential for integrating semantic point correspondence into video editing, provided that we can obtain an accurate semantic point sequence for the target edit.

To unleash the potential of semantic point correspondence, we introduce the VideoSwap framework for customized video subject swapping, which comprises the following primary designs: 1) Integrating the motion layer into the image diffusion model to ensure essential temporal consistency. 2) Registering semantic points on the source video and utilizing them to transfer the motion trajectory of source subject to the target edit. 3) Introducing user-point interactions (*e.g.*, removing or dragging points) for various semantic point correspondence.

Our contributions are summarized as follows:

- Empirical observations that reveal the potential of semantic point correspondence for aligning motion trajectories and changing shapes in video editing.
- The VideoSwap framework, which minimizes user intervention while unleashing the potential of semantic point correspondence in customized video subject swapping.
- State-of-the-art results in customized video subject swapping, as demonstrated in Fig. 2.

2. Related Work

2.1. Diffusion-Based Video Editing

Structure-Preserved Video Editing. FateZero [31] and Video-P2P [23] extract cross- and self-attention from the source video to control spatial layout. To achieve stricter alignment of temporal consistency with the source video, Rerender-A-Video [46], Gen-1 [10], ControlVideo [49], and TokenFlow [12] extract and align optical flow, depth/edge maps, and nearest-neighbour field from the source video respectively, resulting in improved temporal consistency. StableVideo [5], VidEdit [6], and CoDEF [28] learn the canonical space for editing following the Layered Neural Atlas [19] or the deformation field in Dynamic Nerf [24, 30]. While achieving promising results in structure-preserved video editing, the above methods based on various dense correspondence are not suitable for handling subject swapping involving shape changes.

Video Editing with Shape Change. Tune-A-Video (TAV) [43] and FateZero [31] with TAV checkpoint can be utilized for video editing with shape change, as they implicitly encode the motion in model weights through tuning on the source video. However, they suffer from structure and appearance leakage due to model tuning. Shape-aware editing [21] is built on the Layered Neural Atlas [19], employing semantic correspondences to estimate shape deformation and warp the atlas. Nevertheless, texture warping will lead to unrealistic textures.

Video Diffusion Models. Previous video editing primarily relies on the text-to-image diffusion model [32]. However, recent advancements are occurring in video foundation models [3, 14, 41, 48]. In this work, we add a motion layer [14] to the image diffusion model to provide essential temporal consistency for video editing, and we focus on exploiting semantic point correspondence to align the motion trajectory of the video subject.

2.2. Point Correspondence

Point Correspondence in Diffusion Models. DIFT [39] initially uncovers robust semantic point correspondences in diffusion models. Building upon the observation in DIFT, subsequent works, DragDiffusion [36] and DragonDiffusion [26] extend DragGAN [29] to support interactive point-based image editing in diffusion models. However, point correspondences and interactive drag-based editing are seldom investigated for video editing.

Tracking Any Point in Video. TAP-Vid [8] first introduces the problem of tracking any point (TAP). Unlike optical flow estimation, TAP requires the establishment of long-range correspondence for all points in a video. In our work, we use TAP to reduce human intervention in annotating subject keypoints and acquire long-range motion estimation. Although several works [8, 9, 18, 42] address the TAP problem, we choose to employ Co-Tracker [18], which is the most efficient solution available.

2.3. Concept Customization

Concept customization is mainly categorized into tuning-based approaches [11, 13, 20, 33, 40] and tuning-free solu-

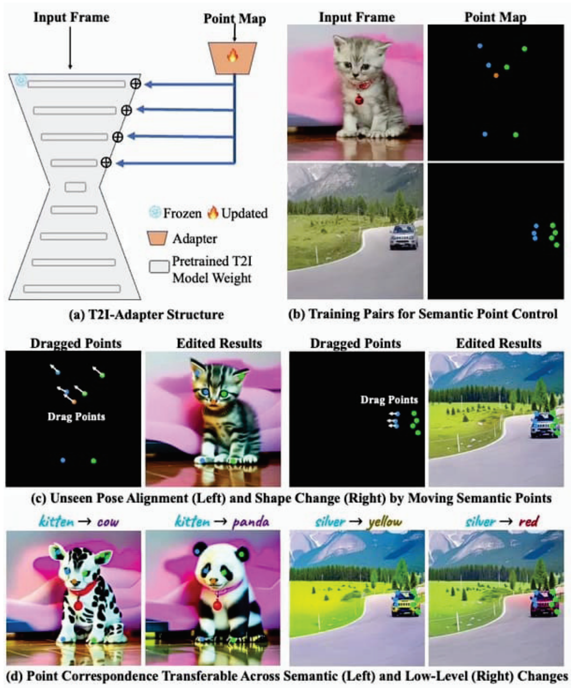

Figure 3. Toy experiment exploring semantic point correspondence. We encourage readers to click and play the video clips in this figure using Adobe Acrobat.

tions [17, 34, 35, 45, 47]. Tuning-free solutions are fast but typically adhere closely to the provided reference image and lack variation. On the other hand, tuning-based solutions can leverage multi-view images to ensure variation in the given concepts and maintain the same inference behavior to pretrained diffusion models. In this paper, we employ the tuning-based ED-LoRA [13] for encoding subject identity.

3. VideoSwap

In this section, we start by presenting a toy experiment to illustrate our motivation to explore semantic point correspondence in Sec. 3.1. Subsequently, we offer an overview of the VideoSwap pipeline for customized video subject swapping in Sec. 3.2. Following that, we explain the process of injecting semantic point correspondence in Sec. 3.3 to Sec. 3.5.

3.1. Motivation

Dense correspondences explored in previous video editing methods [5, 10, 12, 28, 46] restrict the subject's shape change in the edited video. Therefore, our goal is to find a more flexible correspondence that can transfer the source subject's motion trajectory without imposing strict shape constraints. Motivated by this, we investigate sparse semantic points as correspondences. Unlike dense correspondences such as depth, edge, and optical flow, which are low-level cues shared across all video subjects, semantic points vary with different open-world concepts. Therefore, it is not feasible to train a general condition model for injecting semantic point correspondence. Instead, the question we aim to address in this section is, ***is it possible to learn semantic point control for a specific source video subject using only a small number of source video frames***?

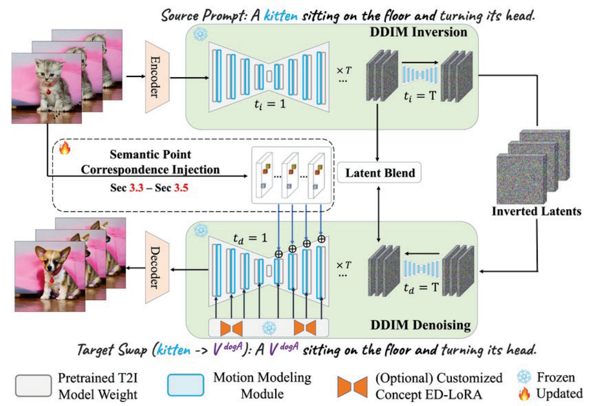

Figure 4. Overview of the VideoSwap pipeline for customized video subject swapping.

Toy Experiment Setting. To address the above question, we perform a toy experiment. Firstly, for a given video, we manually define a set of semantic points. Next, we annotate the same set of points on eight frames of this video. Finally, we train a T2I-Adapter [27] on these data pairs, as illustrated in Fig. 3(a, b), to determine whether these semantic points can be used to control the source video subject.

Observation 1: ***Semantic points optimized on source video frames have the potential to align the subject's motion trajectory and change the subject's shape.*** As shown in Fig. 3(c, left), we drag the points on the cat's face. Despite this edited point map is not in the training data, the resulting image closely follows the adjusted semantic points, effectively generating the unseen pose of the subject. As shown in Fig. 3(c, right), when we drag the boundary semantic points of the car, the edited subject will also reshape to align with the semantic point. This suggests the potential of utilizing semantic points to align the motion trajectory or change the shape when a sequence of dragged points for all video frames is accessible.

Observation 2: ***Semantic points optimized on source video frames can transfer across semantic and low-level changes.*** As demonstrated in Fig. 3(d), when we replace the source subject with different semantics or modify the low-level information in the prompts, the optimized semantic point can also control the pose or shape of the target concept. This suggests that the semantic point can be transferred across both semantic and low-level changes.

3.2. Overview of VideoSwap

3.2.1 Task Formulation

In this paper, we focus on customized video subject swapping, with the goal of ***subject replacement*** and ***background***

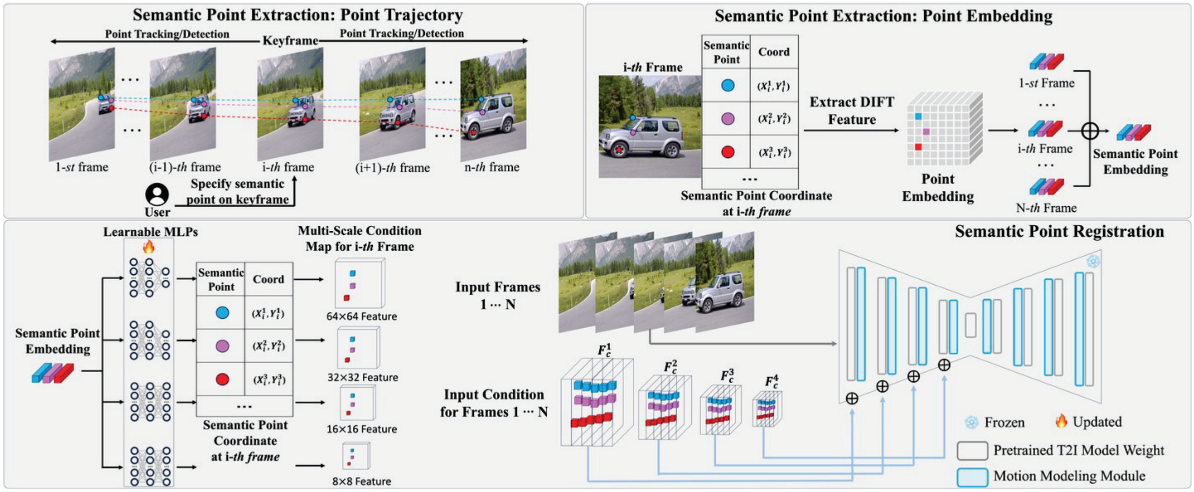

Figure 5. Pipelines for semantic point extraction (Sec. 3.3) and semantic point registration (Sec. 3.4) in VideoSwap. In semantic point extraction, users define semantic points at a keyframe. We then extract the trajectory and embedding of those semantic points from the video. In semantic point registration, the semantic point embedding is projected by multiple 2-layer learnable MLPs, placed in empty features based on their coordinates, and then added element-wise to the diffusion model as motion guidance.

preservation. Subject replacement requires preserving the identity of the target subject in the swapped results, encompassing both its appearance and shape. Simultaneously, background preservation requires the unedited background area to remain the same with the source video. The primary challenge of this task lies in aligning the motion trajectory of the source subject while preserving the identity of the target concept, particularly its shape.

3.2.2 Overall Pipeline

The VideoSwap pipeline is illustrated in Fig. 4. Following the latent diffusion model [32], we encode the source video with a VAE encoder to obtain the latent space representation z_0. Subsequently, DDIM inversion [7, 38] is applied to transform the clean latent z_0 back to the noisy latent z_T. After obtaining the DDIM inverted noise z_T, we replace the source subject in the text prompt with the target subject and denoise it using the DDIM scheduler [38]. In this denoising process, we introduce semantic point correspondence to guide the subject's motion trajectory, as detailed in Sec. 3.3-Sec. 3.5. To preserve the unedited background, we leverage the concept of latent blending [1, 2], further explained *in the supplementary material*. Additionally, we incorporate the following designs:

Adopting Motion Layers. We integrate the motion layers [3, 14] into the image diffusion model to ensure essential temporal consistency for video editing.

Supporting Predefined and Customized Concepts. We support both predefined concepts from the pretrained model and customized concepts. To create customized concepts, we train ED-LoRA [13] on a set of representative images to encode their identity. After training, these concept ED-LoRAs can be used at the inference time.

3.3. Semantic Point Extraction

We first extract the trajectories of semantic points and their associated semantic embeddings from the source video.

Point Trajectory Extraction. As depicted in Fig. 5, for a video containing N frames, users specify K semantic points at a keyframe i. These user-defined semantic points are then propagated to the remaining $N-1$ frames using a point tracker [18] or detector [4]. Subsequently, the motion trajectory of all semantic points in the entire video is obtained and represented as $\mathbf{P}_{coord} = \{Tra(k)|k = 1...K\}$, where $Tra(k) = \{(x_n^k, y_n^k, n)|n = 1...N\}$ represents the motion trajectory of semantic point k across all N frames.

Point Embedding Extraction. To leverage semantic point correspondence, it is crucial to associate each point with its semantics. Specifically, we extract the DIFT [39] feature $\mathbf{D}_n$ for each frame n. Subsequently, the point embedding for semantic point k at frame n is obtained as $\mathbf{v}_n^k = \mathbf{D}_n(x_n^k, y_n^k)$, where (x_n^k, y_n^k) is retrieved from $\mathbf{P}_{coord}$. Following this, we aggregate the point embeddings acquired from all N frames to obtain the final embedding for each semantic point k by $\mathbf{v}^k = \frac{1}{N}\sum_{n=1}^{N} \mathbf{v}_n^k$. Finally, we obtain the semantic embedding for all semantic points, succinctly represented as $\mathbf{P}_{emb} = \{\mathbf{v}^k|k = 1...K\}$.

3.4. Semantic Point Registration

After acquiring the motion trajectory $\mathbf{P}_{coord}$ and the embedding $\mathbf{P}_{emb}$ for semantic points, we register these semantic points on the source video to enable them to offer motion guidance for the video subject.

Sparse Motion Feature Creation. To utilize semantic points as guidance, we generate sparse motion features infused with semantic point embeddings, making them compatible with the Unet encoder. We denote $\mathbf{F}_{enc}$ =

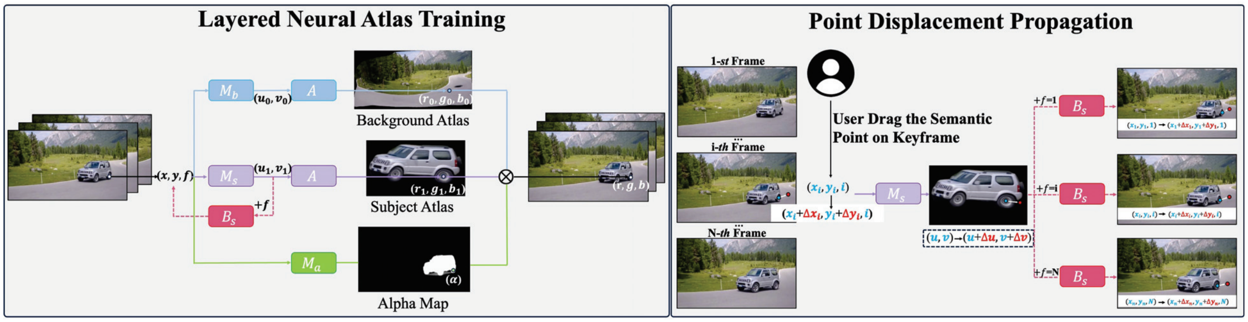

Figure 6. Point displacement propagation based on layered neural atlas (LNA) [16, 19]. Once a trained LNA is provided, users can drag a semantic point at the keyframe, and this displacement is consistently propagated to every frame through the canonical space of the LNA.

$\{\mathbf{F}^1_{enc}, \mathbf{F}^2_{enc}, \mathbf{F}^3_{enc}, \mathbf{F}^4_{enc}\}$ as the multi-scale intermediate feature of the Unet encoder. For an input VAE-encoded latent with spatial-temporal size (H, W, N), the feature size of $\mathbf{F}^l_{enc}$ for each Unet stage $l \in [1, 4]$ can be computed as $(H/2^l, W/2^l, N)$.

As depicted in Fig. 5 and detailed in Algorithm. 1, we create a series of multi-scale conditional features $\mathbf{F}_c = \{\mathbf{F}^1_c, \mathbf{F}^2_c, \mathbf{F}^3_c, \mathbf{F}^4_c\}$. Notably, $\mathbf{F}_c$ shares the same size as $\mathbf{F}_{enc}$ and is initialized with zero vectors. And we introduce a series of learnable MLPs $\phi = \{\phi^l | l \in \{1, 2, 3, 4\}\}$, each of ϕ^l project the point embedding to match the feature dimension of the corresponding $\mathbf{F}^l_c$. Then, for each point $(x^k_n, y^k_n, n) \in \mathbf{P}_{coord}$, we compute its corresponding spatial-temporal coordinate (x, y, n) at l-th Unet stage and assign the projected embedding based on the coordinate by $\mathbf{F}^l_c(x, y, n) = \phi^l(\mathbf{P}_{emb}(k))$.

It is crucial to emphasize that the motion feature $\mathbf{F}_c$ demonstrates high sparsity, with only the semantic point trajectories containing the feature embeddings. Finally, $\mathbf{F}_c$ are added element-wise into the intermediate feature $\mathbf{F}_{enc}$ of Unet encoder as motion guidance:

$$\mathbf{F}^l_{enc} = \mathbf{F}^l_{enc} + \mathbf{F}^l_c, l \in \{1, 2, 3, 4\}. \tag{1}$$

Algorithm 1: Sparse Motion Feature Creation

Input:
1. Point Trajectory $\mathbf{P}_{coord}$,
2. Point Embedding $\mathbf{P}_{emb}$,
3. Learnable MLPs $\phi = \{\phi^l | l \in \{1, 2, 3, 4\}\}$

Output:
Motion Feature $\mathbf{F}_c = \{\mathbf{F}^l_c | l \in \{1, 2, 3, 4\}\}$

```
Initialize: F_c = {F^l_c = 0 | l ∈ {1, 2, 3, 4}};
for l ← 1 to 4 do
    foreach (x^k_n, y^k_n, n) in P_coord do
        x, y = round(x^k_n / 2^l), round(y^k_n / 2^l);
        F^l_c(x, y, n) = φ^l(P_emb(k))
    end
end
```

Semantic Point Registration on Source Video. Our objective is to optimize the projection MLPs (ϕ) to facilitate better motion guidance from semantic points. This optimization objective is defined as

$$\min_{\phi} E_{\epsilon\sim N(0,I),t\sim U(T_{min},T)}||[\epsilon-\epsilon_\theta(z_t,t,p,\phi(\mathbf{P}_{emb}))]\odot\Omega(\mathbf{P}_{coord})||^2_2, \tag{2}$$

where p represents the embedding for the text prompt, and $\Omega(\mathbf{P}_{coord})$ denotes the binary mask that only turns on around the semantic point.

We adopt two techniques to enhance the learning of semantic point correspondences in Eq. (2). The first technique is ***semantic-enhanced schedule***, controlled by T_{min}. We set $T_{min} = T/2$ to enhance the learning at the higher timestep, which prevents overfitting to low-level details and improves semantic point alignment. The second technique, ***point patch loss***, constrains the computation of the loss to a local patch near each semantic point, which reduces structure leakage into the target swap. This is implemented by the a loss mask Ω in Eq. (2).

3.5. User-Point Interaction at Inference Time

In Sec. 3.3, users define semantic points at the source video keyframe, and we subsequently extract their trajectory and semantic embedding in the video. To utilize these semantic points as correspondences, we register them on the source video in Sec. 3.4. Following these two steps, these semantic points become applicable for controlling the motion of the target object. In this section, we introduce user-point interaction to address various semantic point correspondence.

Adopting Source Point Sequence. If there exist one-to-one semantic point correspondence between the source subject and the target subject, such as swapping the *dog* with V^{catA} as illustrated in Fig. 1, we directly use the source point sequence as the motion guidance.

Removing Parts of Semantic Point. When there only exists partial semantic point correspondence between the source subject and the target subject, we can remove redundant points to loosen shape constraints. For example, in scenarios such as swapping an *airplane* for a *helicopter*, as depicted in Fig. 1, we remove semantic points associated with the airplane's wings, given that helicopters typically lack wings. The remaining semantic points on the airplane's nose and tail are retained as motion guidance.

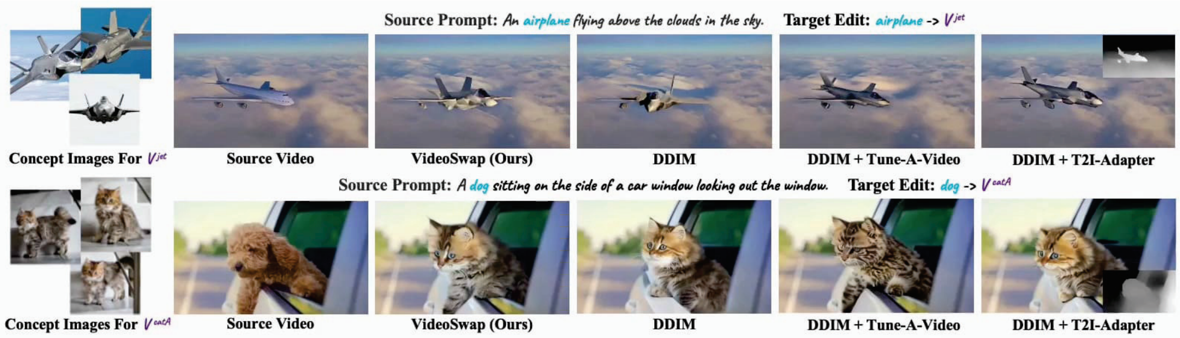

Figure 7. Comparison of VideoSwap with several baselines built upon the same foundational model. The only difference lies in adopting different motion guidance. We encourage readers to click and play the video clips in this figure using Adobe Acrobat.

Dragging of Semantic Point. In situations where there exists semantic point correspondence between the source and target subjects, but misalignment occurs due to shape morphing, we provide users the option to manually drag the semantic points on a keyframe for better alignment of the shape changes. For instance, when swapping a *jeep* (tall and narrow) for a $V_{car\,A}$ (sports car, low and wide) as illustrated in Fig. 1, users can drag the semantic points to accurately reflect the shape change.

Editing the shape change by dragging semantic points on a single frame is straightforward. However, consistently propagating these semantic point displacement over time is non-trivial, mainly because of the complex camera and subject motion in the video. Therefore, we introduce point displacement propagation to solve this problem.

As illustrated in Fig. 6, we follow the Layered Neural Atlas (LNA) [16, 19] to learn the canonical space, as detailed *in the supplementary material*. Following LNA [16, 19], we establish a forward coordinate mapping from the video to the canonical space, denoted as M: $(x, y, f) \rightarrow (u, v)$, along with its corresponding backward mapping B: $(u, v, f) \rightarrow (x, y)$.

Given a semantic point, with coordinates at the keyframe f_{key} represented as (x, y, f_{key}), its trajectory over time can be expressed as a function of time f: $(x(f), y(f)) = P(f)$. Suppose a user drag it to a new position at $(x + dx, y + dy, f_{key})$, we aim to estimate the edited trajectory $P'(f)$ for $f = \{0, ..., N\}$. We resort to LNA's representation, and first compute a linearized estimation of its shifted position on the canonical coordinate:

$$[du, dv]^T = J_M(x, y, f_{key})[dx, dy]^T, \quad (3)$$

where J_M denote the Jacobian matrix with respect to (x, y). Next, at a given time f, we estimate the edited coordinate in the pixel space as

$$P'(f) = P(f) + J_B(u, v, f)[du, dv]^T, \quad (4)$$

where $(u, v) = B(x, y, f_{key})$ and J_B denote the Jacobian matrix with respect to (u, v). We then use this edited trajectory $P'(f)$ for the dragged semantic point during inference.

	Methods/Metrics	Subject Identity	Motion Alignment	Temporal Consistency	Overall Preference
	Compare to Previous Video Editing Methods				
VideoSwap v.s.	Tune-A-Video	**80%** v.s. 20%	**72%** v.s. 28%	**80%** v.s. 20%	**78%** v.s. 22%
	FateZero	**74%** v.s. 26%	**67%** v.s. 33%	**73%** v.s. 27%	**70%** v.s. 30%
	Text2Video-Zero	**76%** v.s. 24%	**71%** v.s. 29%	**80%** v.s. 20%	**80%** v.s. 20%
	Rerender-A-Video	**81%** v.s. 19%	**72%** v.s. 28%	**84%** v.s. 16%	**84%** v.s. 16%
	Compare to Baselines on AnimateDiff				
	w/ DDIM	**70%** v.s. 30%	**74%** v.s. 26%	**69%** v.s. 31%	**73%** v.s. 27%
	w/ DDIM+TAV	**68%** v.s. 32%	**60%** v.s. 40%	**66%** v.s. 34%	**69%** v.s. 31%
	w/ DDIM+T2I-Adapter	**66%** v.s. 34%	**59%** v.s. 41%	**65%** v.s. 35%	**66%** v.s. 34%

Table 1. Human Evaluation on Video Subject Swapping Results.

4. Experiments

We implement our method using the Latent Diffusion Model [32] and adopt the motion layer in AnimateDiff [14] as the foundational model. All videos consist of 16 frames. Additional implementation details are included *in the supplementary material*.

4.1. Qualitative Comparison

Comparison with the State-of-the-Art. We qualitatively compare to Tune-A-Video [43], FateZero [31], Rerender-A-Video [46], TokenFlow [12] and StableVideo [5] in Fig. 1. Previous methods are less effective in revealing the correct shape of the target subject. Compared to them, VideoSwap can achieve a significant shape change while aligning the source motion trajectory.

Comparison with Baselines on AnimateDiff. As most state-of-the-art methods are based on image diffusion models, we also compare VideoSwap to several baselines on the AnimateDiff. The only distinctions from VideoSwap lie in different motion guidance, as shown in the results in Fig. 7:

- DDIM: The DDIM sampling without other motion guidance cannot produce the correct motion trajectory.
- DDIM+Tune-A-Video: If we tune the model as [43] to inject source motion, it achieves correct motion but suffers from severe structure and appearance leakage.
- DDIM+T2I-Adapter: If we add spatial controls [27], such as depth, to control the editing, we observe that 1) the shape is restricted by the source, and 2) the deformable motion cannot follow the source video.

Compared to all constructed baselines, our VideoSwap with semantic point correspondence can effectively align the motion trajectory while preserving the target concept's identity.

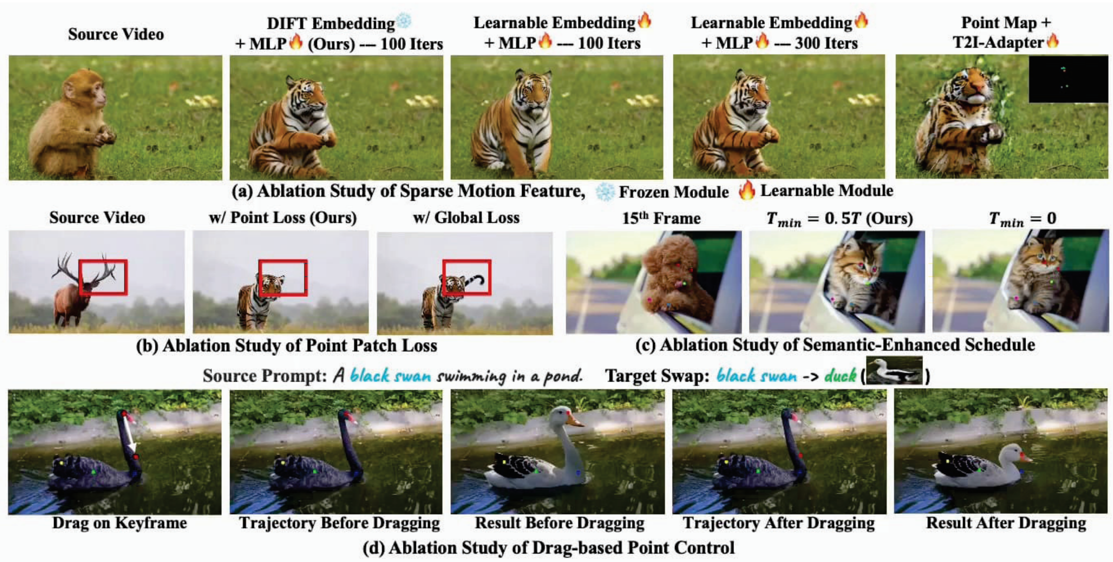

Figure 8. Ablation Study of our VideoSwap. We encourage readers to click and play the video clips in this figure using Adobe Acrobat.

4.2. Quantitative Comparison

We conduct both automatic and human evaluations to quantitatively compare VideoSwap with previous state-of-the-art methods and several baselines on AnimateDiff. Detailed evaluation settings and automatic evaluation results are provided *in the supplementary material*. For human evaluation, we distribute 1000 questionnaires on Amazon Mturk to assess various criteria in customized video subject swapping. From the human evaluation results in Table. 1, our method achieves a clear advantage over the compared methods.

4.3. Ablation Study

Sparse Motion Feature. There are several variants for encoding semantic points to generate motion guidance. The most straightforward approach is to encode the point map of semantic points using T2I-Adapter [27]. However, this method leads to severe overfitting, as illustrated in Fig. 8(a). The issue arises because the encoded feature added to the diffusion model is non-sparse, and the background also contains features that may overfit to the source video. We use MLPs to encode point embeddings, positioning them in the empty feature to create sparse motion features for guidance without compromising video quality. Regarding point embeddings, we opt for DIFT embeddings, which inherently carry robust semantics. Compared to randomly initialized learnable embeddings, our approach achieves similar results with 3× less time for point registration step.

Point Patch Loss. In the semantic point registration, we employ a point patch loss to reconstruct the local patch surrounding each semantic point. If we omit the point patch loss and opt to directly reconstruct the entire video, we observe that the source structure leaks into the target swapped results and thus produce artifacts, as depicted in Fig. 8(b).

Semantic-Enhanced Schedule. Our goal is to employ semantic points to transfer motion trajectories, acting as a linkage between the source and target subjects. Therefore, we aim for the semantic points to emphasize high-level semantic alignment without transferring low-level details. This objective is achieved by registering points only on the early sampling steps, *i.e.*, $T_{\min} = \frac{T}{2}$ in Eq. (2). As shown in Fig. 8(c), this technique prevents the model from learning excessive low-level details and enhances point alignment.

Drag-based Point Control. As illustrated in Fig. 8(d), our goal is to swap the *black swan* to the *duck*. If we directly use the source point sequence as guidance, the *duck*'s neck conforms to the shape of the *black swan*, resulting in an inferior identity. However, by employing the proposed point displacement propagation, we can drag the semantic point at the keyframe, ensuring a consistent motion trajectory after dragging. Utilizing the dragged point trajectory as motion guidance results in the correct identity of the *duck*.

5. Conclusion

This paper uncovers the potential of semantic point correspondence in aligning motion trajectories and altering the subject's identity in video editing. From there, we present VideoSwap, a framework that minimizes human intervention while utilizing semantic point correspondences for customized video subject swapping. Through user-point interactions like point removal or dragging, we address various semantic point correspondence. VideoSwap facilitates shape changes in the target swap while aligning the motion trajectory with the source subject, demonstrating state-of-the-art results in customized video subject swapping.

Acknowledgement This project is supported by the National Research Foundation, Singapore under its NRFF Award NRF-NRFF13-2021-0008.

References

[1] Omri Avrahami, Dani Lischinski, and Ohad Fried. Blended diffusion for text-driven editing of natural images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18208–18218, 2022. 5

[2] Omri Avrahami, Ohad Fried, and Dani Lischinski. Blended latent diffusion. *ACM Trans. Graph.*, 42(4), 2023. 5

[3] 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. 3, 5

[4] 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. 5

[5] Wenhao Chai, Xun Guo, Gaoang Wang, and Yan Lu. Stablevideo: Text-driven consistency-aware diffusion video editing. *arXiv preprint arXiv:2308.09592*, 2023. 1, 2, 3, 4, 7

[6] Paul Couairon, Clément Rambour, Jean-Emmanuel Haugeard, and Nicolas Thome. Videdit: Zero-shot and spatially aware text-driven video editing. *arXiv preprint arXiv:2306.08707*, 2023. 2, 3

[7] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021. 5

[8] Carl Doersch, Ankush Gupta, Larisa Markeeva, Adrià Recasens, Lucas Smaira, Yusuf Aytar, João Carreira, Andrew Zisserman, and Yi Yang. Tap-vid: A benchmark for tracking any point in a video. *Advances in Neural Information Processing Systems*, 35:13610–13626, 2022. 3

[9] Carl Doersch, Yi Yang, Mel Vecerik, Dilara Gokay, Ankush Gupta, Yusuf Aytar, Joao Carreira, and Andrew Zisserman. Tapir: Tracking any point with per-frame initialization and temporal refinement. *arXiv preprint arXiv:2306.08637*, 2023. 3

[10] Patrick Esser, Johnathan Chiu, Parmida Atighehchian, Jonathan Granskog, and Anastasis Germanidis. Structure and content-guided video synthesis with diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7346–7356, 2023. 1, 2, 3, 4

[11] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion. *arXiv preprint arXiv:2208.01618*, 2022. 3

[12] Michal Geyer, Omer Bar-Tal, Shai Bagon, and Tali Dekel. Tokenflow: Consistent diffusion features for consistent video editing. *arXiv preprint arXiv:2307.10373*, 2023. 3, 4, 7

[13] Yuchao Gu, Xintao Wang, Jay Zhangjie Wu, Yujun Shi, Yunpeng Chen, Zihan Fan, Wuyou Xiao, Rui Zhao, Shuning Chang, Weijia Wu, et al. Mix-of-show: Decentralized low-rank adaptation for multi-concept customization of diffusion models. *arXiv preprint arXiv:2305.18292*, 2023. 2, 3, 4, 5

[14] Yuwei Guo, Ceyuan Yang, Anyi Rao, Yaohui Wang, Yu Qiao, Dahua Lin, and Bo Dai. Animatediff: Animate your personalized text-to-image diffusion models without specific tuning. *arXiv preprint arXiv:2307.04725*, 2023. 2, 3, 5, 7

[15] 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

[16] Jiahui Huang, Leonid Sigal, Kwang Moo Yi, Oliver Wang, and Joon-Young Lee. Inve: Interactive neural video editing. *arXiv preprint arXiv:2307.07663*, 2023. 6, 7

[17] Xuhui Jia, Yang Zhao, Kelvin CK Chan, Yandong Li, Han Zhang, Boqing Gong, Tingbo Hou, Huisheng Wang, and Yu-Chuan Su. Taming encoder for zero fine-tuning image customization with text-to-image diffusion models. *arXiv preprint arXiv:2304.02642*, 2023. 4

[18] Nikita Karaev, Ignacio Rocco, Benjamin Graham, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. Cotracker: It is better to track together. *arXiv preprint arXiv:2307.07635*, 2023. 3, 5

[19] Yoni Kasten, Dolev Ofri, Oliver Wang, and Tali Dekel. Layered neural atlases for consistent video editing. *ACM Transactions on Graphics (TOG)*, 40(6):1–12, 2021. 3, 6, 7

[20] Nupur Kumari, Bingliang Zhang, Richard Zhang, Eli Shechtman, and Jun-Yan Zhu. Multi-concept customization of text-to-image diffusion. *arXiv preprint arXiv:2212.04488*, 2022. 3

[21] Yao-Chih Lee, Ji-Ze Genevieve Jang, Yi-Ting Chen, Elizabeth Qiu, and Jia-Bin Huang. Shape-aware text-driven layered video editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14317–14326, 2023. 3

[22] Jun Hao Liew, Hanshu Yan, Jianfeng Zhang, Zhongcong Xu, and Jiashi Feng. Magicedit: High-fidelity and temporally coherent video editing. *arXiv preprint arXiv:2308.14749*, 2023. 1, 2

[23] Shaoteng Liu, Yuechen Zhang, Wenbo Li, Zhe Lin, and Jiaya Jia. Video-p2p: Video editing with cross-attention control. *arXiv preprint arXiv:2303.04761*, 2023. 2, 3

[24] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 3

[25] 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. 1

[26] 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. 3

[27] Chong Mou, Xintao Wang, Liangbin Xie, Jian Zhang, Zhongang Qi, Ying Shan, and Xiaohu Qie. T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models. *arXiv preprint arXiv:2302.08453*, 2023. 4, 7, 8

[28] Hao Ouyang, Qiuyu Wang, Yuxi Xiao, Qingyan Bai, Juntao Zhang, Kecheng Zheng, Xiaowei Zhou, Qifeng Chen, and Yujun Shen. Codef: Content deformation fields for temporally consistent video processing. *arXiv preprint arXiv:2308.07926*, 2023. 3, 4

[29] Xingang Pan, Ayush Tewari, Thomas Leimkühler, Lingjie Liu, Abhimitra Meka, and Christian Theobalt. Drag your gan: Interactive point-based manipulation on the generative image manifold. In *ACM SIGGRAPH 2023 Conference Proceedings*, pages 1–11, 2023. 3

[30] Albert Pumarola, Enric Corona, Gerard Pons-Moll, and Francesc Moreno-Noguer. D-nerf: Neural radiance fields for dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10318–10327, 2021. 3

[31] 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 preprint arXiv:2303.09535*, 2023. 1, 2, 3, 7

[32] 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, 5, 7

[33] 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

[34] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Wei Wei, Tingbo Hou, Yael Pritch, Neal Wadhwa, Michael Rubinstein, and Kfir Aberman. Hyperdreambooth: Hypernetworks for fast personalization of text-to-image models. *arXiv preprint arXiv:2307.06949*, 2023. 4

[35] Jing Shi, Wei Xiong, Zhe Lin, and Hyun Joon Jung. Instantbooth: Personalized text-to-image generation without test-time finetuning. *arXiv preprint arXiv:2304.03411*, 2023. 4

[36] Yujun Shi, Chuhui Xue, Jiachun Pan, Wenqing Zhang, Vincent YF Tan, and Song Bai. Dragdiffusion: Harnessing diffusion models for interactive point-based image editing. *arXiv preprint arXiv:2306.14435*, 2023. 3

[37] 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. 2

[38] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 5

[39] Luming Tang, Menglin Jia, Qianqian Wang, Cheng Perng Phoo, and Bharath Hariharan. Emergent correspondence from image diffusion. *arXiv preprint arXiv:2306.03881*, 2023. 3, 5

[40] Andrey Voynov, Qinghao Chu, Daniel Cohen-Or, and Kfir Aberman. $p+$: Extended textual conditioning in text-to-image generation. *arXiv preprint arXiv:2303.09522*, 2023. 3

[41] 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. 3

[42] Qianqian Wang, Yen-Yu Chang, Ruojin Cai, Zhengqi Li, Bharath Hariharan, Aleksander Holynski, and Noah Snavely. Tracking everything everywhere all at once. *arXiv preprint arXiv:2306.05422*, 2023. 3

[43] Jay Zhangjie Wu, Yixiao Ge, Xintao Wang, Stan Weixian Lei, Yuchao Gu, Yufei Shi, 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. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7623–7633, 2023. 1, 2, 3, 7

[44] Jay Zhangjie Wu, Xiuyu Li, Difei Gao, Zhen Dong, Jinbin Bai, Aishani Singh, Xiaoyu Xiang, Youzeng Li, Zuwei Huang, Yuanxi Sun, et al. Cvpr 2023 text guided video editing competition. *arXiv preprint arXiv:2310.16003*, 2023. 1

[45] Guangxuan Xiao, Tianwei Yin, William T Freeman, Frédo Durand, and Song Han. Fastcomposer: Tuning-free multi-subject image generation with localized attention. *arXiv preprint arXiv:2305.10431*, 2023. 4

[46] Shuai Yang, Yifan Zhou, Ziwei Liu, and Chen Change Loy. Rerender a video: Zero-shot text-guided video-to-video translation. *arXiv preprint arXiv:2306.07954*, 2023. 2, 3, 4, 7

[47] Hu Ye, Jun Zhang, Sibo Liu, Xiao Han, and Wei Yang. Ip-adapter: Text compatible image prompt adapter for text-to-image diffusion models. *arXiv preprint arXiv:2308.06721*, 2023. 4

[48] David Junhao Zhang, Jay Zhangjie Wu, Jia-Wei Liu, Rui Zhao, Lingmin Ran, Yuchao Gu, Difei Gao, and Mike Zheng Shou. Show-1: Marrying pixel and latent diffusion models for text-to-video generation. *arXiv preprint arXiv:2309.15818*, 2023. 3

[49] Min Zhao, Rongzhen Wang, Fan Bao, Chongxuan Li, and Jun Zhu. Controlvideo: Adding conditional control for one shot text-to-video editing. *arXiv preprint arXiv:2305.17098*, 2023. 1, 2, 3