# AnimateAnything: Consistent and Controllable Animation for Video Generation

## Supplementary Material

In the supplementary materials, we provide additional details about the metrics, methods, and training process, along with a more extensive and diverse set of qualitative results. Since our work focuses on video generation, we also present the corresponding video outputs on an anonymous website (please click https://yu-shaonian.github.io/Animate_Anything/). This allows for a clearer demonstration of the overall visual quality of generated videos and provides a clearer view of temporal-scale variations.

## A. Trajectory Evaluation

To evaluate the accuracy of camera motion guidance, we first extract the camera trajectories of generated videos and then apply two metrics to evaluate the alignments between the input camera trajectories and the extracted ones.

### A.1. Trajectory Extraction

Since estimating camera poses inherently involves potential errors and uncertainties, to ensure the accuracy of pose estimation and maintain fairness in comparison, we employed three different methods to estimate camera poses and compared them with corresponding poses from similar methods, evaluated under the same conditions. The adopted camera pose estimation methods are VGGSfM [4], DUSt3R [5], and ParticleSfM [8]. It should be noted that CameraCtrl [1] had estimated camera poses directly using the classical COLMAP [3]. However, it achieved only about a 20% success rate when tested in real-world scenarios. [1]

### A.2. Evaluation Metrics

Similar to He et al. [1], Wang et al. [6], we adopt *rotation error* and *translation error* to evaluate the scale and differences in the rotation and translation terms of the camera matrix. The specific evaluation metrics are as follows:

- *Rotation Error* $R_{err}$: The relative rotation distances are then converted to radians, and we sum the total error across all frames,

$$R_{\mathbf{err}} = \sum_{\mathbf{i=1}}^{\mathbf{n}} \mathbf{arcos}\left(\frac{\mathbf{tr}(R_{\mathbf{out_i}}^{\mathbf{T}} R_{\mathbf{gt_i}}) - \mathbf{1}}{\mathbf{2}}\right) \tag{1}$$

- *Translation Error* $T_{err}$: The norm of the relative translation vector for each frame is also summed together to form the translation error of the whole video,

$$T_{\mathbf{err}} = \sum_{\mathbf{i=1}}^{\mathbf{n}} \|T_{\mathbf{out_i}} - T_{\mathbf{gt_i}}\|_{\mathbf{2}} \tag{2}$$

[1] We have confirmed this point with the original authors of CameraCtrl.

## B. Objects Controllability Evaluation.

Following DragAnything [7], we evaluate object motion control (ObjMC) by computing the Euclidean distance between ground truth and predicted object trajectories extracted using Co-Tracker [2]. Tab. A demonstrates the effectiveness of our approach in object-level motion control.

## C. More Details For Frequency Stabilization

As shown in Fig.4, we apply the Fast Fourier Transform to the Query (Q), Key (K), and Value (V) of the 3D Full Attention mechanism in each DiT block. Next, the frequency domain features are modulated by the weight matrix $W$ with a pixel-wise production to adjust the distributions of different features. After modulation, we perform the inverse Fast Fourier Transform to revert them to the original domain. Finally, we execute the standard Scaled Dot-Product Attention operation to obtain the output of our frequency-based stabilization module. This method enhances the model's ability to capture frequency-based interactions among tokens in the input sequence as shown in Fig. B. We observed that incorporating these learnable frequency stabilization modules significantly enhanced the model's stability. However, directly analyzing the changes in the weight matrices proved to be quite challenging. We have attempted to analyze the functional mechanisms of the weight matrix $W$ using numerical or visual analytical approaches, but the interpretability of these frequency-domain interaction mechanisms remains an open research question due to the inherent challenges in accurately characterizing spectral features.

## D. More Training Details

We conducted experiments on a server equipped with $8\times$ NVIDIA Tesla A800 80G GPUs. We use AdamW optimizer for both the flow generation and video generation stages. In the first stage, we train the flow generation network with a learning rate $1e^{-4}$, batch size 1 per GPU, and a linear warming up of 500 steps. The whole training takes 50,000 steps and training with fixed-camera direction takes 10,000 steps. In the second stage, we train the frequency-based stabilization module, the flow encoder and the Vit block of our base network for 20,000 steps with batch size 2 per GPU to enable the video generation network to support our unified optical flow input.

## E. More Qualitative Results

In this section, we provide additional qualitative experiments to further demonstrate the effectiveness of our

Table A. Objects controllability evaluation.

| | DragAnything | MotionCtrl | Motion-I2V | MOFA-Video | DragNUWA | Ours |
|---|---|---|---|---|---|---|
| ObjMC↓ | 382 | 432 | 397 | 351 | 405 | 315 |

Figure A. combining both camera and explicit control.

method.

**Various User Annotations.** Our methods support diverse and fine-grained user motion annotation for detailed motion control. In Fig. C, we demonstrate the generated results of controlling fine-grained local motion of a toy bear with arrow-based motion annotation. The results show that our method is capable of subtle annotations and can accurately generate corresponding motion videos while maintaining a stable background.

**Camera Guidance.** In Fig. D, we showcase additional results of camera motion control, including the generation of various camera trajectories for the same outdoor building. We also present examples of applying diverse camera motions to dynamic outdoor scenes shown in Fig. E.

**Human Face Animation.** In Fig. F, we demonstrate that our method can effectively drive facial motions with different reference face images, using optical flows extracted from a reference facial motion video.

**Style Transfer.** Fig. G showcases the inherent capability of our method to perform video style transfer without any additional refinement. We first extract the optical flow and the first frame of a reference video, and then utilize the flow to animate the transformed version of the specific frame. The high-quality generated videos demonstrate the effectiveness of our method in the video style transformation task.

## F. Limitation and Improvement Strategies

Our research on video generation using unified optical flow representation encounters several technical and practical challenges. First, the accuracy of pre-trained optical flow estimation models can significantly impact the quality of the generated videos. Second, completely decoupling camera motion from object motion is an extremely challenging task as shown in Fig. A, and this process heavily relies on high-quality datasets. Furthermore, the diversity and quality of the training data may also influence the model's performance in specific applications. To tackle these issues, we will investigate methods to enhance the model training process, including utilizing higher-quality datasets and implementing improved decoupling strategies, to boost the accuracy and performance of video generation.

## G. Social Impact

Our research utilizes a unified optical flow representation to achieve camera motion or drive arbitrary subjects in video generation. This technology will enhance the efficiency of video production, fostering innovation in industries such as film and gaming while creating more engaging entertainment and educational experiences. However, as the technology becomes more widespread, society may face ethical and legal challenges, including issues related to the authenticity of video content and copyright. Therefore, establishing appropriate management regulations to ensure the responsible use of this technology is an important task that we must address.

Figure B. Visual examples of frequency stabilization(Top without)

## References


[1] Hao He, Yinghao Xu, Yuwei Guo, Gordon Wetzstein, Bo Dai, Hongsheng Li, and Ceyuan Yang. Cameractrl: Enabling camera control for text-to-video generation. *arXiv:2404.02101*, 2024. 1

[2] Nikita Karaev, Ignacio Rocco, Benjamin Graham, Natalia Neverova, Andrea Vedaldi, and Christian Rupprecht. Cotracker: It is better to track together. In *Proc. ECCV*, 2024. 1

[3] Johannes L. Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *CVPR*, 2016. 1

[4] Jianyuan Wang, Nikita Karaev, Christian Rupprecht, and David Novotny. Vggsfm: Visual geometry grounded deep structure from motion. In *CVPR*, 2024. 1

[5] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *CVPR*, 2024. 1

[6] Zhouxia Wang, Ziyang Yuan, Xintao Wang, Yaowei Li, Tianshui Chen, Menghan Xia, Ping Luo, and Ying Shan. Motionctrl: A unified and flexible motion controller for video generation. In *SIGGRAPH*, 2024. 1

[7] Weijia Wu, Zhuang Li, Yuchao Gu, Rui Zhao, Yefei He, David Junhao Zhang, Mike Zheng Shou, Yan Li, Tingting Gao, and Di Zhang. Draganything: Motion control for anything using entity representation. In *ECCV*, 2024. 1

[8] Wang Zhao, Shaohui Liu, Hengkai Guo, Wenping Wang, and Yong-Jin Liu. Particlesfm: Exploiting dense point trajectories for localizing moving cameras in the wild. In *ECCV*, 2022. 1

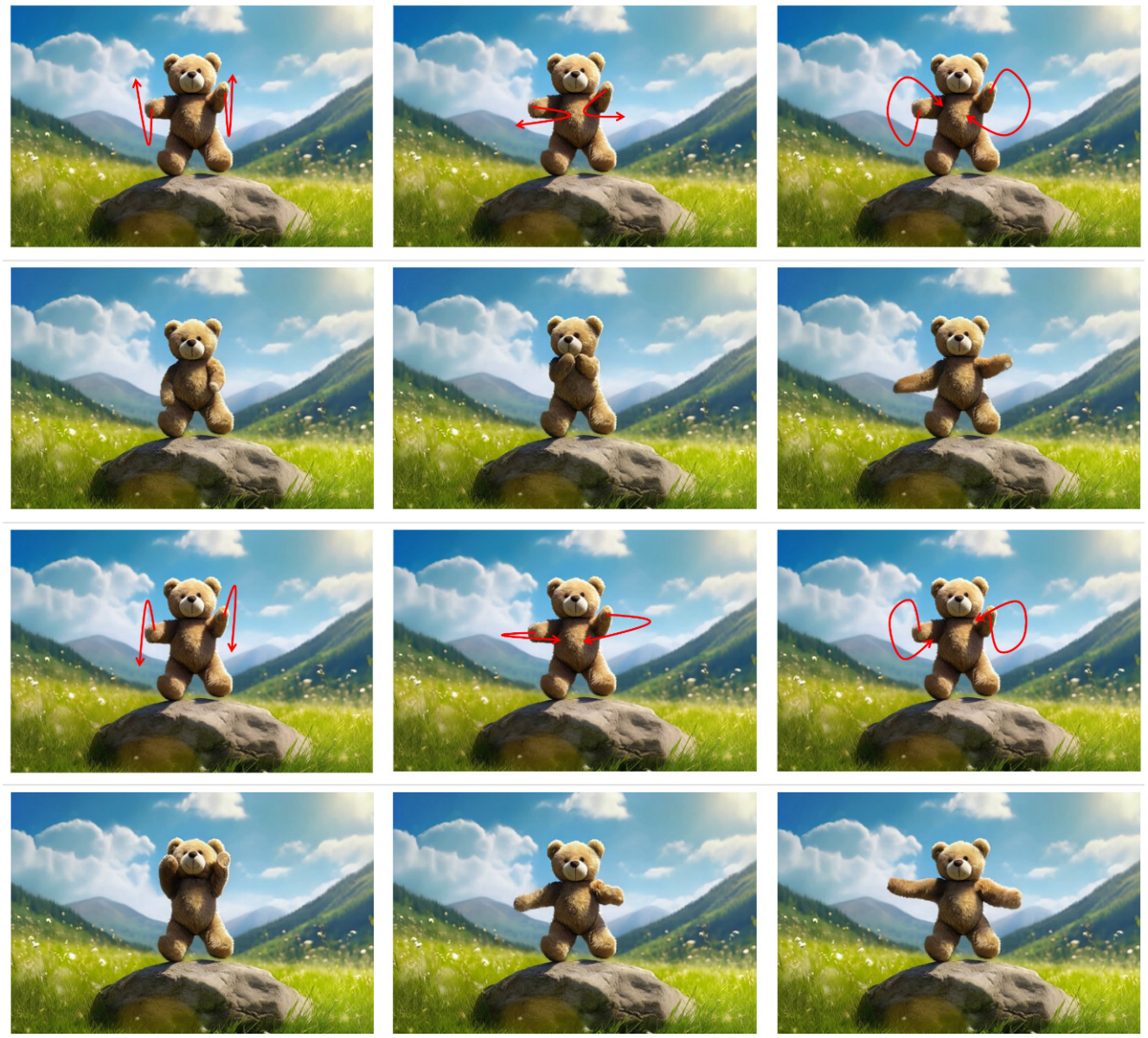

Figure C. Various kinds of User Annotations.

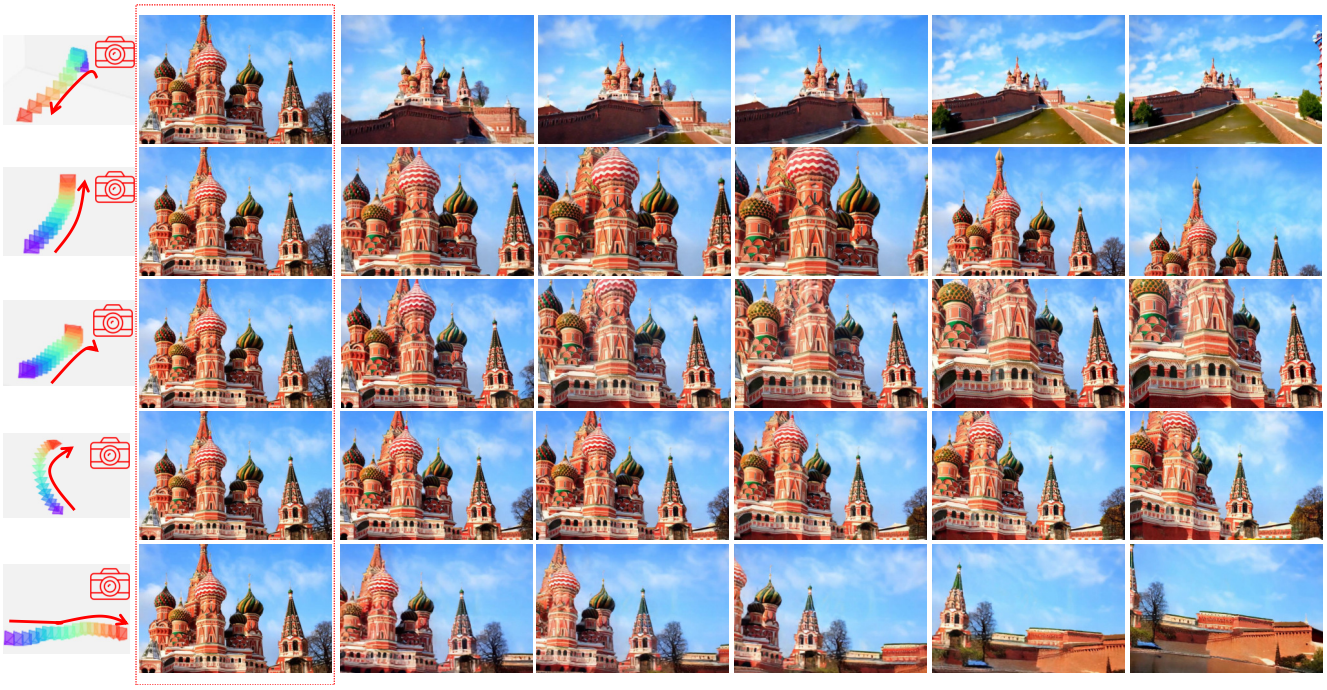

Figure D. The qualitative results of camera trajectory guided video generation with the same building image. The red box is the input reference images.

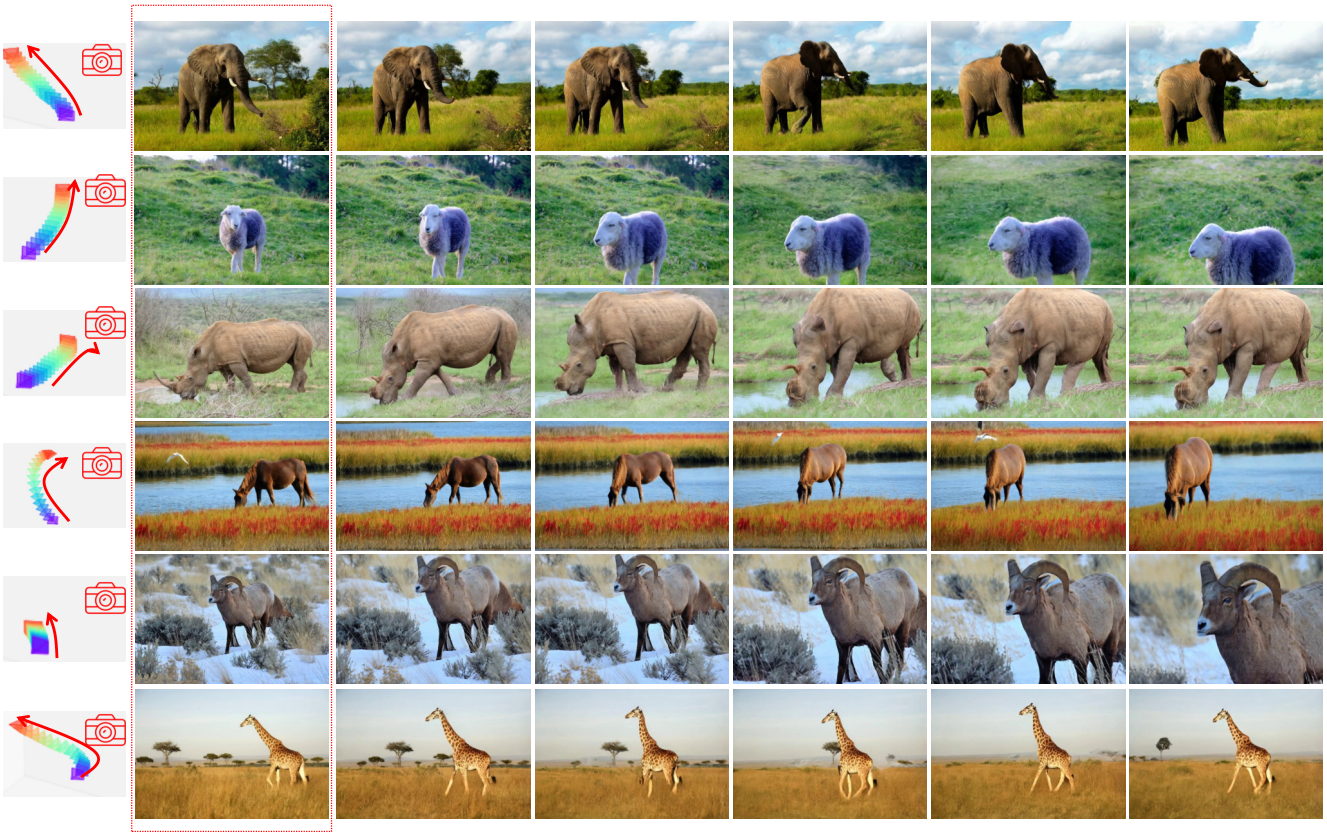

Figure E. The qualitative results of camera trajectory guided video generation on wild dynamic animals. The red box is the input reference images.

Figure F. The qualitative results of human face animation driven by the same facial motion video (Top). The red box is the reference image and the estimated optical flow maps.

Figure G. The qualitative results of video style transfer. The red boxes are the input reference images.