Track4Gen: Teaching Video Diffusion Models to Track Points Improves Video Generation

Hyeonho Jeong[1,2,*] Chun-Hao P. Huang[1] Jong Chul Ye[2] Niloy J. Mitra[1,3] Duygu Ceylan[1]
[1]Adobe Research [2]KAIST [3]University College London

Abstract

While recent foundational video generators produce visually rich output, they still struggle with appearance drift, where objects gradually degrade or change inconsistently across frames, breaking visual coherence. We hypothesize that this is because there is no explicit supervision in terms of spatial tracking at the feature level. We propose Track4Gen, a spatially aware video generator that combines video diffusion loss with point tracking across frames, providing enhanced spatial supervision on the diffusion features. Track4Gen merges the video generation and point tracking tasks into a single network by making minimal changes to existing video generation architectures. Using Stable Video Diffusion [4] as a backbone, Track4Gen demonstrates that it is possible to unify video generation and point tracking, which are typically handled as separate tasks. Our extensive evaluations show that Track4Gen effectively reduces appearance drift, resulting in temporally stable and visually coherent video generation. Project page: hyeonho99.github.io/track4gen

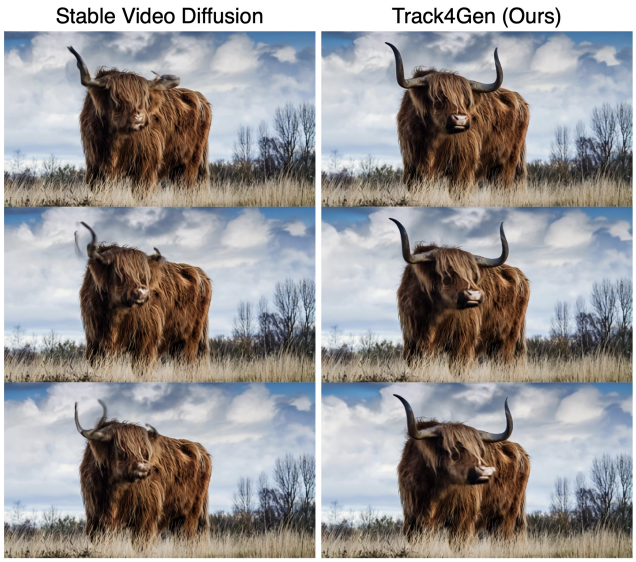


Figure 1. **Motivation.** Videos generated by Stable Video Diffusion [4] suffer from appearance drift, while those from our method, Track4Gen, are free from such appearance inconsistency issues.

1. Introduction

Diffusion-based video generators [4, 6, 47] are making rapid strides in creating temporally consistent and visually rich video content. This progress marks a significant shift, as the unification of generation and control has the potential to transform the traditional workflow of first capturing and then digitally editing video.

Despite impressive capabilities, video generators often suffer from *appearance drift*, where visual elements gradually change, mutate, or degrade over time, causing inconsistencies in the objects. For example, in Fig. 1, we observe the horns of the cow distorting and morphing unrealistically over time, breaking the plausibility of the generated content. This is in striking contrast to humans, who develop a sense of *appearance constancy* as early as infancy through observation and interaction with the world [72].

Unfortunately, appearance drift remains a persistent issue in current video models, even with increased training data and more advanced architectures. We speculate that this limitation arises from supervision being based solely on video diffusion loss (i.e., denoising score matching [64]) in the pixel/latent space, without explicit spatial awareness guidance in the feature space. Hence, in this paper, we ask *if and how we can empower video diffusion models with appearance constancy by providing additional supervision*.

We present *Track4Gen* as a spatially aware video generator that receives supervision both in terms of the original diffusion-based objective as well as (dense) point correspondence across frames, which we refer to as *tracks*. We demonstrate that it is possible to provide such track-level supervision in the diffusion feature space by making minimal architecture changes. Our generated videos do not suffer from degradation of video quality (according to the usual video generation metrics), while being significantly more

*Work done during internship at Adobe.

spatially coherent as the highlight cow in Fig. 1.

We train Track4Gen using the latest Stable Video Diffusion [4] as the backbone and evaluate on the publicly available VBench dataset [30]. We report significant improvement in terms of appearance constancy of subjects, both in quantitative and qualitative (i.e., via user studies) evaluations. In summary, we demonstrate that it is possible to upgrade existing video generators, by supervising them with additional correspondence tracking loss, to produce videos without significant appearance drifts, a problem commonly encountered in diffusion-based video generators.

2. Related Work

Diffusion-based video generation. Building on the success of diffusion models in image synthesis [11, 51], diffusion-based video generators have seen significant advancements [4, 6, 29, 47]. A commonly adopted approach is to extend text-to-image models to the video domain by incorporating temporal layers to facilitate interactions across video frames [5, 22, 54]. While some works have adopted cascaded approaches to produce both spatially and temporally high-resolution videos [28, 47, 54, 67, 76, 78], others have utilized lower-dimensional latent space modeling to reduce computational demands [5, 8, 24, 80]. We build on top of one such approach, Stable Video Diffusion (SVD, [4]), which introduces a latent image-to-video diffusion model trained on a large-scale and curated video data.

With advances in generation, systematic evaluation of generation quality has become crucial. Traditionally, metrics such as Fréchet Inception Distance (FID, [26]), Fréchet Video Distance (FVD, [63]), and CLIPSIM [49] are used. Additionally, comprehensive benchmark suites [30, 68] have been introduced to provide a more robust evaluation aligned with human perception. Inspired by such work, we thoroughly evaluate our approach and demonstrate improved video generation quality with respect to both conventional metrics and the recent VBench metrics [30].

Foundational models as feature extractors. Various foundational models such as vision transformers [15] or diffusion-based generators [50] have been utilized as feature extractors for various tasks including semantic matching [16, 25, 41], classification [38], segmentation [66, 71], and editing [19, 21, 61]. There have been efforts to boost their performance by post-processing the feature maps obtained from the pre-trained models, e.g., by upsampling [18, 58]. In a recent effort, Yue et al. [75] lift semantic per-frame features from a foundational model into a 3D Gaussian representation. They fine-tune the foundational model with such 3D-aware features resulting in improved performance in downstream tasks. Similarly, Sundaram et al. [57] fine-tune state-of-the-art foundational models on human similarity judgments yielding improved representations across downstream tasks. In a concurrent effort, Yu et al. [74] propose to align the internal features of an image generation model with external discriminative features [45], which results in more effective training of the generator.

Our work also enhances the internal feature representation of a foundational generation model but with significant differences compared to previous literature. First, unlike most previous work that focus on image level foundational models, we exploit the power of recently emerging video models. Second, instead of post-processing, we enhance the spatial awareness of the intermediate features by training the generator to jointly perform an additional tracking task. We show that this joint training boosts the performance of intermediate features in correspondence tracking, leading to improved video generation quality.

Tracking any point in a video. The task involves following any arbitrary query point across a long video sequence. First introduced by PIPs [23] and later re-framed by TAP-Vid [12], several methods have emerged in recent years to tackle long-term point tracking. PIPs [23] revisits the classical particle-based representation [53] and introduces MLP-based networks that predict point tracks within an 8-frame window. Subsequent works have improved performance by capturing longer temporal context through advanced architectures [2, 13, 23, 33], as well as by enabling the simultaneous tracking of multiple queries [10, 33]. More recent training-based trackers [10, 34, 40, 70] have achieved remarkable performance by leveraging high-capacity neural networks to learn robust priors from large-scale training data. While high-quality data is crucial for accurate tracking, manually annotating point tracks is prohibitively expensive. Hence, synthetic videos [20] with automatic annotations, have become an alternative and have demonstrated effectiveness in real-world video tracking. An alternative approach is self-supervised adaptation at test time, where tracking is learned without ground-truth labels [32, 62, 65]. In a recent study, Aydemir et al. [1] evaluate the effectiveness of several image foundational model features for point tracking both in zero-shot setting as well as with supervised training using low-rank adapter layers. To the best of our knowledge, we are the first to exploit the features of a foundational video diffusion model for dense point tracking.

3. Method

In this section, we provide a comprehensive discussion of the Track4Gen framework. We begin with a concise overview of latent video diffusion models (Sec. 3.1). Next, we discuss how video diffusion features relate to temporal correspondences both for real and generated videos (Sec. 3.2). Finally, we detail the design of Track4Gen both in terms of network architecture and the employed supervision signals (Sec. 3.3). An overview is depicted in Fig. 2.

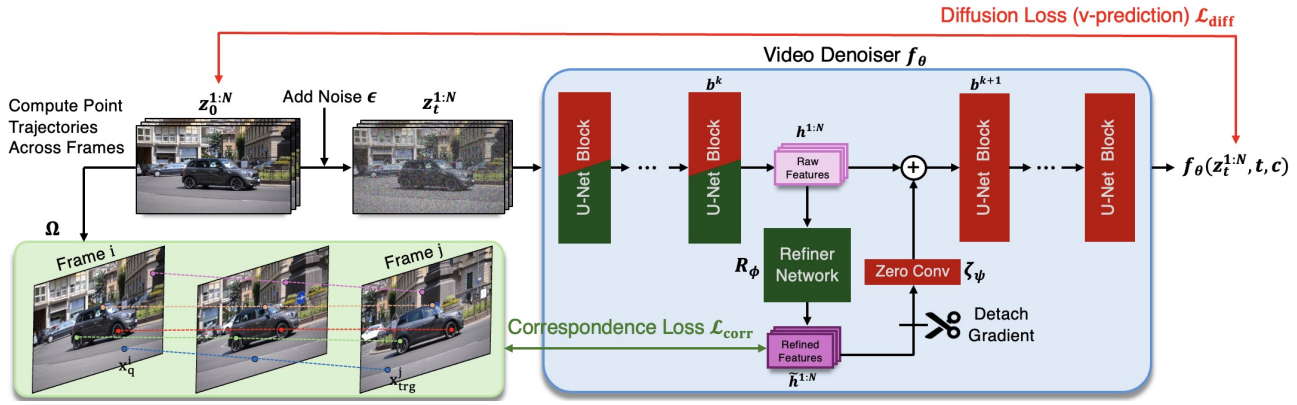


Figure 2. **Track4Gen overview**. Red-colored blocks represent layers optimized by the diffusion loss $\mathcal{L}_{\text{diff}}$, while green blocks are optimized by the correspondence loss $\mathcal{L}_{\text{corr}}$. Blocks colored both red and green are influenced by the joint loss, $\mathcal{L}_{\text{diff}} + \lambda\mathcal{L}_{\text{corr}}$. See text for details.

3.1. Background: Stable Video Diffusion

Starting from random Gaussian noise, diffusion models aim to generate clean images or videos via an iterative denoising process [27, 55]. This process reverses a fixed, time-dependent diffusion forward process, which gradually corrupts the data by adding Gaussian noise. While our method is applicable to general video diffusion models, in this paper, we design our architecture based on Stable Video Diffusion (SVD), a latent video diffusion model which employs the EDM-framework [35]. The diffusion process operates in the lower-dimensional latent space of a pre-trained VAE [37], consisting of an encoder $\mathcal{E}(\cdot)$ and a decoder $\mathcal{D}(\cdot)$.

Given a clean sample $\boldsymbol{x}_0^{1:N} \sim p_{\text{data}}(\boldsymbol{x})$ of an N-frame video sequence, the frames are first encoded into the latent space as $\boldsymbol{z}_0^{1:N} = \mathcal{E}(\boldsymbol{x}_0^{1:N})$. Gaussian noise $\boldsymbol{\epsilon} \sim \mathcal{N}(0, I)$ is then added to the latents to produce the intermediate noisy latents via the forward process $\boldsymbol{z}_t^{1:N} = \alpha_t \boldsymbol{z}_0^{1:N} + \sigma_t \boldsymbol{\epsilon}$, where t represents the diffusion timestep, and α_t, σ_t are the discretized noise scheduler parameters. The diffusion denoiser $\boldsymbol{f}_\theta$ is trained by minimizing the v-prediction loss:

$$\min_\theta \mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{N}(0,I), t\sim U[1,T]} \left[\left\| \boldsymbol{f}_\theta(\boldsymbol{z}_t^{1:N}, t, c) - \boldsymbol{y} \right\|_2^2 \right], \quad (1)$$

where $\boldsymbol{y}$ is defined as $\boldsymbol{y} = \alpha_t \boldsymbol{\epsilon} - \sigma_t \boldsymbol{z}_0^{1:N}$. In the image-to-video variant of SVD, the condition c refers to the CLIP image embedding [49], replacing the typical text embeddings. For the remainder of this paper, we will refer to Eq. 1 as the *video diffusion loss* $\mathcal{L}_{\text{diff}}$.

Once trained, the diffusion model generates videos by iteratively denoising a noisy latent $\boldsymbol{z}_T^{1:N}$ sequence sampled from pure Gaussian distribution. At each diffusion step, the model predicts the noise in the input latent. Once the clean latent $\boldsymbol{z}_0^{1:N}$ is obtained, the decoder $\mathcal{D}$ maps it to the higher-dimensional pixel space $\boldsymbol{x}_0^{1:N} = \mathcal{D}(\boldsymbol{z}_0^{1:N})$. For further details, we refer to the Appendix D of [4].

3.2. Video Diffusion Features

Previous studies have demonstrated that image diffusion models learn discriminative features in their hidden states that are effective for various analysis tasks and propose methods for improving the representation power of such features [9, 69, 73, 74]. Similarly, we argue that while also being powerful, internal representations of pre-trained video diffusion models may not be fully temporally consistent, resulting in appearance drift in generated videos.

To better investigate this hypothesis, we first evaluate the long-term video tracking capabilities of U-Net-based video diffusion models [4, 56, 78]. Specifically, we evaluate the effectiveness of the features from each block of the U-Net for the task of point tracking. Given a real-world video, we add a small amount of noise and extract feature maps from each layer in each block. We perform a cosine-similarity-based nearest-neighbor search [44, 59] over these feature maps for a given set of fixed query points on the first frame (we use a similarity threshold of 0.6 [62] in our experiments). We also perform a similar analysis for generated videos where we extract the feature maps corresponding to diffusion steps with small amount of noise.

Based on this feature analysis, we make some important observations. Notably, regardless of the model (we analyze both Zeroscope T2V [56] and SVD I2V [4]), we find out that output features from the *upsampler layer of the third decoder block* consistently yield stronger temporal correspondences, as shown in Fig. 3. Hence, we use this block when extracting features for the remainder of our experiments. Furthermore, when we analyze generated videos and point tracks estimated based on the feature maps (as shown in Fig. 4), we observe that there is a correlation between *tracking failures* that reveal feature-space inconsistencies and *appearance drifts* that reveal pixel-space inconsistencies. Hence, we hypothesize that enriching feature consistency can help mitigate such appearance drifts. Next, we

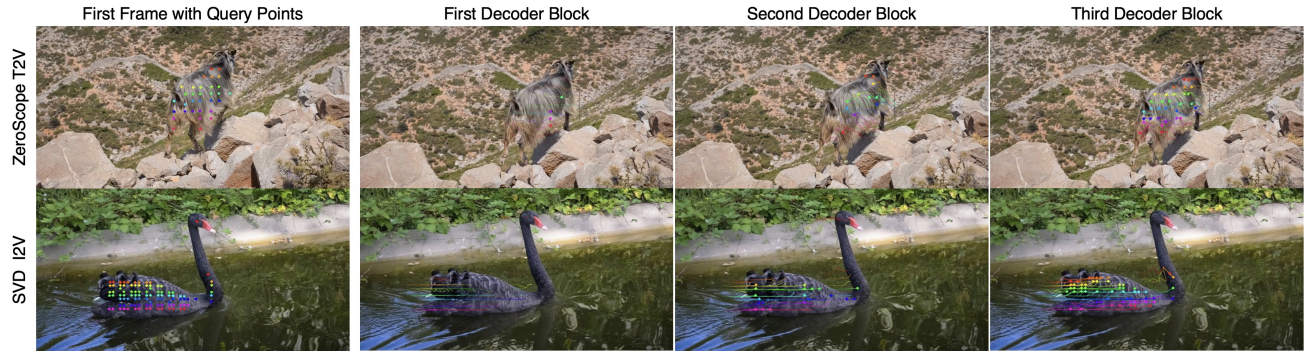


Figure 3. **Real-world video tracking using different video diffusion features**. Given color-coded query points on the first frame (*Leftmost column*), we display tracked points on target frames using features from different blocks (*right columns*). The 13th frame (*first row*) and 8th frame (*second row*) are shown as target frames. Full results are available in the supplementary and on our page.

introduce Track4Gen where we accomplish this goal by supervising video diffusion models with a joint tracking loss.

3.3. Track4Gen

Track4Gen aims to utilize point tracking as an additional supervision signal to enhance the spatial-awareness of video diffusion features. Given that we build on top of a pre-trained video generation model, to retain the prior knowledge and avoid tampering the original features directly, we propose a novel architecture change as shown in Fig. 2. Specifically, instead of directly using the raw diffusion features for correspondence estimation, we propose a trainable *refiner module* $\boldsymbol{R}_\phi$, which is designed to refine the raw features by projecting them into a correspondence-rich feature space. The refined features, which are spatially-aware, are then both used to estimate point tracks with an explicit supervision as well as feeding back to the generation backbone. We empirically find out that this design is more effective compared to fine-tuning the original model with no refinement module (see Sec. 4.2).

Given an N-frame video sequence $\boldsymbol{x}_0^{1:N}$, its corresponding latent $\boldsymbol{z}_0^{1:N}$, and a diffusion timestep t, in order to train Track4Gen we continue to utilize the standard diffusion training loss as defined in Eq. (1), where we adopt the velocity prediction objective [4, 35, 52] for $\mathcal{L}_{\text{diff}}$.

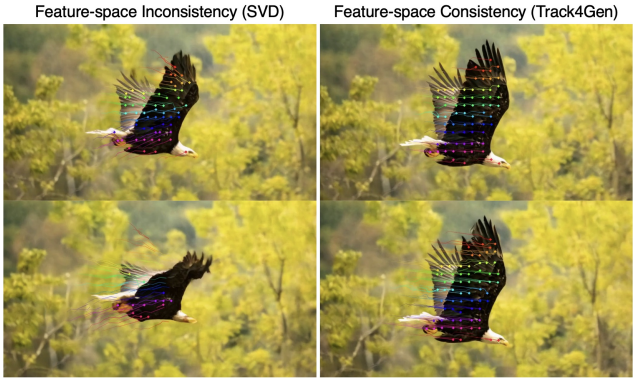


Figure 4. **Generated video tracking using video diffusion features.** Tracks based on diffusion features are annotated on the generated videos. Track4Gen generates more consistent results.

To enable tracking supervision, we assume access to a dense set of point trajectories $\Omega = \{(\mathrm{x}^i, \mathrm{x}^j)\}$ across frames where a point x^i in frame i corresponds to a matching point x^j in frame j and vice versa. Given the corresponding noisy video latent sequence $\boldsymbol{z}_t^{1:N}$, we first extract raw diffusion features as the hidden states $\boldsymbol{h}^{1:N} \in \mathbb{R}^{N\times H\times W\times C}$ from a specific block b^k within the U-Net, where b^k is set to the upsampler layer of the third decoder block (see Sec. 3.2). We then pass these features through the refiner module to obtain the refined feature map $\tilde{\boldsymbol{h}}^{1:N} = \boldsymbol{R}_\phi(\boldsymbol{h}^{1:N})$.

We sample a query point x_q^i along with its ground-truth target point $\mathrm{x}_\mathrm{trg}^j$ from the correspondence set Ω. Given the query point feature $\tilde{\boldsymbol{h}}^i(\mathrm{x}_\mathrm{q}) \in \mathbb{R}^{1\times 1\times C}$ and the target feature map $\tilde{\boldsymbol{h}}^j \in \mathbb{R}^{H\times W\times C}$, we calculate the cost volume $\boldsymbol{S} \in \mathbb{R}^{H\times W\times 1}$ as follows:

$$\boldsymbol{S}(\mathrm{p}) = \text{cos-sim}(\tilde{\boldsymbol{h}}^i(\mathrm{x}_\mathrm{q}), \tilde{\boldsymbol{h}}^j(\mathrm{p})), \tag{2}$$

where cos-sim denotes cosine similarity. The predicted target point $\hat{\mathrm{x}}_\mathrm{trg}$ is then determined using the differentiable soft-argmax operation:

$$\hat{\mathrm{x}}_\mathrm{trg} = \frac{\sum_{\mathrm{p}\in\Omega'} \boldsymbol{S}(\mathrm{p}) \cdot \mathrm{x}_\mathrm{p}}{\sum_{\mathrm{p}\in\Omega'} \boldsymbol{S}(\mathrm{p})}, \tag{3}$$

where $\Omega' = \{p : \left\|\mathrm{x}_\mathrm{p} - \mathrm{x}_{\mathrm{p}_{\max}}\right\|_2 \leq R\}$[1]. Thus, the target point prediction can be expressed as $\hat{\mathrm{x}}_\mathrm{trg} = \xi(\mathrm{x}_\mathrm{q}^i, j, \tilde{\boldsymbol{h}}^{1:N})$, and the predicted tracklet for x_q^i is given by $\mathcal{T}_{\mathrm{x}_\mathrm{q}^i} = \{\hat{\mathrm{x}}_n : \hat{\mathrm{x}}_n = \xi(\mathrm{x}_\mathrm{q}^i, n, \tilde{\boldsymbol{h}}^{1:N}), n = 1, ..., N\}$. Finally, the correspon-

[1]The feature maps have a resolution of 44×81 for an input video resolution of 320×576, and we set $R = 35$.

Figure 5. **Image-to-video generation results of the original SVD and Track4Gen.** Please visit our page for full video view.

dence loss $\mathcal{L}_{\text{corr}}$ is computed using the Huber loss L_H [31]:

$$\mathcal{L}_{\text{corr}}(\tilde{\boldsymbol{h}}^{1:N}, \Omega) = \sum_{(\text{x}_\text{q}^i, \text{x}_\text{trg}^j) \in \Omega} L_H(\xi(\text{x}_\text{q}^i, j, \tilde{\boldsymbol{h}}^{1:N}), \text{x}_\text{trg}^j) \quad (4)$$

When training Track4Gen, we initialize the refiner module as an *identity mapping* to fully leverage the prior of the base model at the start of finetuning. To re-route the refined features to the backbone generator, we introduce a trainable *zero convolution* layer [77], denoted as $\boldsymbol{\zeta}_\psi$. While the diffusion loss $\mathcal{L}_{\text{diff}}$ back-propagates to all the blocks of the video diffusion model, we detach the gradients of $\tilde{\boldsymbol{h}}^{1:N}$ before passing into $\boldsymbol{\zeta}_\psi$ such that refiner module can solely focus on acquiring the correspondence prior. Hence, given that the output of block b^k is $\boldsymbol{h}^{1:N}$, the input to the subsequent block b^{k+1} is computed as $\boldsymbol{h}^{1:N} + \boldsymbol{\zeta}_\psi(\text{stop-gradient}(\boldsymbol{R}_\phi(\boldsymbol{h}^{1:N})))$. Fig. 2 visualizes this architecture design, with red and green colors indicating the objective that optimizes each module.

4. Experiments

4.1. Implementation Details

To train Track4Gen, we construct a training dataset consisting of 567 video-trajectory pairs, with each video having a resolution of 320×576 and a duration of 24 frames. Since no real-world video with (dense) ground-truth trajectory annotations exist at the time of this work, we utilize optical flow to generate trajectory annotations. A key challenge is the need for accurate video segmentation maps to ensure a balanced distribution of trajectory points between foreground objects and the background [12]. To address this, we utilize public video datasets paired with ground-truth segmentation maps [7, 17, 39, 46, 48], where we split longer videos into 24-frame segments.

We use Stable Video Diffusion (SVD) image-to-video pretrained checkpoints[2] as the base video generator. Our proposed refiner module consists of eight stacked 2D convolution layers and is attached to the third decoder block of the SVD UNet. The refiner module preserves the shape of the hidden states throughout and is initialized as the identity mapping. Further details are provided in the supplementary. We finetune this enhanced video generator architecture for 20K steps with our joint loss $\mathcal{L}_{\text{diff}} + \lambda\mathcal{L}_{\text{corr}}$, where λ is set to 8. Rather than finetuning the entire model, we finetune only the temporal transformer blocks, the refiner module $\boldsymbol{R}_\phi$, and the zero convolution $\boldsymbol{\zeta}_\psi$. In each iteration, we sample 512 correspondence pairs from the precomputed trajectories. We use the AdamW optimizer [43] with a learning rate of $1e{-}5$, β_1=0.9, β_2=0.999, and a weight decay of $1e{-}2$. We train the model on $4\times$ H100 GPUs with a total batch size of 4. For sampling new videos, we apply the

[2] https://huggingface.co/stabilityai/stable-video-diffusion-img2vid-xt

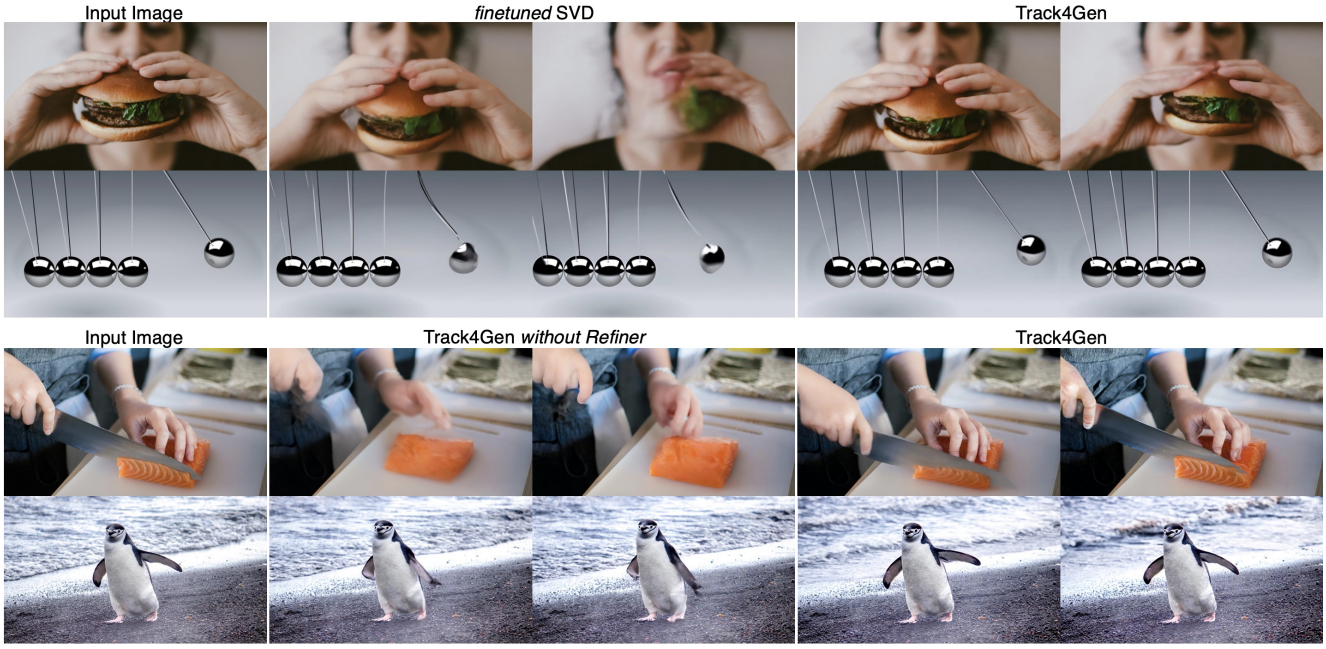


Figure 6. **Qualitative ablation on video generation.** Track4Gen is compared with *finetuned* SVD (SVD finetuned on the same training videos without any correspondence supervision) and Track4Gen trained without the Refiner module.

default settings using 30 steps with the EDM sampler [35], *motion bucket id* =127, and *fps*=7.

Table 1. **Quantitative comparison on video generation performance**. We compare Track4Gen to the pre-trained SVD * as well as a finetuned SVD on the same dataset (*finetuned* SVD). We also train a variant of Track4Gen without the refiner module. All videos are generated at 320x576 resolution, except SVD* (576p) which operates at 576x1024 resolution.

	Subject Consistency	Temporal Flickering	Motion Smoothness	Imaging Quality	Video-Image Alignment	FID	FVD
SVD*	0.9535	0.9464	0.9774	0.6648	0.9539	29.0	776
finetuned SVD	0.9665	0.9800	0.9909	0.6766	0.9771	27.0	735
Track4Gen *w/o refiner*	0.9506	0.9725	0.9791	0.6653	0.9614	27.1	**718**
Track4Gen	**0.9746**	**0.9806**	**0.9921**	**0.6835**	**0.9814**	**26.6**	724
SVD* (576p)	0.9576	0.9478	0.9795	0.6812	0.9582	-	-

4.2. Track4Gen for Video Generation

We evaluate Track4Gen for the image-to-video generation task via a series of experiments using multiple datasets, automated metrics, and human evaluations.

Evaluation Setup. We compare Track4Gen against the original SVD (SVD*) [4], as well as a version of SVD that is finetuned on the same videos as Track4Gen (*finetuned* SVD). Furthermore, we train a variant of Track4Gen without the refiner module. For VBench metrics [30], evaluations are conducted on the VBench-I2V dataset, containing 355 diverse images. FID and FVD are measured using the DAVIS [48] dataset as reference. We generate 24-frame videos conditioned on each input image.

Automatic metrics. We first report five key metrics from VBench [30]: (1) *Subject Consistency*—assesses subject appearance consistency of the video by computing the similarity of DINO [45] features. (2) *Temporal Flickering*—detects temporal consistency by taking static frames and calculating the mean absolute difference across frames. (3) *Motion Smoothness*—measures smoothness of motion, and how well it adheres to real-world physics, using video frame interpolation priors [42]. (4) *Image Quality*—evaluates distortions (e.g., noise, blur) using a pre-trained, multi-scale image quality predictor [36]. (5) *Video-Image Alignment*—measures alignment between the subject in the input image and in the generated video using DINO features. We additionally report FID [26] and FVD [63].

Human evaluation. We further evaluate Track4Gen against baselines through a user study. We ask 64 participants to compare our results with a randomly selected baseline. We ask the users to evaluate how consistent main objects appear across the frames in a generated video as well as how natural the depicted motion is. We provide further details of the user study in the supplementary material.

Qualitative results. Qualitative comparisons with the base SVD are shown in Fig. 5. As illustrated, Track4Gen generates videos with strong appearance consistency, avoiding issues of appearance drift. In contrast, videos produced by the original SVD exhibit noticeable inconsistencies: the sheep's head (row 1) mutates, the plane's wing (row 2) shows unnatural transitions, and the cars (row 3) disappear. Further comparisons with *finetuned* SVD and Track4Gen without the refiner module are shown in Fig. 6 and highlight the superior visual coherence of the proposed Track4Gen.

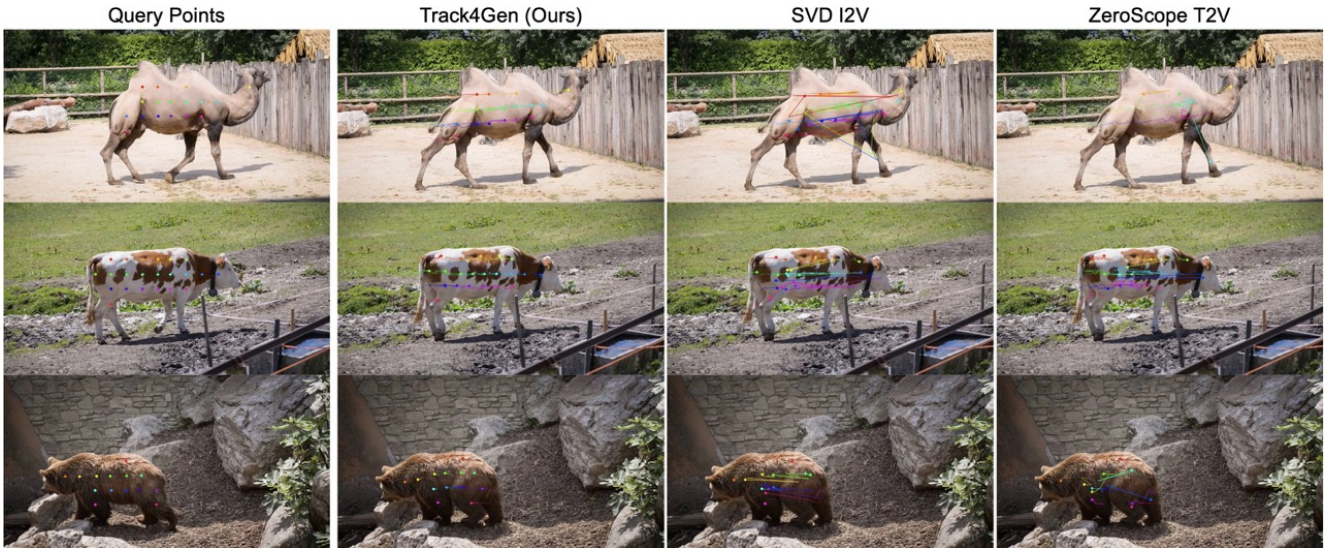
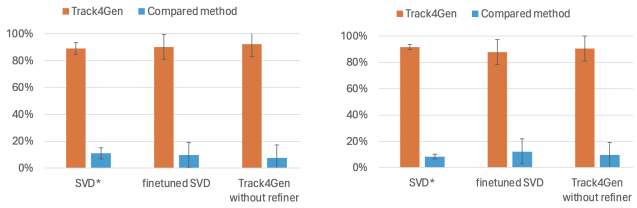


Figure 7. **Qualitative comparison of Track4Gen and baselines for real-world video tracking.** The leftmost column displays query points in the first frame, while the following three columns show tracking results using features from each model.



(a) Identity preservation (b) Motion naturalness

Figure 8. **User study results.** Our study shows that Track4Gen better preserves object identity and produces more natural motion.

Quantitative results. As shown in Tab. 1, our method achieves the highest scores across all 5 metrics from VBench, along with the lowest FID and second-lowest FVD values, outperforming the base SVD by substantial margins. Fig. 8 provides the user study results where the majority of the participants agreed that Track4Gen is superior both in terms of identity preservation and naturalness of motion.

4.3. Track4Gen for Video Tracking

We evaluate Track4Gen's capability to *track any point* in real videos by adding a small amount of noise to the input video [59] and passing it through the video denoiser $\boldsymbol{f}_\theta$ to extract feature maps. We first compare tracking results with such features against other raw features [4, 56, 60] in Sec. 4.3.1. In Sec. 4.3.2, we utilize Track4Gen's features in a test-time optimization method [62] and compare to both self-supervised and fully supervised video trackers.

4.3.1. Zero-shot Feature Comparison

We evaluate the precision of predicted tracks using the features from Track4Gen, the original SVD model (SVD*), and RAFT [60]. We also test another text-to-video model, ZeroScope T2V [56], to demonstrate how raw features from pre-trained video generators typically work out of the box.

Table 2. **Quantitative zero-shot feature comparison on video tracking benchmarks**. Track4Gen features are compared to the features of SVD* [4], ZeroScope [56], and RAFT [60]. For all the metrics, higher values indicate better performance.

	DAVIS-480p (24-frame)			BADJA (24-frame)		DAVIS-480p (whole duration)			BADJA (whole duration)	
Method	δ^x_{avg}	OA	AJ	δ^{seg}	δ^{3px}	δ^x_{avg}	OA	AJ	δ^{seg}	δ^{3px}
ZeroScope	46.2	67.0	39.4	27.5	2.8	37.2	59.5	27.8	19.9	2.0
SVD*	42.4	79.7	36.4	26.2	2.9	35.4	70.1	26.5	19.4	2.2
Track4Gen	69.7	**85.8**	**56.5**	52.3	7.7	58.9	**78.4**	**40.2**	40.4	5.0
RAFT	**73.3**	-	-	**54.8**	**8.7**	**66.7**	-	-	**45.0**	**5.8**

For RAFT, tracking is achieved by chaining optical flow displacements, while the others use nearest neighbor matching between its encoded features.

Datasets. We use TAP-Vid DAVIS [12] and BADJA [3] as benchmark datasets. Additionally, we include two shorter benchmarks, DAVIS (24-frame) and BADJA (24-frame), which focus on the first 24 frames with query and target points within this range. Details on encoding long videos with the video models are in the supplemental.

Metrics. For evaluating the TAP-Vid benchmarks, we use the following metrics: (i) Position Accuracy (δ^x_{avg}) evaluates the average accuracy of visible points, where each δ^x represents the fraction of predicted points that lie within x pixels of the ground-truth position, with $x \in \{1, 2, 4, 8, 16\}$. (ii) Occlusion Accuracy (OA) evaluates the correctness of occlusion predictions. (iii) Average Jaccard (AJ) jointly assesses both position and occlusion accuracy. For the BADJA dataset, we report δ^{seg}, which measures the accuracy of tracked keypoints within a distance of $0.2\sqrt{A}$ from the ground-truth annotation, where A is the area of the foreground object. We also report δ^{3px}, which assesses accuracy within a 3-pixel threshold. A cosine similarity threshold of 0.6 is used for occlusion prediction.

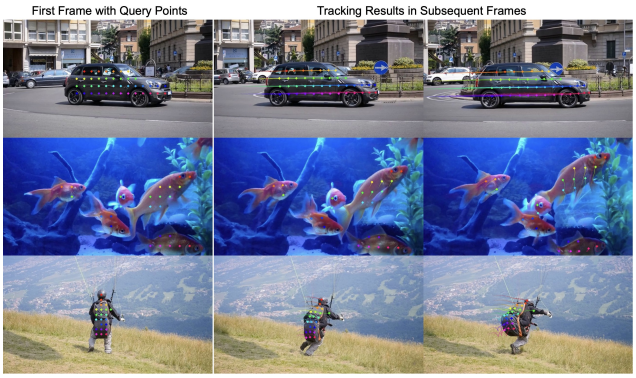


Figure 9. **Extending Track4Gen with test-time adaptation [62].**

Table 3. **Quantitative comparison with video trackers**. Although primarily designed for *video generation*, Track4Gen combined with a test-time optimization method [62] achieves performance comparable to dedicated *video tracking* frameworks, even when compared to supervised methods.

Method	DAVIS-480			BADJA	
	δ^x_{avg}	OA	AJ	δ^{seg}	δ^{3px}
TAP-Net⋆	66.4	79.0	46.0	45.4	9.6
PIPs++⋆	73.6	-	-	59.0	9.8
TAPIR⋆	77.3	**89.5**	**65.7**	68.7	10.5
Omnimotion†	74.1	84.5	58.4	45.2	6.9
DINO-Tracker†	**80.4**	88.1	64.6	**72.4**	**14.3**
DINO-Tracker w/ Track4Gen†	72.5	84.5	55.7	48.4	10.9

⋆ – supervised. † – test-time training.

Table 4. **Ablation on trainable modules and refiner**.

Trainable modules	Subject Consistency	Temporal Flickering	Motion Smoothness	Imaging Quality	Video-Image Alignment
spatial + temporal	0.9734	**0.9811**	0.9917	**0.6863**	0.9807
spatial	0.9726	0.9801	0.9919	0.6852	0.9811
temporal	**0.9746**	0.9806	**0.9921**	0.6835	**0.9814**
Refiner architecture					
2D convolutions	**0.9746**	**0.9806**	**0.9921**	**0.6835**	0.9814
3D convolutions	0.9687	0.9734	0.9904	0.6833	**0.9820**

Results. We present the qualitative results in Fig. 7 and the quantitative results in Tab. 2. Although primarily designed for video generation, Track4Gen boosts the poor performance of the pre-trained video models significantly, approaching the accuracy of RAFT optical flow chaining.

4.3.2. Extending Track4Gen with Test-time Adaptation

To further evaluate Track4Gen's long-term tracking capabilities, we integrate our features with test-time adaptation algorithm of DINO-Tracker [62], where a per-video optimization is performed using optical flow supervision. We replace the originally used DINOv2 [45] with the features from Track4Gen. We evaluate using the same datasets and metrics outlined in Sec. 4.3.1, against both fully-supervised trackers [12, 13, 79] and self-supervised methods [62, 65].

Tab. 3 shows that Track4Gen features optimized with [62] achieve performance comparable to dedicated trackers. Qualitative results are in Fig. 9 and in the supplemental.

Table 5. **Quantitative ablation on using annotated, but synthetic videos [20]**. *Left*: Video generation metrics. *Right*: Video tracking metrics.

Dataset composition	Subject Consistency	Motion Smoothness	Imaging Quality	BADJA δ^{seg}	BADJA δ^{3px}
real videos	**0.9747**	**0.9921**	**0.6833**	40.4	**5.0**
real + synthetic videos	0.9708	0.9892	0.6793	**42.1**	4.8

4.4. Ablation Studies

We present an ablation study in Tab. 4 where we train different set of modules. Each spatio-temporal block of SVD includes both spatial and temporal transformers. We compare training only spatial transformers, only temporal transformers, or both. We also ablate the architecture of the refiner module using either 2D or 3D convolution layers. Our analysis shows that while results are similar across settings, training only the temporal transformers in SVD with 2D convolutions as the refiner module yields optimal video generation quality. We further analyze our training dataset by additionally incorporating Kubric [20] simulated videos (1K video-track pairs from the Panning MOVi-E data [10, 13]) with automatically annotated trajectories into training. As shown in Tab. 5, optical flow-chained tracklets from real provides provide as effective correspondence guidance as tracklets from synthetic data, while synthetic videos negatively impact the video generation quality.

5. Conclusion and Future Work

We have presented the first unified framework that bridges two distinct tasks: video generation and dense point tracking. We demonstrated that this produces temporally consistent feature representations and appearance-consistent videos. As for limitations, videos generated by Track4Gen tend to exhibit less dynamic motion compared to those from other video generators. Additionally, failure cases are included in the supplementary material.

Future work. Recently, cutting-edge video trackers [10, 14, 34] have emerged, enabling dense, accurate, and long-term tracking, especially with better handling of occlusions. This opens up promising future directions for extending our work to utilize real-world videos, automatically annotated by these advanced trackers.

Acknowledgments. We thank Seokju Cho and Narek Tumanyan for their invaluable feedback on video point tracking. We also extend our gratitude to Mingi Kwon, Joon-Young Lee, and Gabriel Huang for their insightful discussions. Hyeonho Jeong and Jong Chul Ye are supported by the National Research Foundation of Korea (NRF) under Grants RS-2024-00336454 and RS-2023-00262527, and by the Institute for Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (RS-2019-II190075, Artificial Intelligence Graduate School Program, KAIST).

References

[1] Görkay Aydemir, Weidi Xie, and Fatma Güney. Can visual foundation models achieve long-term point tracking? *arXiv preprint arXiv:2408.13575*, 2024. 2

[2] Weikang Bian, Zhaoyang Huang, Xiaoyu Shi, Yitong Dong, Yijin Li, and Hongsheng Li. Context-tap: Tracking any point demands spatial context features. *arXiv preprint arXiv:2306.02000*, 3, 2023. 2

[3] Benjamin Biggs, Thomas Roddick, Andrew Fitzgibbon, and Roberto Cipolla. Creatures great and smal: Recovering the shape and motion of animals from video. In *Computer Vision–ACCV 2018: 14th Asian Conference on Computer Vision, Perth, Australia, December 2–6, 2018, Revised Selected Papers, Part V 14*, pages 3–19. Springer, 2019. 7

[4] Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023. 1, 2, 3, 4, 6, 7

[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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22563–22575, 2023. 2

[6] Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, Clarence Ng, Ricky Wang, and Aditya Ramesh. Video generation models as world simulators. 2024. 1, 2

[7] Sergi Caelles, Alberto Montes, Kevis-Kokitsi Maninis, Yuhua Chen, Luc Van Gool, Federico Perazzi, and Jordi Pont-Tuset. The 2018 davis challenge on video object segmentation. *arXiv preprint arXiv:1803.00557*, 2018. 5

[8] Haoxin Chen, Menghan Xia, Yingqing He, Yong Zhang, Xiaodong Cun, Shaoshu Yang, Jinbo Xing, Yaofang Liu, Qifeng Chen, Xintao Wang, et al. Videocrafter1: Open diffusion models for high-quality video generation. *arXiv preprint arXiv:2310.19512*, 2023. 2

[9] X Chen, Z Liu, S Xie, and K He. Deconstructing denoising diffusion models for self-supervised learning. arxiv 2024. *arXiv preprint arXiv:2401.14404*. 3

[10] Seokju Cho, Jiahui Huang, Jisu Nam, Honggyu An, Seungryong Kim, and Joon-Young Lee. Local all-pair correspondence for point tracking. *arXiv preprint arXiv:2407.15420*, 2024. 2, 8

[11] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021. 2

[12] Carl Doersch, Ankush Gupta, Larisa Markeeva, Adria Recasens, Lucas Smaira, Yusuf Aytar, Joao 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. 2, 5, 7, 8

[13] 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. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10061–10072, 2023. 2, 8

[14] Carl Doersch, Yi Yang, Dilara Gokay, Pauline Luc, Skanda Koppula, Ankush Gupta, Joseph Heyward, Ross Goroshin, João Carreira, and Andrew Zisserman. Bootstap: Bootstrapped training for tracking-any-point. *arXiv preprint arXiv:2402.00847*, 2024. 8

[15] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021. 2

[16] Niladri Shekhar Dutt, Sanjeev Muralikrishnan, and Niloy J. Mitra. Diffusion 3d features (diff3f): Decorating untextured shapes with distilled semantic features. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4494–4504, 2024. 2

[17] Qingnan Fan, Fan Zhong, Dani Lischinski, Daniel Cohen-Or, and Baoquan Chen. Jumpcut: non-successive mask transfer and interpolation for video cutout. *ACM Trans. Graph.*, 34 (6):195–1, 2015. 5

[18] Stephanie Fu, Mark Hamilton, Laura E. Brandt, Axel Feldmann, Zhoutong Zhang, and William T. Freeman. Featup: A model-agnostic framework for features at any resolution. In *ICLR*, 2024. 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

[20] Klaus Greff, Francois Belletti, Lucas Beyer, Carl Doersch, Yilun Du, Daniel Duckworth, David J Fleet, Dan Gnanapragasam, Florian Golemo, Charles Herrmann, et al. Kubric: A scalable dataset generator. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3749–3761, 2022. 2, 8

[21] Yuchao Gu, Yipin Zhou, Bichen Wu, Licheng Yu, Jia-Wei Liu, Rui Zhao, Jay Zhangjie Wu, David Junhao Zhang, Mike Zheng Shou, and Kevin Tang. Videoswap: Customized video subject swapping with interactive semantic point correspondence. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7621–7630, 2024. 2

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

[23] Adam W Harley, Zhaoyuan Fang, and Katerina Fragkiadaki. Particle video revisited: Tracking through occlusions using point trajectories. In *European Conference on Computer Vision*, pages 59–75. Springer, 2022. 2

[24] Yingqing He, Tianyu Yang, Yong Zhang, Ying Shan, and Qifeng Chen. Latent video diffusion models for high-fidelity long video generation. *arXiv preprint arXiv:2211.13221*, 2022. 2

[25] Eric Hedlin, Gopal Sharma, Shweta Mahajan, Hossam Isack, Abhishek Kar, Andrea Tagliasacchi, and Kwang Moo Yi. Unsupervised semantic correspondence using stable diffusion. In *NIPS*, 2023. 2

[26] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017. 2, 6

[27] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 3

[28] 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

[29] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *Advances in Neural Information Processing Systems*, 35:8633–8646, 2022. 2

[30] Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, et al. Vbench: Comprehensive benchmark suite for video generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21807–21818, 2024. 2, 6

[31] Peter J Huber. Robust estimation of a location parameter. In *Breakthroughs in statistics: Methodology and distribution*, pages 492–518. Springer, 1992. 5

[32] Allan Jabri, Andrew Owens, and Alexei Efros. Space-time correspondence as a contrastive random walk. *Advances in neural information processing systems*, 33:19545–19560, 2020. 2

[33] 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. 2

[34] Nikita Karaev, Iurii Makarov, Jianyuan Wang, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. Cotracker3: Simpler and better point tracking by pseudo-labelling real videos. *arXiv preprint arXiv:2410.11831*, 2024. 2, 8

[35] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *Advances in neural information processing systems*, 35:26565–26577, 2022. 3, 4, 6

[36] Junjie Ke, Qifei Wang, Yilin Wang, Peyman Milanfar, and Feng Yang. Musiq: Multi-scale image quality transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5148–5157, 2021. 6

[37] Diederik P Kingma. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 3

[38] Alexander C. Li, Mihir Prabhudesai, Shivam Duggal, Ellis Brown, and Deepak Pathak. Your diffusion model is secretly a zero-shot classifier. In *ICCV*, 2023. 2

[39] Fuxin Li, Taeyoung Kim, Ahmad Humayun, David Tsai, and James M Rehg. Video segmentation by tracking many figure-ground segments. In *Proceedings of the IEEE international conference on computer vision*, pages 2192–2199, 2013. 5

[40] Hongyang Li, Hao Zhang, Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, and Lei Zhang. Taptr: Tracking any point with transformers as detection. *arXiv preprint arXiv:2403.13042*, 2024. 2

[41] Xinghui Li, Jingyi Lu, Kai Han, and Victor Prisacariu. Sd4match: Learning to prompt stable diffusion model for semantic matching. In *CVPR*, 2023. 2

[42] Zhen Li, Zuo-Liang Zhu, Ling-Hao Han, Qibin Hou, Chun-Le Guo, and Ming-Ming Cheng. Amt: All-pairs multi-field transforms for efficient frame interpolation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9801–9810, 2023. 6

[43] Ilya Loshchilov, Frank Hutter, et al. Fixing weight decay regularization in adam. *arXiv preprint arXiv:1711.05101*, 5, 2017. 5

[44] Grace Luo, Lisa Dunlap, Dong Huk Park, Aleksander Holynski, and Trevor Darrell. Diffusion hyperfeatures: Searching through time and space for semantic correspondence. *Advances in Neural Information Processing Systems*, 36, 2024. 3

[45] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023. 2, 6, 8

[46] Federico Perazzi, Jordi Pont-Tuset, Brian McWilliams, Luc Van Gool, Markus Gross, and Alexander Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 724–732, 2016. 5

[47] Adam Polyak, Amit Zohar, Andrew Brown, Andros Tjandra, Animesh Sinha, Ann Lee, Apoorv Vyas, Bowen Shi, Chih-Yao Ma, Ching-Yao Chuang, et al. Movie gen: A cast of media foundation models. *arXiv preprint arXiv:2410.13720*, 2024. 1, 2

[48] 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. 5, 6

[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*, pages 8748–8763. PMLR, 2021. 2, 3

[50] Robin Rombach, A. Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2021. 2

[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. 2

[52] Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. *arXiv preprint arXiv:2202.00512*, 2022. 4

[53] Peter Sand and Seth Teller. Particle video: Long-range motion estimation using point trajectories. *International journal of computer vision*, 80:72–91, 2008. 2

[54] 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

[55] 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

[56] Spencer Sterling. Zeroscope, 2023. https://huggingface.co/cerspense/zeroscope_v2_576w. 3, 7

[57] Shobhita Sundaram, Stephanie Fu, Lukas Muttenthaler, Netanel Y. Tamir, Lucy Chai, Simon Kornblith, Trevor Darrell, and Phillip Isola. When does perceptual alignment benefit vision representations? In *NIPS*, 2024. 2

[58] Saksham Suri, Matthew Walmer, Kamal Gupta, and Abhinav Shrivastava. Lift: A surprisingly simple lightweight feature transform for dense vit descriptors. In *ECCV*, pages 110–128. Springer, 2024. 2

[59] Luming Tang, Menglin Jia, Qianqian Wang, Cheng Perng Phoo, and Bharath Hariharan. Emergent correspondence from image diffusion. *Advances in Neural Information Processing Systems*, 36:1363–1389, 2023. 3, 7

[60] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*, pages 402–419. Springer, 2020. 7

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

[62] Narek Tumanyan, Assaf Singer, Shai Bagon, and Tali Dekel. Dino-tracker: Taming dino for self-supervised point tracking in a single video. *arXiv preprint arXiv:2403.14548*, 2024. 2, 3, 7, 8

[63] Thomas Unterthiner, Sjoerd Van Steenkiste, Karol Kurach, Raphael Marinier, Marcin Michalski, and Sylvain Gelly. Towards accurate generative models of video: A new metric & challenges. *arXiv preprint arXiv:1812.01717*, 2018. 2, 6

[64] Pascal Vincent. A connection between score matching and denoising autoencoders. *Neural computation*, 23(7):1661–1674, 2011. 1

[65] Qianqian Wang, Yen-Yu Chang, Ruojin Cai, Zhengqi Li, Bharath Hariharan, Aleksander Holynski, and Noah Snavely. Tracking everything everywhere all at once. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19795–19806, 2023. 2, 8

[66] Qian Wang, Abdelrahman Eldesokey, Mohit Mendiratta, Fangneng Zhan, Adam Kortylewski, Christian Theobalt, and Peter Wonka. Zero-shot video semantic segmentation based on pre-trained diffusion models, 2024. 2

[67] Yaohui Wang, Xinyuan Chen, Xin Ma, Shangchen Zhou, Ziqi Huang, Yi Wang, Ceyuan Yang, Yinan He, Jiashuo Yu, Peiqing Yang, et al. Lavie: High-quality video generation with cascaded latent diffusion models. *arXiv preprint arXiv:2309.15103*, 2023. 2

[68] Jay Zhangjie Wu, Guian Fang, Haoning Wu, Xintao Wang, Yixiao Ge, Xiaodong Cun, David Junhao Zhang, Jia-Wei Liu, Yuchao Gu, Rui Zhao, et al. Towards a better metric for text-to-video generation. *arXiv preprint arXiv:2401.07781*, 2024. 2

[69] Weilai Xiang, Hongyu Yang, Di Huang, and Yunhong Wang. Denoising diffusion autoencoders are unified self-supervised learners. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15802–15812, 2023. 3

[70] Yuxi Xiao, Qianqian Wang, Shangzhan Zhang, Nan Xue, Sida Peng, Yujun Shen, and Xiaowei Zhou. Spatialtracker: Tracking any 2d pixels in 3d space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20406–20417, 2024. 2

[71] Jiarui Xu, Sifei Liu, Arash Vahdat, Wonmin Byeon, Xiaolong Wang, and Shalini De Mello. ODISE: Open-Vocabulary Panoptic Segmentation with Text-to-Image Diffusion Models. In *CVPR*, 2023. 2

[72] Jiale Yang, So Kanazawa, Masami K Yamaguchi, and Isamu Motoyoshi. Pre-constancy vision in infants. *Current Biology*, 25(24):3209–3212, 2015. 1

[73] Xingyi Yang and Xinchao Wang. Diffusion model as representation learner. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 18938–18949, 2023. 3

[74] Sihyun Yu, Sangkyung Kwak, Huiwon Jang, Jongheon Jeong, Jonathan Huang, Jinwoo Shin, and Saining Xie. Representation alignment for generation: Training diffusion transformers is easier than you think. *arXiv preprint arXiv:2410.06940*, 2024. 2, 3

[75] Yuanwen Yue, Anurag Das, Francis Engelmann, Siyu Tang, and Jan Eric Lenssen. Improving 2D Feature Representations by 3D-Aware Fine-Tuning. In *ECCV*, 2024. 2

[76] 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. 2

[77] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3836–3847, 2023. 5

[78] Shiwei Zhang, Jiayu Wang, Yingya Zhang, Kang Zhao, Hangjie Yuan, Zhiwu Qin, Xiang Wang, Deli Zhao, and Jingren Zhou. I2vgen-xl: High-quality image-to-video synthesis via cascaded diffusion models. *arXiv preprint arXiv:2311.04145*, 2023. 2, 3

[79] Yang Zheng, Adam W Harley, Bokui Shen, Gordon Wetzstein, and Leonidas J Guibas. Pointodyssey: A large-scale synthetic dataset for long-term point tracking. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19855–19865, 2023. 8

[80] Daquan Zhou, Weimin Wang, Hanshu Yan, Weiwei Lv, Yizhe Zhu, and Jiashi Feng. Magicvideo: Efficient video generation with latent diffusion models. *arXiv preprint arXiv:2211.11018*, 2022. 2