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

Supplementary Material

This supplementary material is structured as follows: Sec. 1 provides additional implementation details for the experiments. In Sec. 2, we report supplementary quantitative metrics for video generation assessment. Sec. 3 presents additional qualitative results for image-to-video generation, while Sec. 4 focuses on qualitative video tracking results. Following this, we discuss the potential limitations and failure cases of Track4Gen in Sec. 5.

A comprehensive view of results in the form of videos is available on our project page. Furthermore, an extensive video generation comparison against all baselines can be found on this page.

1. Experimental Details

1.1. Preprocessing Video Correspondence

We utilize RAFT optical flow [10] to compute dense point trajectories across video frames. RAFT has demonstrated robust point tracking performance across various input types [12], even compared to supervised trackers like TAP-Net [5]. Following previous tracking literature [11, 12], we first compute pairwise correspondences between all consecutive frames. Tracks are then formed by chaining the estimated flow fields and filtered using a cycle consistency constraint. Specifically, given a point x^i in frame i and optical flow between frames i and $i+1$ denoted as $\boldsymbol{f}_{i\rightarrow i+1}$, the corresponding point in frame $i+1$ is estimated as $\mathrm{x}^{i+1} = \mathrm{x}^i + \boldsymbol{f}_{i\rightarrow i+1}(\mathrm{x}^i)$. We retain the pair $(\mathrm{x}^i, \mathrm{x}^{i+1})$ only if it satisfies $\|\mathrm{x}^i - (\mathrm{x}^{i+1} + \boldsymbol{f}_{i+1\rightarrow i}(\mathrm{x}^{i+1}))\|_2 \leq 1.5$, where $h\times w$ is set as 320×576. Also, a pair $(\mathrm{x}^i, \mathrm{x}^j)$ is filtered out if $\|\mathrm{x}^j - \mathrm{x}^{i\rightarrow j}\|_2 \geq 2$ and $\|\mathrm{x}^i - (\mathrm{x}^{i\rightarrow j} + \boldsymbol{f}_{j\rightarrow i}(\mathrm{x}^{i\rightarrow j}))\|_2 \leq 1.5$.

1.2. Refiner Network

When training Track4Gen, we design a convolutional neural network for the refiner module $\boldsymbol{R}_\phi$. The network comprises 8 layers, each with a fixed channel dimension of 640, a kernel size of 3, stride of 1, and padding of 1. The first 7 layers follow the structure `Conv2d` → `BatchNorm2d` → `ReLU`, except for the last layer which consists of `Conv2d` → `ReLU`.

To better demonstrate the architecture of the baseline *Track4Gen without Refiner*, we provide a visualization in Fig. 1. The figure compares the training schemes of this baseline with Track4Gen. In this variant, the correspondence loss $\mathcal{L}_{\text{corr}}$ is computed directly from the raw video diffusion features $\boldsymbol{h}^{1:N}$.

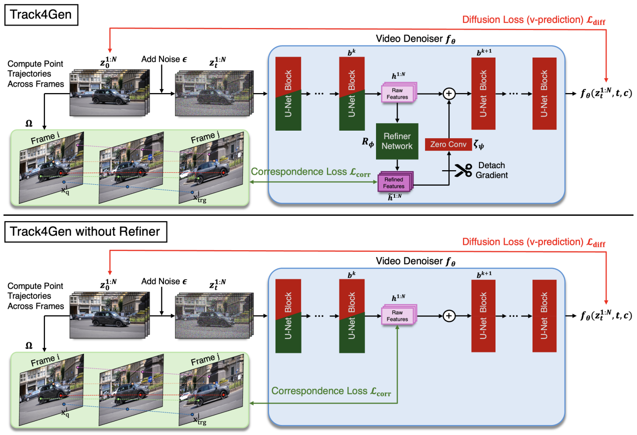


Figure 1. **Comparison of Track4Gen with and without Refiner.** *Top:* Correspondence loss $\mathcal{L}_{\text{corr}}$ is computed using the refined features $\tilde{\boldsymbol{h}}^{1:N}$. *Bottom:* Correspondence loss $\mathcal{L}_{\text{corr}}$ is computed using the raw diffusion features $\boldsymbol{h}^{1:N}$.

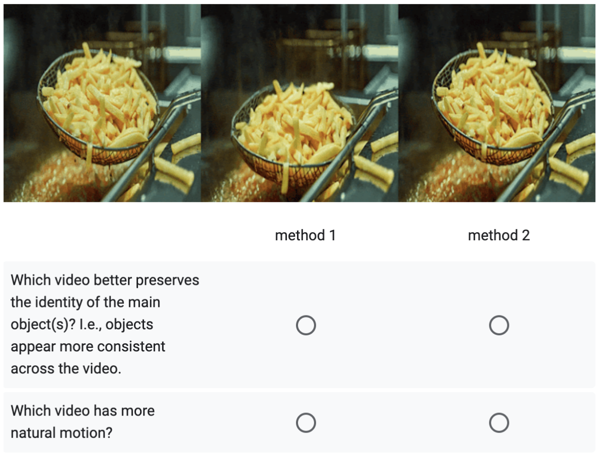
1.3. User Study Details

Fig. 2 shows an example of our user evaluation page. The input image is displayed on the left, while the middle and right columns show two generated videos for comparison. One result is from Track4Gen, and the other is randomly selected from four baselines: pretrained Stable Video Diffusion [3], finetuned Stable Video Diffusion without correspondence supervision, and Track4Gen trained without the refiner module. Note that the order of Track4Gen and the baseline is randomly shuffled (i.e., Track4Gen may appear first or the baseline may appear first). Participants are asked to answer two questions: (i) *Identity preservation*: Which video better preserves the identity of the main object(s)? (ii) *Motion naturalness*: Which video has more natural motion?

1.4. Encoding Long Videos with Video Diffusion Models

Majority of video diffusion models struggle with flexibility in temporal resolution. Specifically, if a model is trained on a fixed temporal resolution of N frames (e.g., $N = 24$), the quality of generated videos significantly degrades when attempting to generate videos with a much larger number of frames. Similarly, when these models are used as video feature extractors, the extracted features are invalid if the input video contains significantly more frames than the model was trained to handle.

This limitation poses a challenge, as most videos in



	method 1	method 2
Which video better preserves the identity of the main object(s)? I.e., objects appear more consistent across the video.	○	○
Which video has more natural motion?	○	○

Figure 2. **Example user evaluation page.** The order of Track4Gen and the baseline is randomly shuffled to ensure a fair comparison.

Table 1. **CLIP similarity and LPIPS comparison for assessing temporal consistency**. We compare Track4Gen to the pre-trained SVD as well as a finetuned SVD on the same dataset (*finetuned* SVD), and a variant of Track4Gen without the refiner module.

	CLIPSIM ↑	LPIPS ↓
Pretrained SVD	0.9839	0.1373
finetuned SVD	0.9869	0.0913
Track4Gen *without refiner*	0.9923	0.0547
Track4Gen	**0.9924**	**0.0533**

video tracking benchmarks contain more frames than the training resolution of video diffusion models. To address this, for a benchmark video with temporal resolution M, where $M \gg N$, we split the M-frame video into N-frame segments and encode each segment independently. For the final segment, which may contain fewer than N frames, we extend it by borrowing frames from the previous segment. For instance, if the last segment is 14 frames long and $N = 24$, we append the last 10 frames from the previous segment to complete the sequence. This extended segment is then passed through the video diffusion model to extract features. After encoding, we discard the features of the the borrowed frames, retaining only the features for the original frames in the segment.

2. Additional Metrics

To further evaluate the temporal consistency of generated videos, we report CLIPSIM [8] and LPIPS [14] metrics. For CLIPSIM, we compute the average CLIP similarity between all neighboring frame pairs using the CLIP Image Encoder. Similarly, we calculate the average LPIPS distance between neighboring frame pairs to assess perceptual differences. As shown in Tab. 1, Track4Gen achieves the highest CLIP similarity and lowest LPIPS distance, demonstrating its superior temporal consistency in the videos it generates.

3. Additional Video Generation Results

3.1. Comparisons

In Fig. 4 and 5, we present a comparison of Track4Gen against all three baselines: (1) the pretrained Stable Video Diffusion, (2) Stable Video Diffusion finetuned without the tracking loss, and (3) Track4Gen trained without the Refiner module. For a better view, please visit *page 2 (top)* of our project page.

3.2. Video Generation with Embedded Tracks

To demonstrate that Track4Gen generates videos with temporally consistent feature representations, we visualize the predicted point tracks annotated on the generated videos in Fig. 3. These tracks are computed in a zero-shot setting, using the intermediate features extracted from the final denoising step.

4. Additional Video Tracking Results

4.1. Feature Comparisons

DINO features [4, 7] are widely recognized for their accuracy in image correspondence tasks [1, 7, 13] and have also been shown to excel in temporal correspondence matching across videos [2, 11]. Thus, in Fig. 6, we present additional comparisons of video tracking using the intermediate features of pretrained models, including Track4Gen, DINOv2 [7], Stable Video Diffusion [3], and Zeroscope [9]. Furthermore, Fig. 7 offers a direct comparison between Track4Gen and DINOv2 features. While Track4Gen features demonstrate robustness, they are less effective in videos with occlusions.

4.2. Track4Gen with DINO-Tracker

We present additional results of adapting Track4Gen features with DINO-Tracker [11] in Fig. 8. Moreover, the optimization progress is visualized in Fig. 9, showing how the optical flow-guided test-time adaptation enhances the incomplete raw Track4Gen features.

5. Discussion on Limitation and Failure

For video results related to this section, please refer to *page 4* of our project page. While Track4Gen significantly enhances appearance constancy in generated videos, it tends to result in reduced camera motion compared to the original Stable Video Diffusion prior, a behavior also observed in the finetuned Stable Video Diffusion baseline. (see Fig. 11). We attribute this to the training dataset used for finetuning. In addition, in some cases Track4Gen produces unrealistic motion and exhibit artifacts on human faces and hands, particularly when the resolution or size of the human subject in the video is small — a common limitation shared by video diffusion models [6], including the baselines. Typical failure cases of video generation are illustrated in Fig. 12.

We also present failure cases of real-world video tracking in Fig. 10. Track4Gen features often struggle to capture accurate correspondences in videos with fast-moving objects and blurred frames. Additionally, Track4Gen lacks robustness in challenging videos with multiple semantically similar objects, where trajectories can shift from one object to another. An interesting direction for future work is augmenting the proposed correspondence loss with additional terms that account for occlusion predictions, which could further improve video generation performance.

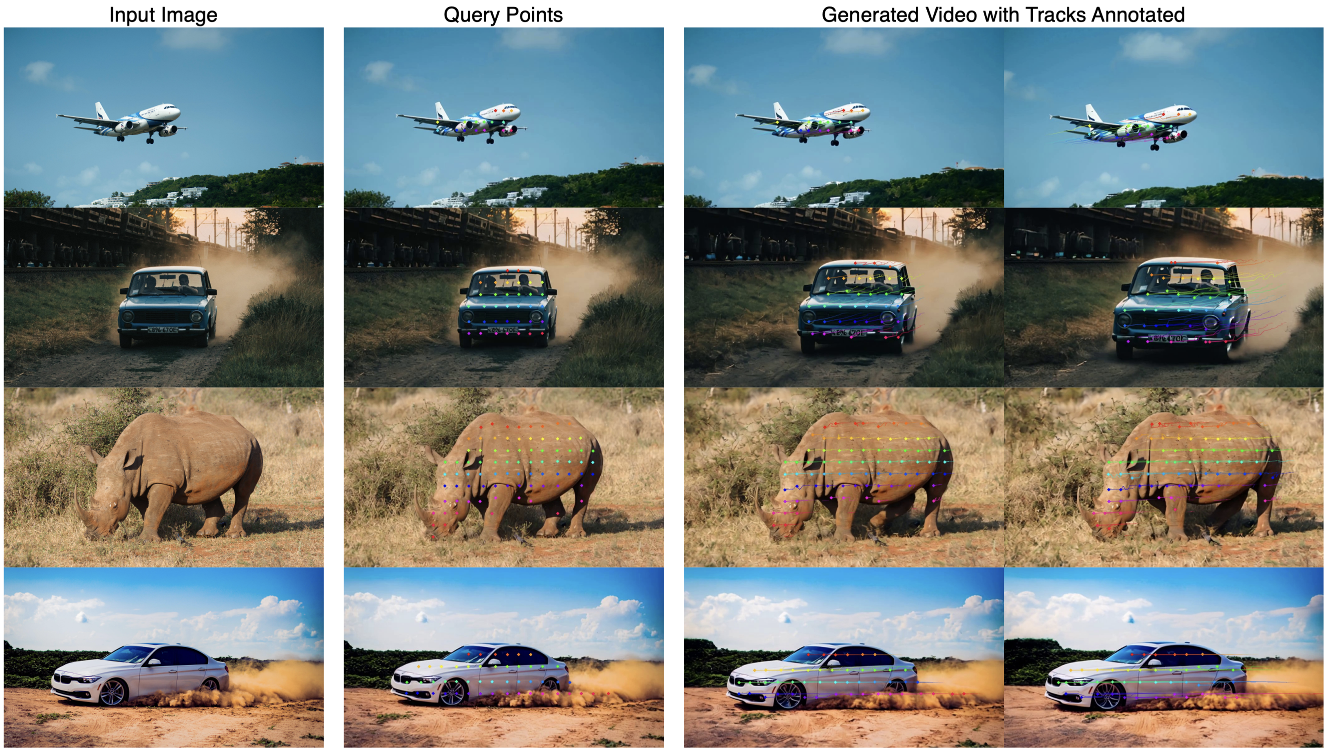


Figure 3. **Generated Videos with Embedded Tracks.** Predicted point tracks are annotated on the videos generated by Track4Gen.

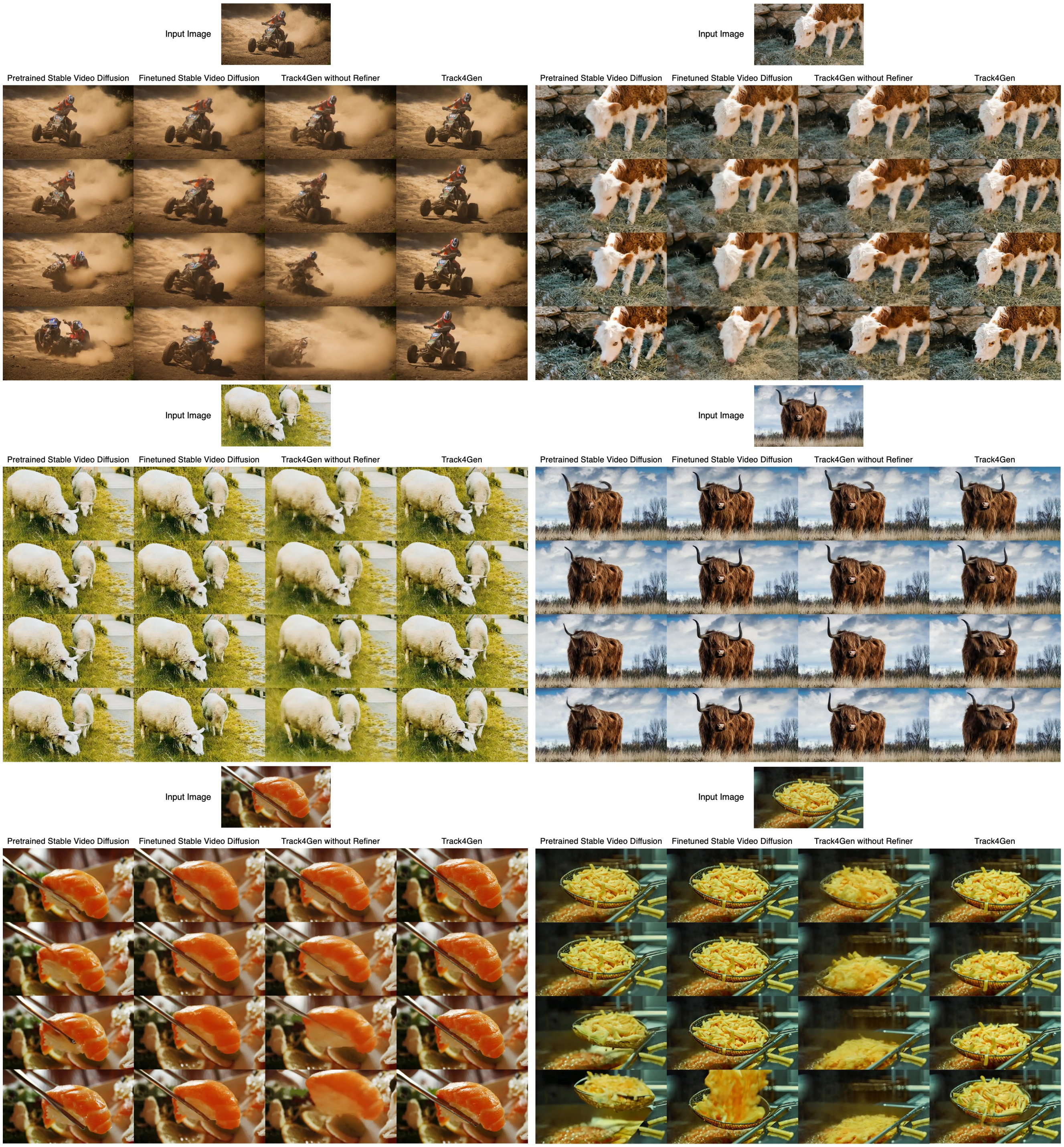

Input Image
Input Image
Pretrained Stable Video Diffusion
Finetuned Stable Video Diffusion
Track4Gen without Refiner
Track4Gen
Pretrained Stable Video Diffusion
Finetuned Stable Video Diffusion
Track4Gen without Refiner
Track4Gen
Input Image
Input Image
Pretrained Stable Video Diffusion
Finetuned Stable Video Diffusion
Track4Gen without Refiner
Track4Gen
Pretrained Stable Video Diffusion
Finetuned Stable Video Diffusion
Track4Gen without Refiner
Track4Gen
Input Image
Input Image
Pretrained Stable Video Diffusion
Finetuned Stable Video Diffusion
Track4Gen without Refiner
Track4Gen
Pretrained Stable Video Diffusion
Finetuned Stable Video Diffusion
Track4Gen without Refiner
Track4Gen

Figure 4. **Qualitative video generation results:** Track4Gen compared against all three baselines.

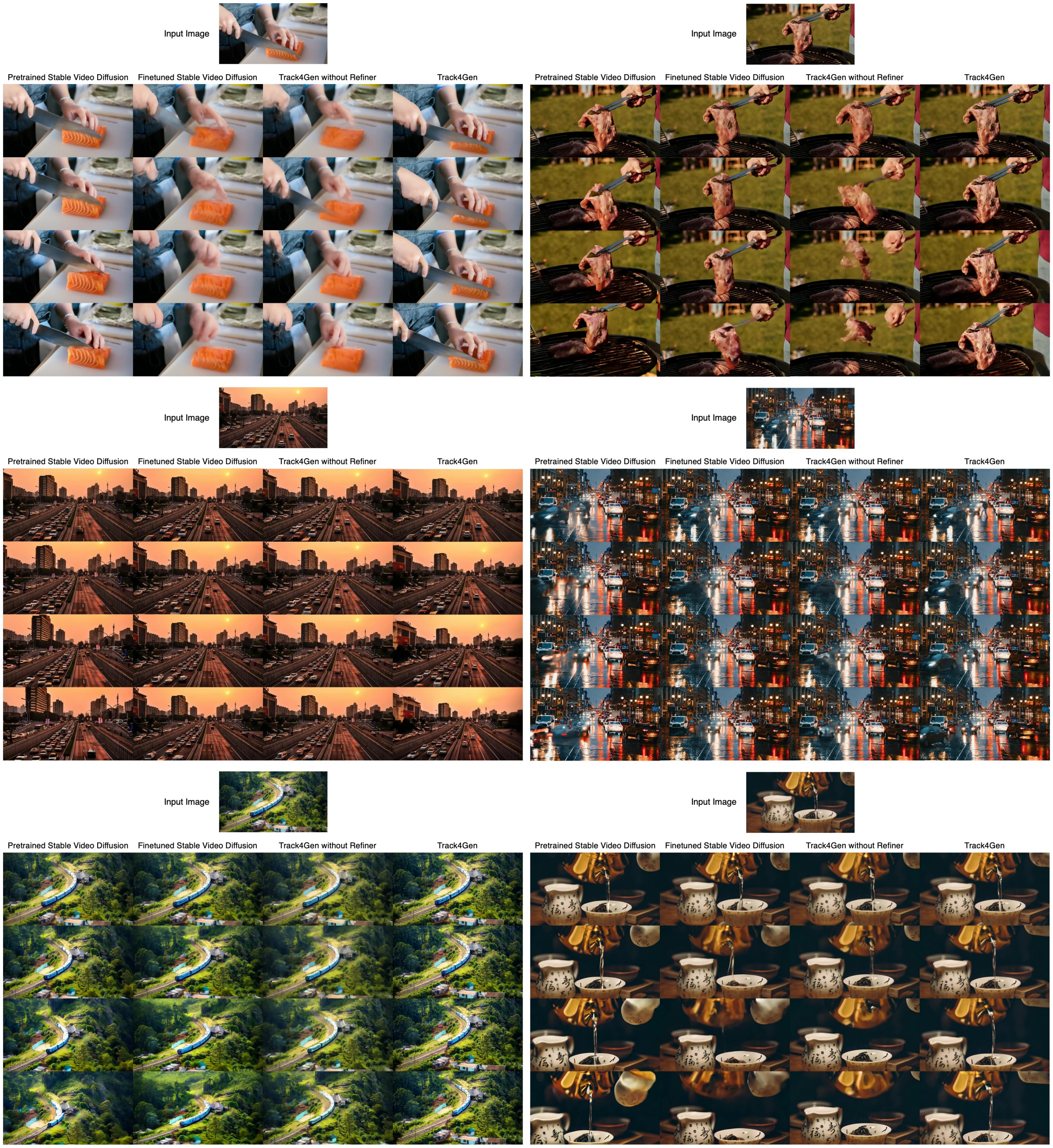

Input Image
Pretrained Stable Video Diffusion
Finetuned Stable Video Diffusion
Track4Gen without Refiner
Track4Gen
Input Image
Pretrained Stable Video Diffusion
Finetuned Stable Video Diffusion
Track4Gen without Refiner
Track4Gen
Input Image
Pretrained Stable Video Diffusion
Finetuned Stable Video Diffusion
Track4Gen without Refiner
Track4Gen
Input Image
Pretrained Stable Video Diffusion
Finetuned Stable Video Diffusion
Track4Gen without Refiner
Track4Gen
Input Image
Pretrained Stable Video Diffusion
Finetuned Stable Video Diffusion
Track4Gen without Refiner
Track4Gen
Input Image
Pretrained Stable Video Diffusion
Finetuned Stable Video Diffusion
Track4Gen without Refiner
Track4Gen

Figure 5. **Qualitative video generation results:** Track4Gen compared against all three baselines.

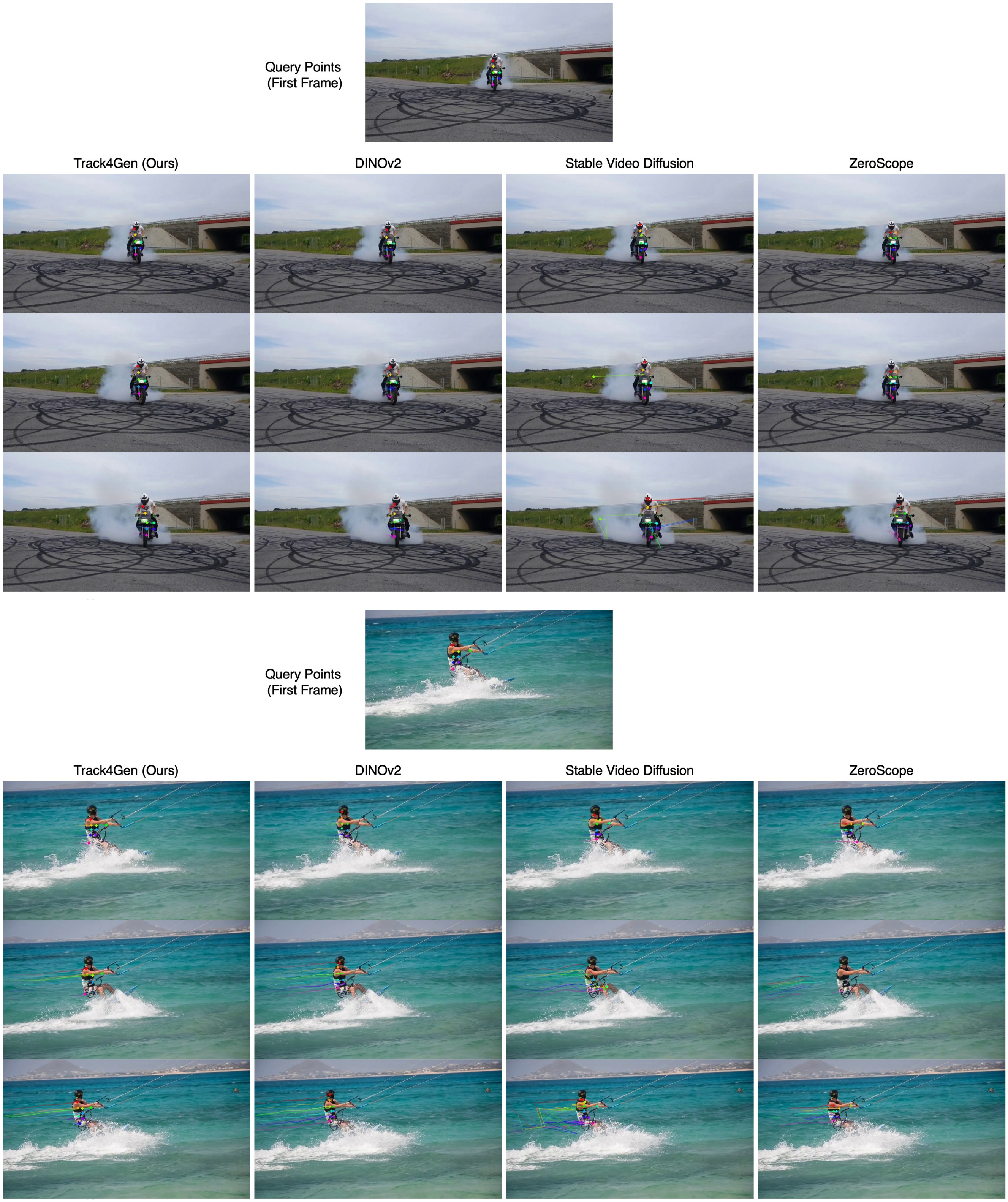


Figure 6. **Additional feature comparison on real-world video tracking:** Track4Gen vs DINOv2 vs Stable Video Diffusion vs ZeroScope

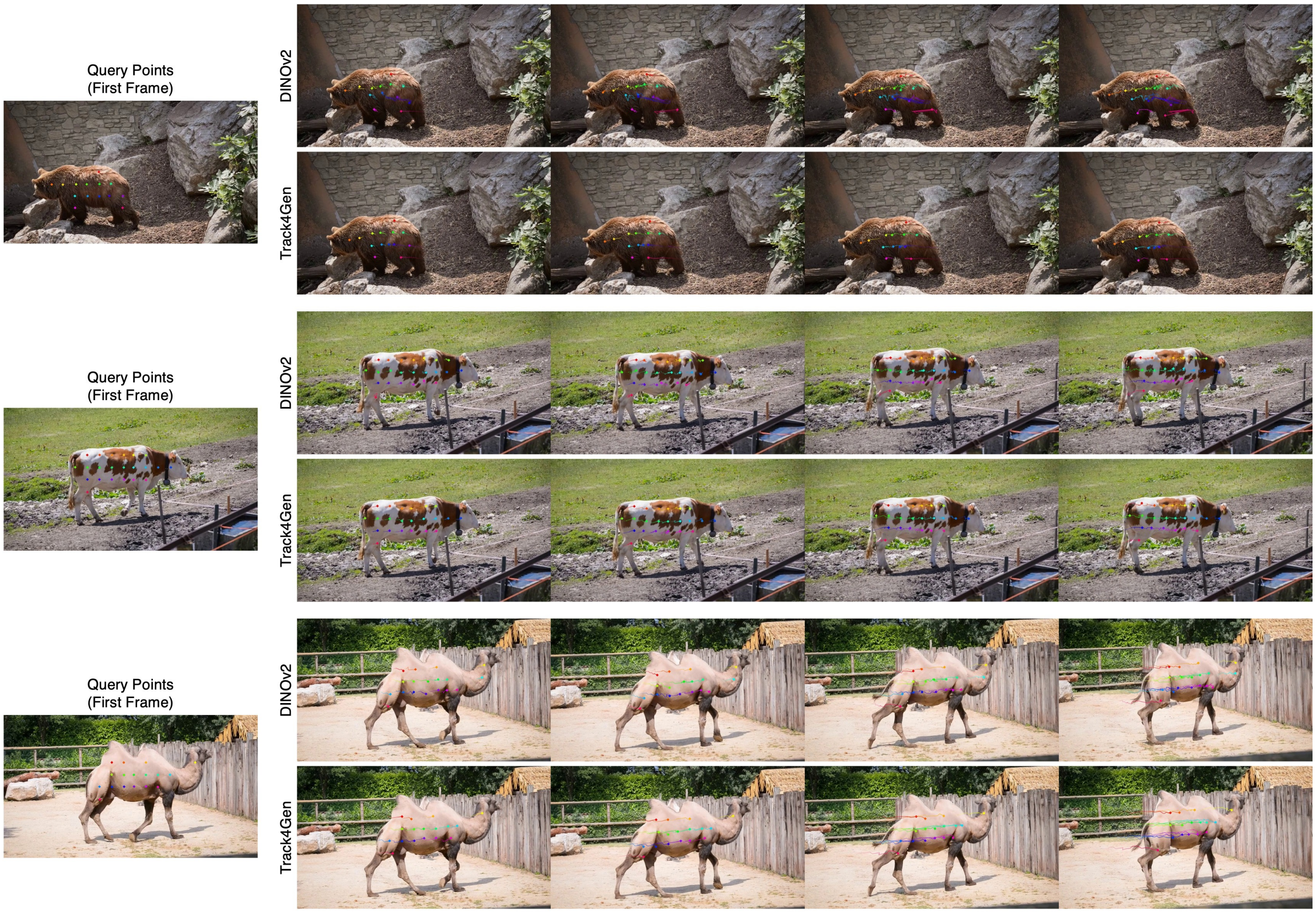


Figure 7. **Additional feature comparison on real-world video tracking:** Track4Gen vs DINOv2

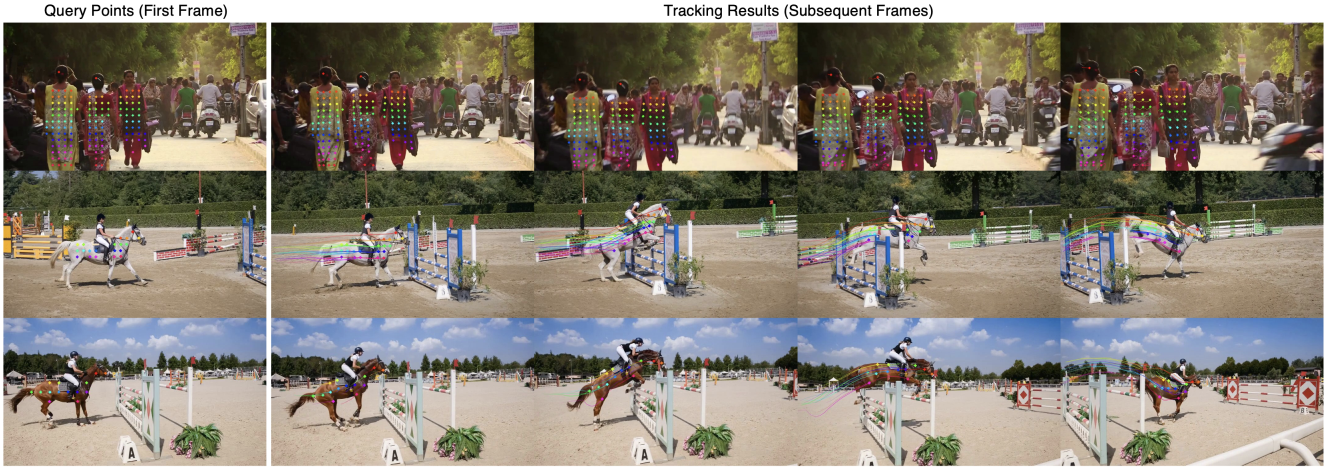

Query Points (First Frame)
Tracking Results (Subsequent Frames)

Figure 8. **Extending Track4Gen features with test-time adaptation [11].**

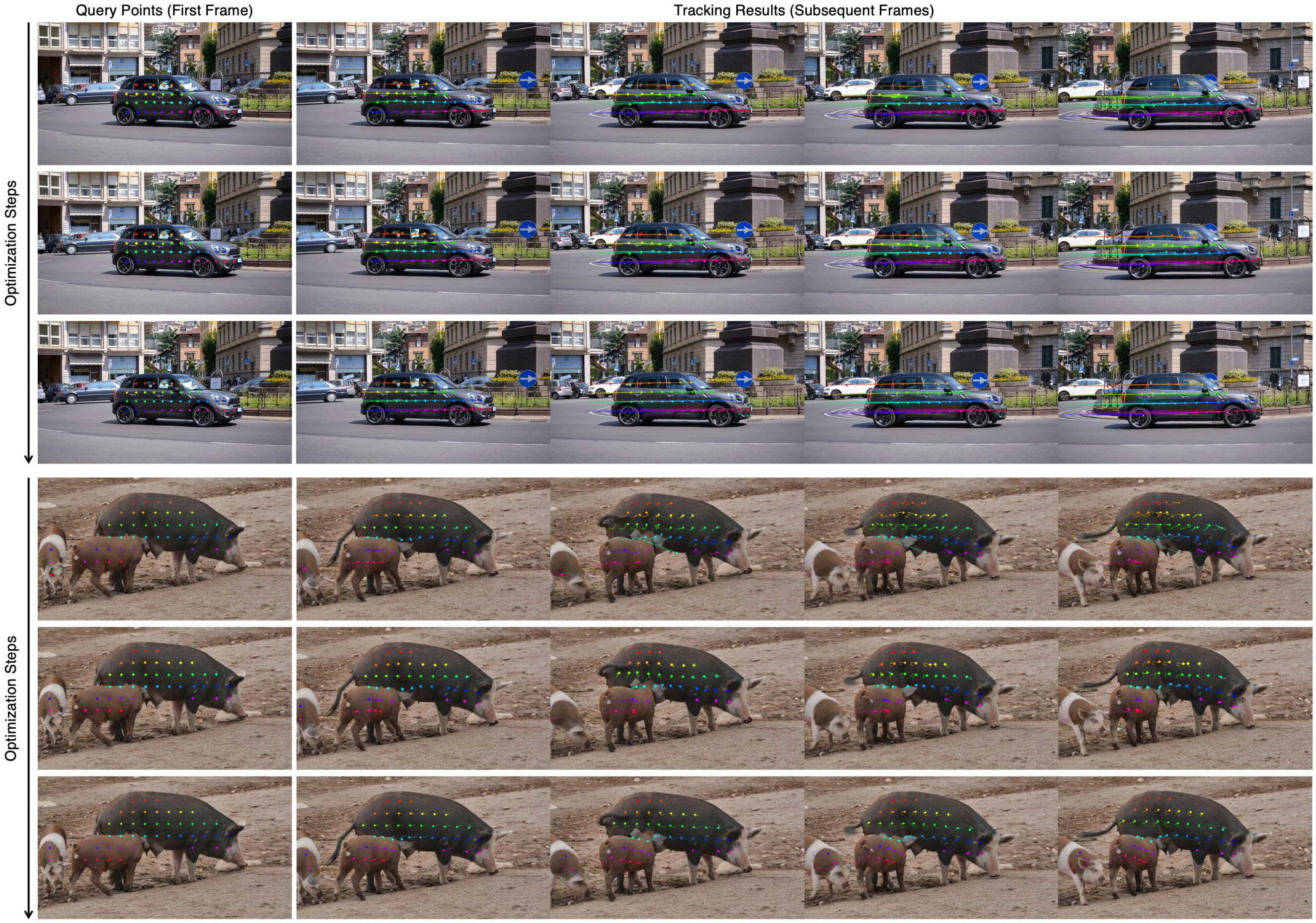

Query Points (First Frame)
Tracking Results (Subsequent Frames)
Optimization Steps
Optimization Steps

Figure 9. **Optimization progress visualization.** The first rows show tracking results using zero-shot Track4Gen features, while the third rows display results after 5,000 optimization steps.

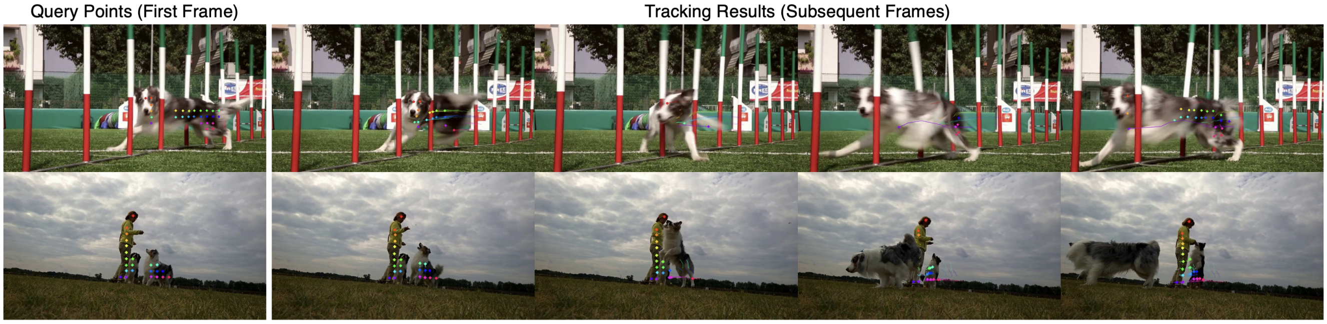

Query Points (First Frame)
Tracking Results (Subsequent Frames)

Figure 10. **Video tracking failure cases.** Track4Gen features struggle to capture point correspondences in videos with fast-moving objects or multiple semantically similar objects.

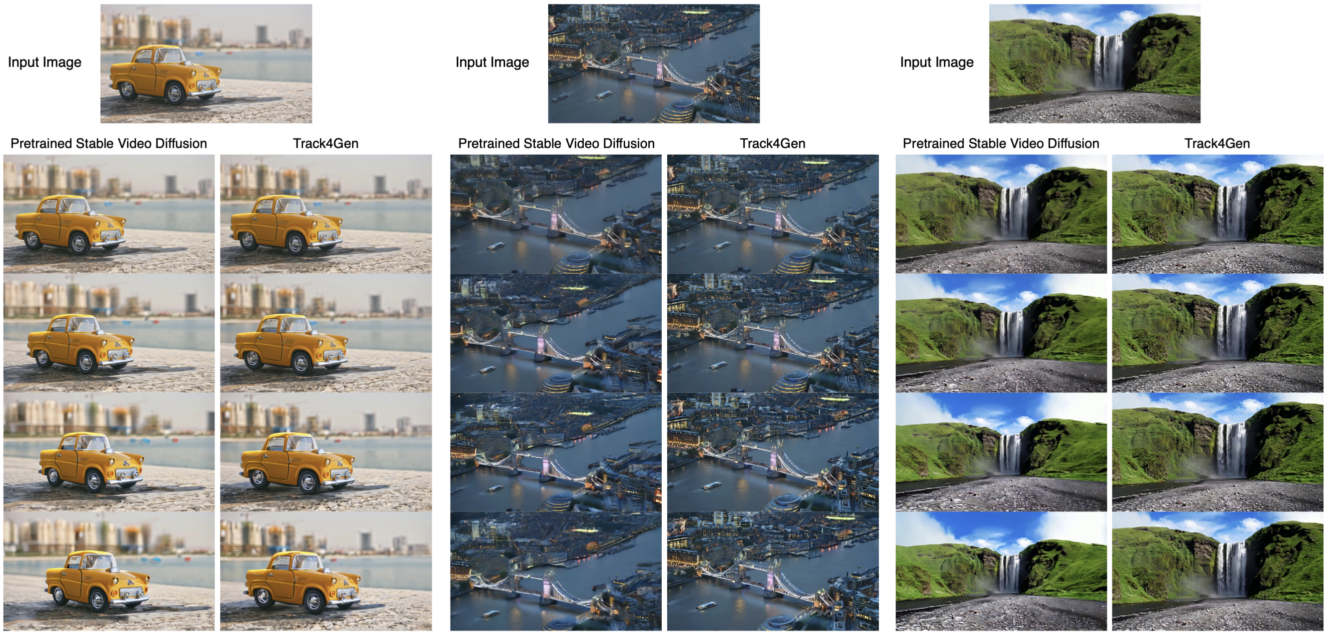

Input Image
Input Image
Input Image
Pretrained Stable Video Diffusion
Track4Gen
Pretrained Stable Video Diffusion
Track4Gen
Pretrained Stable Video Diffusion
Track4Gen

Figure 11. **Limitation.** Generated videos of Track4Gen may exhibit reduced camera motion.

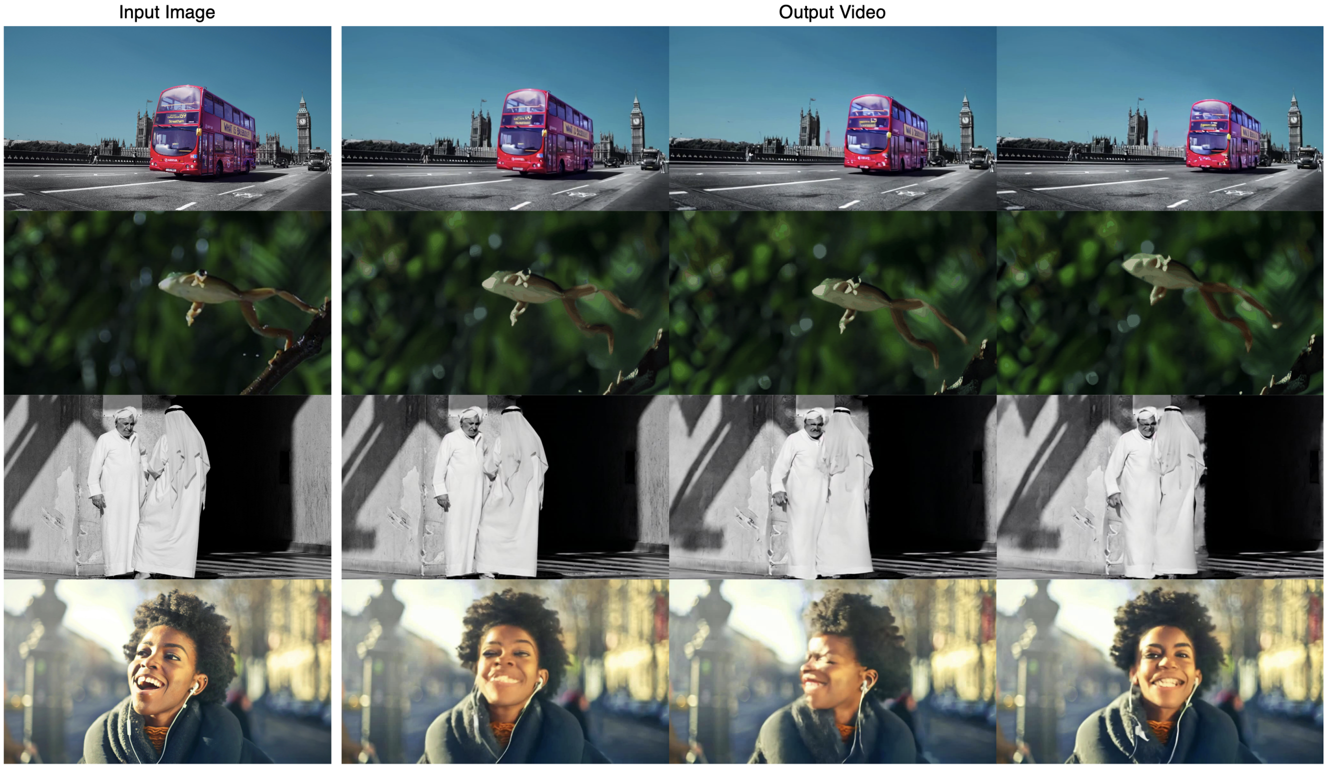


Figure 12. **Video generation failure cases.** Track4Gen may generate videos with physically unrealistic motion and artifacts on human faces. For instance, the red bus (row 1) drives backward, the frog (row 2) jumps mid-air, and the faces (row 3,4) display artifacts.

References

[1] Shir Amir, Yossi Gandelsman, Shai Bagon, and Tali Dekel. Deep vit features as dense visual descriptors. *arXiv preprint arXiv:2112.05814*, 2(3):4, 2021. 2

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

[3] 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

[4] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021. 2

[5] 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. 1

[6] Mingi Kwon, Seoung Wug Oh, Yang Zhou, Difan Liu, Joon-Young Lee, Haoran Cai, Baqiao Liu, Feng Liu, and Youngjung Uh. Harivo: Harnessing text-to-image models for video generation. *arXiv preprint arXiv:2410.07763*, 2024. 3

[7] 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

[8] 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

[9] Spencer Sterling. Zeroscope, 2023. `https://huggingface.co/cerspense/zeroscope_v2_576w`. 2

[10] 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. 1

[11] 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. 1, 2, 8

[12] 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. 1

[13] Junyi Zhang, Charles Herrmann, Junhwa Hur, Luisa Polania Cabrera, Varun Jampani, Deqing Sun, and Ming-Hsuan Yang. A tale of two features: Stable diffusion complements dino for zero-shot semantic correspondence. *Advances in Neural Information Processing Systems*, 36, 2024. 2

[14] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018. 2