CamMimic: Zero-Shot Image To Camera Motion Personalized Video Generation Using Diffusion Models

Pooja Guhan[1] Divya Kothandaraman[1,2] Tsung-Wei Huang[2] Guan-Ming Su[2]
Dinesh Manocha[1]
[1] University of Maryland College Park
{pguhan, dkr, dmanocha}@umd.edu
[2] Dolby Laboratories USA
{tsung-Wei.Huang, guanming.su}@dolby.com
https://cammimic.github.io

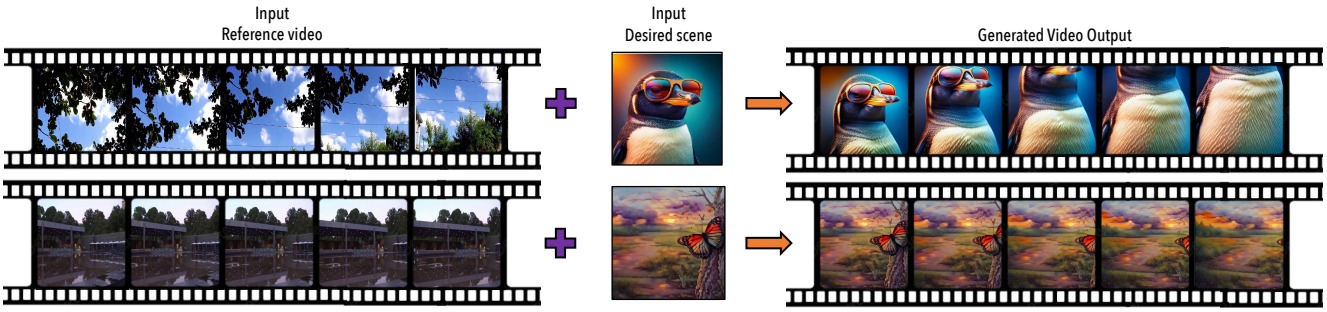

Figure 1. CamMimic aims to transfer an arbitrary camera motion observed in *a reference video* onto *an image* of a user's desired scene in a *zero-shot* manner without using any additional 3D details or camera information. Each row represents a *single* set of reference video and image given to obtain the corresponding output video. For each example, we can observe that CamMimic seamlessly transfers the camera motion in the reference video as well as preserves the image features of the desired scene and constructs unknown aspects further meaningfully to generate the output video.

Abstract

We introduce CamMimic, an innovative algorithm tailored for dynamic video editing needs. It is designed to seamlessly transfer the camera motion observed in a given reference video onto any scene of the user's choice in a zero-shot manner without requiring any additional data. Our algorithm achieves this using a two-phase strategy by leveraging a text-to-video diffusion model. In the first phase, we develop a multi-concept learning method using a combination of LoRA layers and an orthogonality loss to capture and understand the underlying spatial-temporal characteristics of the reference video as well as the spatial features of the user's desired scene. The second phase proposes a unique homography-based refinement strategy to enhance the temporal and spatial alignment of the generated video. We demonstrate the efficacy of our method through experiments conducted on a dataset containing combinations of diverse scenes and reference videos containing a variety of camera motions. In the absence of an established metric for assessing camera motion transfer between unrelated scenes, we propose CameraScore, a novel metric that utilizes homography representations to measure camera motion similarity between the reference and generated videos. Extensive quantitative and qualitative evaluations demonstrate that our approach generates high-quality, motion-enhanced videos. Additionally, a user study reveals that 70.31% of participants preferred our method for scene preservation, while 90.45% favored it for motion transfer. We hope this work lays the foundation for future advancements in camera motion transfer across different scenes.

1. Introduction

Personalized image [17, 29, 43] and video generation [15, 54] has transformed content creation by enabling the seamless integration of unique, custom concepts into pretrained models. This allows creators to bring their imagination

to life, incorporating personalized elements such as distinct characters [61], backgrounds [27], and tailored motions [33, 52] that were never part of the model's original training data. These innovations have significant applications in content creation, that include but are not limited to animation, storytelling, and filmmaking [47].

As digital content continues to grow and gain widespread popularity, video creators are increasingly driven to craft experiences that not only captivate but also deeply resonate with their audiences. A crucial element in achieving this lies in the thoughtful selection and execution of camera motions. *Camera motions*, which fundamentally represents the visual shifting and movement of the camera in a scene to shape the viewer's perspective, evoke emotional responses and add depth and realism to the visual narrative [2]. The way a camera moves, whether it be sweeping pans, steady tracking shots or dramatic zooms, can profoundly impact how a story is perceived and felt [4, 5]. At the core of aligning a video's visual storytelling with its intended emotional tone and narrative direction is the need for *exemplar-based camera motion personalization*. This enables video creators to effortlessly transfer camera movements observed in a reference video onto their own scene of interest, eliminating the reliance on expensive equipment, technical expertise or the need for substantial production time [6]. Beyond democratizing video production further, this technology grants creators the flexibility to experiment freely, refine their artistic vision and push the creative boundaries of their craft.

While advanced video foundation models [3, 48] now support the generation of high-definition videos with cinematic camera movement effects based on nuanced text prompts, the camera motions are still not personalized based to a single reference video and are generic to the text prompt [23]. Closely related motion transfer methods such as COMD [21] and AnimateDiff [15], transfer motion by training a motion module using multiple reference videos to learn motion representations, which are then provided as control signals to text-to-video diffusion backbone models. The usage of multiple reference videos is prohibitive for the camera motion personalization task due to differences in angles, and rates at which camera motions occur in different videos. These methods are also *not effective* in transferring the camera motion to a specific scene of interest.

Transferring camera motions across scenes typically requires access to unknown details of depth, structure, and geometry of the single target image - a task that is far from trivial and has not been actively explored. Replicating a reference video's camera motion in a different scene with potentially different structures demands a sophisticated understanding of spatial relationships within both the reference video and target scenes. Unlike works like DreamVideo [51], where motion transfer within a consistent scene allows depth and spatial alignment to be inherently maintained, or methods that transfer local or object motions guided by keypoint or pixel tracking, camera motion transfer necessitates a comprehensive 3D spatial understanding. Moreover, applying dynamic camera motions requires not just reproducing the spatial trajectory but also adapting to perspective shifts and parallax effects to ensure smooth visual coherence. This involves generating consecutive video frames that allow for smooth transitions along the camera path while maintaining scene consistency and realism.

Main Contributions:

In response to these limitations, we introduce CamMimic, the first approach designed to transfer camera motion from *a reference video to a user-provided image* in a *zero-shot manner*, without any additional data such as camera trajectories or 3D information. CamMimic introduces a novel inference-time optimization strategy on a pretrained text-to-video model. This involves a dual LoRA network-based fine-tuning process that separates the spatial and temporal content of the reference video and integrates the user's scene by imposing orthogonality between the learned spatial and temporal information. CamMimic further leverages homography representations from classical computer vision to guide the generated video along the desired camera motion.

To enable the evaluation of camera motion transfer, we introduce a new homography-based metric, *CameraScore*, to measure the temporal consistency between the camera motion in the generated video and that of the reference video, as no existing metric currently addresses this task. We curated a dataset consisting of 680 unique combinations of reference videos, encompassing a diverse range of camera motions, along with synthetically generated images depicting various types of scenes.

We perform extensive qualitative and quantitative analyses of our method, comparing it with existing work. Additionally, our user study shows a 70% preference over prior methods for preserving scene consistency with the user scene and an 89.75% preference for accurate camera motion transfer.

In summary, we make the following contributions: (1) We introduce the novel task of camera personalization onto a scene, a task with great practical implications. (2) We propose a novel method, CamMimic, with innovative training and inference strategies designed to address the challenges pertaining to the task. (3) We propose a new metric for evaluating camera motion personalization. (4) We conduct extensive analysis, revealing benefits over prior work (5) Additionally, in a bid to fortify further research in this direction, we curate and release a task-specific dataset.

2. Related Works

2.1. Video Personalization, Controllability, and Camera Motion Transfer

Recent advancements in pretrained video generation models, such as Stable Video Diffusion [11], ModelScope [49], SORA and Movie Gen [37], have enabled the creation of videos from text or image prompts. These models are typically trained on enormous amounts of image and video data, and are continuting to improve with newer diffusion and transformer modeling techniques. By leveraging these capabilities, video generation models can produce high-quality videos that reflect complex narratives and interactions, making them valuable tools in fields such as entertainment, education, and marketing.

This progress has fueled interest in enhancing control over video generation [15, 23, 26, 34, 38, 50, 60], particularly in areas like concept and object motion personalization, as well as camera motion transfer. Methods such as Tune-A-Video [38, 53], operate in a zero-shot manner, utilizing a pretrained model with a few representative images or videos for personalization. The video foundation model is fine-tuned using spatial and temporal LoRA [19] layers, facilitating effective custom video generation. While they excel at transferring motion to subjects, they struggle with camera motion transfer due to biases introduced during fine-tuning and challenges in separating spatial scene characteristics from temporal camera dynamics.

Another set of methods [15, 20] incorporates an additional pretrained stage for a "motion module", which is a neural network that is trained on motion representations. However, this can create data dependency challenges. Techniques focusing on camera controllability [16, 20, 58] often require camera trajectory information, which can be difficult to obtain. Other methods such as ReCapture [58] rely on 3D models for pseudo-4D scene generation, leading to high data and computational demands. Some approaches, such as [24], assume the availability of natural language descriptions for desired camera motion, which may not be practical for all users. *In contrast, our method does not rely on any additional data, including 3D information, motion data, or camera trajectories*.

2.2. Image Personalization & Novel View Synthesis

Research in image personalization [7, 28, 46] has focused on object customization [12, 57], poses [32, 35], and styles [41, 56]. While effective, these methods face challenges when applied to videos for motion personalization, particularly for camera and object movement, due to the need to model the temporal dimension separately. In contrast, ControlNet [59] methods typically require substantial supervised data for finetuning before application to test cases. cases. Our objective, however is to *develop a generalizable data efficient zero-shot approach to transfer camera motions*.

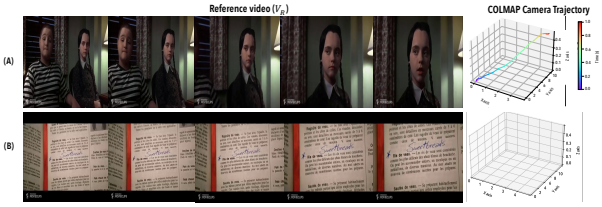

Figure 2. This showcases examples to depict scenarios where COLMAP can and cannot work reliably. In video (A), the camera effect observed is due to the explicit movement of the camera while in (B), the effect is obtained due to changes in camera focal length. COLMAP is not able to converge for videos like (B).

Transferring camera motion from a video to an image involves *generating a sequence of frames from new perspectives while maintaining temporal consistency*. This aligns with the objectives of novel view synthesis methods [30, 44], which have also been applied in ReCapture [58]. However, a significant challenge is the need for detailed camera trajectory information to determine the camera angle for the new scene, which can be limiting. Additionally, robust 4D novel view synthesis methods typically require extensive training data and often struggle with generalizability.

3. CamMimic: Our Approach

Problem Description: Given a *single* reference video V_R, a user-provided image I_u depicting the desired scene, and corresponding text prompts t_R and t_u, our goal is to generate a new video V_u by transferring the camera motion from V_R onto I_u in a zero-shot manner. We use a pretrained text-to-video diffusion model, e.g., zeroscope [1], as the backbone. t_R, the text prompt associated with V_R, is structured as a two-part comma-separated prompt: the first part describes the visual content of V_R, and the second part is the phrase 'camera motion of $< v >$'. $< v >$ is a random token, allowing our method to handle arbitrary and complex camera motions. The text prompt t_u associated with I_u describes the content of I_u.

Key challenges. Intuitively, camera motion transfer can be approached by first extracting camera trajectories using methods like COLMAP and then integrating them with the image using diffusion models. We observed two key issues: *(1)* Camera motions result from explicit movement or changes in focal length. COLMAP struggles with the latter, especially when the camera remains stationary but employs techniques like dolly zoom. *(2)* These methods require training specialized modules or encoders, making them impractical for personalization or scenarios with lim-

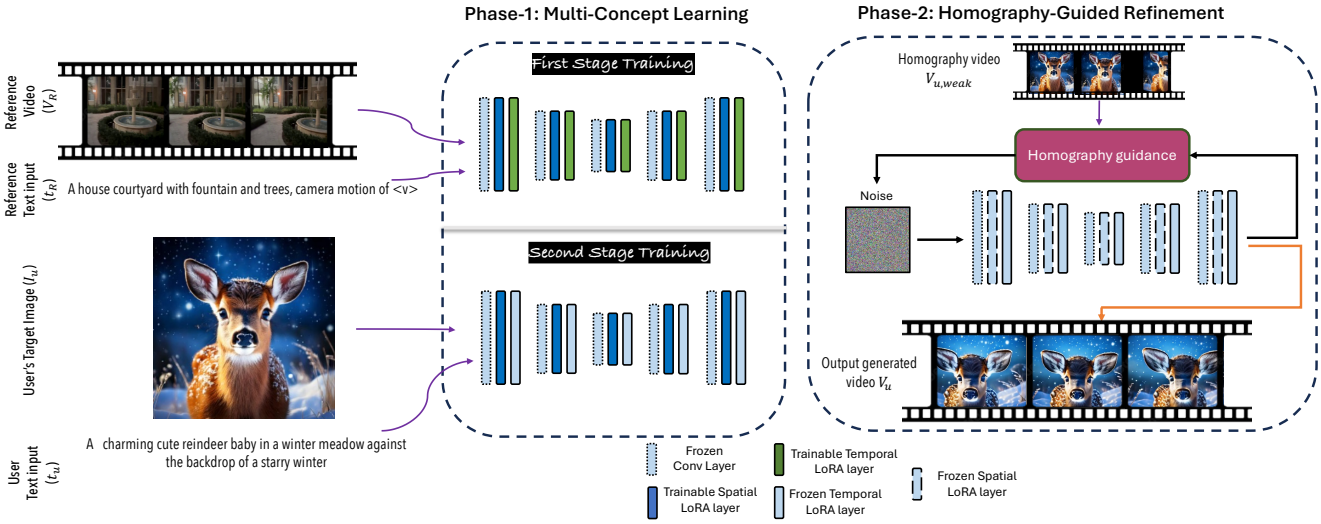

Figure 3. **CamMimic**: We present a *zero-shot* method to transfer camera motion visible in a reference video V_R onto a user provided image I_u to generate video V_u. It's a two-phase algorithm. The first phase involves learning multiple concepts associated with the spatial and temporal features of the reference video as well as the spatial characteristics of the user-provided image. We propose the use of a spatial-temporal orthogonality loss to better learn these concepts. The second phase consists of using a homography-based guidance to refine the generated video to preserve the scene in I_u as well as the camera motion obtained from the reference video V_R.

ited examples. To overcome these limitations and provide greater flexibility, we propose a novel *zero-shot* strategy.

Method overview. CamMimic operates in two key phases, as shown in Fig.3: (i) *Multi-concept finetuning* to disentangle the temporal features from the reference video and then blend the spatial characteristics of the user's target image, following a novel orthogonal loss guided fine-tuning regime; (ii) *Homography-guided inference* to ensure that the generated video preserves the user scene as well as the reference video camera motion.

3.1. Phase-1: Multi-Concept Finetuning

Network structure: Our proposed *zero-shot* algorithm's effectiveness hinges on its ability to discern and interpret the intricate relationships between three basic camera motion transfer (CMT) relevant concepts: the *spatial characteristics* of the reference video V_R, the *camera motion (temporal) features* of V_R and the *spatial characteristics* of the user-provided image I_u. We propose a *multi-concept* LoRA fine-tuning approach on the text-to-video diffusion backbone model to serve our purpose.

LoRA [19] layers in general, are designed to enable the efficient fine-tuning of large models with a small subset of parameters for task-specific adaptation. For a pretrained weight matrix, $W_0 \in \mathcal{R}^{d \times k}$, LoRA constrains its update by a low-rank decomposition:

$$W_0 + \Delta W = W_0 + BA \tag{1}$$

where $B \in \mathcal{R}^{d \times r}$, $A \in \mathcal{R}^{r \times k}$. rank r is much smaller than d and k. We propose using a combination of two types of LoRA matrices, namely *spatial* and *temporal* to understand the camera motion transfer concepts. Spatial LoRAs are integrated into spatial self-attention layers, while temporal LoRAs are incorporated into the temporal self-attention layers in the video diffusion backbone. As their name suggests, spatial LoRAs capture and interpret spatial characteristics, whereas temporal LoRAs focus on modeling camera motion features. We maintain the backbone's cross-attention layers unchanged to preserve its strong textual alignment capabilities.

Training Strategy: We follow a *two-stage* training paradigm. The *first stage* is to capture V_R's spatial and temporal concepts while the *second stage* focuses on meaningfully learning the spatial concept of I_u without tampering with the temporal concepts learned in the first stage.

First stage training: During each training step, we employ a differential training scheme. Temporal LoRAs are consistently trained on all frames of V_R, ensuring a comprehensive grasp of camera motion dynamics. In contrast, spatial LoRAs are trained on a single, randomly selected frame from V_R per step, promoting generalization across diverse spatial configurations. To further enhance regularization and prevent overfitting, we implement a random masking strategy for spatial LoRAs, inspired by [63]. This approach helps balance the learning process

between spatial and temporal aspects while maintaining model efficiency. The training loss can be formulated as:

$$\mathcal{L}_{first\_stage} = \mathcal{L}_{temporal} + \delta\mathcal{L}_{spatial} \quad (2)$$

where $\mathcal{L}_{temporal}$ and $\mathcal{L}_{spatial}$ refers to the noise prediction loss [14] being applied for training temporal and spatial LoRAs respectively and and $\delta \in (0, 1)$.

Second stage training: This training stage leverages the power of transfer learning to create dynamic video content based on a user-specified image I_u while maintaining the temporal characteristics of a reference video V_R.

In the *first stage of training*, the model learns the intricate relationship between spatial and temporal characteristics from V_R through the interaction of spatial and temporal LoRAs. This establishes a strong foundation for understanding how scenes evolve over time. The second stage of training focuses on adapting the spatial components of I_u to the learned spatial LoRA. Instead of training from scratch, we fine-tune the spatial LoRA [9], enabling the model to efficiently align I_u with the pre-learned temporal dynamics. However, to faithfully replicate the temporal characteristics of V_R, it is crucial to maintain the relative strength of both spatial and temporal LoRAs. Uncontrolled spatial fine-tuning could inadvertently suppress key temporal features, disrupting the learned motion dynamics. To address this, we introduce an *orthogonality loss*, ensuring that spatial features remain orthogonal to temporal features. This constraint preserves the strength of temporal dynamics while allowing the spatial characteristics of I_u to integrate seamlessly, maintaining the same spatial-temporal relationship observed in V_R.The second stage training loss can be defined as

$$\mathcal{L}_{second\_stage} = \mathcal{L}_{spatial} + \lambda\mathcal{L}_{ortho} \quad (3)$$

where

$$\mathcal{L}_{ortho} =< W_{spatial\_LoRA}, W_{temporal\_LoRA} > \quad (4)$$

and λ is a user-defined hyperparameter that weighs the impact of $\mathcal{L}_{ortho}$ during the second stage training. $W_{spatial\_LoRA}$ refers to the weight parameters of the spatial LoRA layers and $W_{temporal\_LoRA}$ refers to the weight parameters of the temporal LoRA layers.

3.2. Phase-2: Homography Guided Inference

To improve the synchronization between motion and appearance in the generated video V_u, we introduce a novel weak supervision strategy to guide the *denoising* of latents during the *inference* stage. We leverage *homography* [45], which has been proven effective for novel view synthesis [25], as a weak supervision signal to provide an additional *weak* reference of how the spatial locations of the different scene elements in I_u *might* change under the desired camera motion. *Homography* is a well-established computer vision technique that computes the transformation required to map one image onto another, accounting for spatial relationships such as rotation and translation. This transformation is typically derived by matching key points between images using feature descriptors like SIFT [31], followed by calculation via the RANSAC algorithm [13]. The key intuition driving this supervision strategy is that by offering this weak yet informative guidance, we enable the denoising latents to update in a more contextually aware manner.

The inference process begins by inputting a carefully crafted text prompt t_I into our fine-tuned text-to-video backbone model. t_I is designed to be t_u+'camera motion of $< v >$' (i.e., the second part of t_R). To guide the generation process, we compute frame-to-frame homography transformations using the reference video. For each consecutive frame pair in the reference video, we derive the homography matrix H_i for the transformation from frame $F_{R,i}$ to $F_{R,i+1}$. We then obtain a *psuedo weak* estimate of V_u, referred to as $V_{u,weak}$, using the computed homography matrices H_i with I_u as the first frame ($F_{P,1}$). The subsequent frames of $V_{u,weak}$ are obtained as $F_{P,i+1} = H_i(F_P, i)$ where $i \in [1, N]$. N is the total number of frames in V_R. $F_{P,i}$ refers to the i^{th} frame in $V_{u,weak}$.

At each stage of diffusion denoising, particularly during every sampling step (except the last), we adjust [18, 25] the predicted latents z_t as follows:

$$\hat{z}_t = z_t - \lambda_G \nabla(z_t - z_P)^2 \quad (5)$$

where z_P is the latents corresponding to F_P.

3.3. CameraScore - A New Metric

Popular metrics for motion transfer like optical flow [10] and COLMAP [42] face unique challenges when applied to our task of transferring camera motion between unrelated scenes. This is mainly because of issues such as *structural incompatibility, scaling differences, and video quality dependency* and *computation cost*. We elaborate on these issues in the Appendix. Therefore, considering these challenges, we propose a homography-based metric, *CameraScore*, as a *more suitable* approach for evaluating camera motion transfer across structurally different scenes. This can be computed as

$$\frac{1}{N}\sum_{i=1}^{N} ||\mathcal{H}_{R_i} - \mathcal{H}_{G_i}||^2 \quad (6)$$

where $\mathcal{H}_{R_i}$ is the i^{th} homography matrix obtained between the i^{th} and $(i+1)^{th}$ frame of the reference video. Similarly, $\mathcal{H}_{G_i}$ is the i^{th} homography matrix between the i^{th} and $(i+1)^{th}$ frames of the generated video.

Why Homography? Homography estimation is inherently designed to capture relative transformations between two planes, enabling it to effectively represent camera motions that include but are not limited to panning, tilting, and zooming, regardless of structural differences between the reference video and the target user scene. Its strength lies in capturing broad, global motion patterns rather than pixel-level details. Unlike optical flow, which relies on fine-grained, pixel-by-pixel correspondences and requires structural similarity across scenes, homography encapsulates the high-level transformation between views. This aligns directly with our goal: *to transfer holistic camera motion (such as broad directional shifts and zooms) rather than minor motion details*.

Additionally, by considering homography matrices obtained between every consecutive pair of frames, we capture local motions that often approximate planar transformations, even in complex environments. While each transformation between adjacent frames is largely planar or rotational in nature, the cumulative effect across the entire video sequence represents a more complex, non-linear path through the scene. This approximation proves more valuable for evaluation purposes as compared to other metrics based on COLMAP or optical flow.

4. Experiments and Results

4.1. Dataset

Currently, no dataset exists specifically for our task. For our experiments, we assembled a diverse dataset comprising 680 reference video and user scene combinations, aligning with dataset sizes used in works like DreamBooth [40], Imagic, and MotionDirector [62]. Our reference videos encompass various cinematic camera motions, including panning, tilting, zooming, and advanced effects like dolly shots, capturing both subtle and dynamic motion styles. These videos vary in motion speed, featuring smooth transitions and rapid shifts, enabling comprehensive evaluation across different dynamics. The dataset is curated from movies, dramas, animations, and documentaries, providing diverse motion patterns. Complementing these videos, user scenes were generated using state-of-the-art text-to-image models, covering a broad spectrum of environments, from serene landscapes to bustling urban settings. We employed BLIP to generate image captions, and three experienced annotators provided the first part of t_R, describing the reference video content.

Each reference video is approximately 2–4 seconds long, offering sufficient motion information for accurate analysis and replication. The designed diversity in both scene types and motion styles ensures rigorous testing across various video content, allowing us to evaluate our method's robustness, adaptability, and effectiveness in capturing and transferring camera motion. We will release this dataset post-acceptance.

4.2. Implementation Details

Model Details. In our experiments, we use *zeroscope* [1] as the text-to-video generation backbone. The number of frames sampled and generated per dataset sample is 16.
Training Details. For a given dataset sample consisting of the reference video V_R and the user-provided image I_u, we finetune all models using a *single* A5000 GPU. The training processes (both stages) are run for a total of 150 epochs each. It takes approximately 10 minutes to obtain the generated video V_u corresponding to the given V_R and I_u. We train all our networks using the Adam optimizer with a learning rate of $5e-4$. All codes were implemented using Pytorch [22].
Metrics. We use three metrics to evaluate our method to measure consistency or preservation of the scene with respect to the user image I_u, text consistency, and motion consistency with the reference video. The self-supervised similarity metric DINOv2 [36] and VideoCLIP [55] measure scene preservation and text consistency respectively. The *CameraScore* metric measures motion transfer.

4.3. Comparison with Prior Work

To our knowledge, ours is the first paper to address the problem of *transferring* camera motion from *a single video* onto a target user scene in a *zero-shot manner*. While current SOTA foundation models for video generation can introduce a camera motion in the videos they generate, they *do not support camera motion personalization*. CamMimic can enable them to do so. We therefore present experiments that compare our approach with other similar *zero-shot* methods from three related categories of potential solutions, each offering different ways of approximating this task. These include using traditional computer vision methods like homography, using DreamBooth [40] style training on an existing motion transfer work like Tune-A-Video [53] to introduce scene customization, and using video diffusion-based object motion transfer methods like MotionDirector [62]. To ensure a fair comparison, all chosen baselines are *zero-shot* and do not rely on additional modules such as ControlNet or motion modules. Additional details on the prior works considered for comparison, reasons for choosing them and modifications made (if any) to make them support camera motion transfer have been discussed in the appendix.

Fig. 4 present the *quantitative comparisons*. The homography baseline effectively replicates the reference video's camera motion but suffers from outpainting artifacts. Tune-A-Video maintains high scene consistency but exhibits minimal camera movement. MotionDirector [62] introduces slight camera motion while preserving

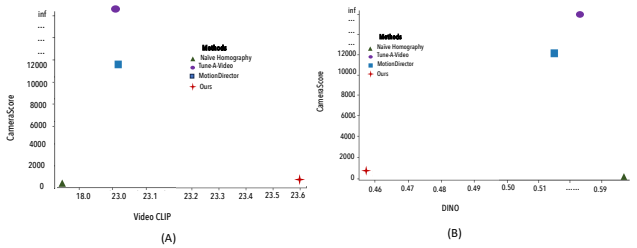

Figure 4. Graph (a) shows the results of the plot of Video CLIP scores with CameraScore for the different diffusion methods being compared. Graph (b) shows the plot of DINO scores with CameraScore for the different diffusion methods being compared. Our approach achieves the *best trade-off*.

scene consistency to some extent, but remains overall ineffective due to its inability to combine I_u spatial characteristics with V_R temporal features. Based on our observations, their strategy can lead to the suppression of temporal features. In contrast, our method successfully integrates both aspects, accurately mimicking the reference camera trajectory while maintaining scene fidelity to the user image. Our method achieves a higher VideoCLIP score than MotionDirector and Tune-A-Video, indicating superior textual consistency and alignment with scene content and camera motion. The lower CameraScore compared to MotionDirector demonstrates better camera motion transfer, whereas Tune-A-Video's lack of motion results in an inflated CameraScore and a lower VideoCLIP score due to its failure to adhere to the motion prompt. Additionally, our method attains a higher DINO score than MotionDirector, showcasing improved scene preservation with respect to the user image. Tune-A-Video, which largely replicates the user image across frames with minimal motion, naturally achieves a high DINO score, though it does not capture intended scene changes. Fig. 6 shows qualitative results which further validate our approach. Overall, our method, CamMimic, effectively transfers camera motion from the reference video while preserving the scene integrity of the user image, outperforming existing approaches.

4.4. Ablation Experiments

We also conducted ablation studies to evaluate the impact of the key components of our method: using a transfer learning paradigm to *learn user image I_u spatial characteristics* (this excludes the orthogonality loss), *introduction of an orthogonality loss function* to Phase-1 to minimize interference of I_u spatial features in suppressing V_R temporal features and ensure a similar relation between them as between V_R spatial and temporal features, and a *homography guidance* to enhance generation of V_u during inference (Phase-2). Fig. 5 shows the results obtained by ablating on one or multiple of these components. We observe that when USC is removed (and therefore no ORTHO as well), the

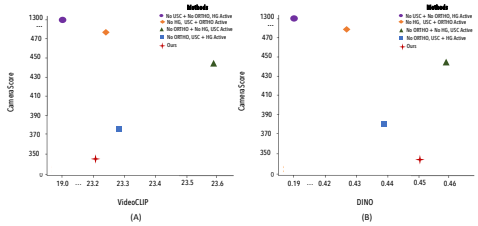

Figure 5. Graph (A) illustrates the relationship between CameraScore and DINO, while Graph (B) shows the relationship between CameraScore and VideoCLIP. These plots are derived from the ablation experiments conducted to evaluate the significance of various key components in CamMimic. Note: **USC** refers to the I_u spatial feature learning training within Phase-1 without considering the orthogonality loss (represented as USC in the graph). **ORTHO** corresponds to including the orthogonality loss function in Phase-1, while **HG** denotes the homography guidance used in Phase-2. Our approach, with all its components, achieves the *best* trade-off.

network loses access to critical information about the user-provided scene. Homography guidance, which incorporates the user's scene and camera motion cues, is designed to reinforce motion and scene dynamics rather than generate them from scratch. Homography guidance, which incorporates the user's scene and camera motion cues, is designed to reinforce motion and scene dynamics rather than generate them from scratch. Without USC, and thus without the user scene information, homography guidance becomes ineffective. As a result, motion transfer fails, leading to a significantly higher CameraScore. The VideoCLIP and DINO scores are better when we just use USC without ORTHO and HG. But as expected the camera motion transfer deteriorates as the temporal features get suppressed compared to when both ORTHO and HG are available. Without ORTHO, spatial features are better captured but the temporal features get disrupted as expected. Overall, our findings demonstrate that spatial blending and homography guidance are complementary. Together, they ensure effective motion transfer aligned with the reference video while preserving spatial scene integrity from the user-provided input.

4.5. User Study

To validate our camera motion transfer approach, we conducted a user study evaluating the accuracy of transferred camera motion and fidelity of user image characteristics. The goal was to reproduce the desired camera motion from a reference video while maintaining the visual integrity of the user's image. We engaged 72 participants, who reviewed 8 sets of examples, totaling 576 responses. Each set included a reference video, the user's desired scene, and outputs from three methods: MotionDirector, Tune-A-Video (DreamBooth modified), and our approach. Participants evaluated each output based on *scene similarity to the*

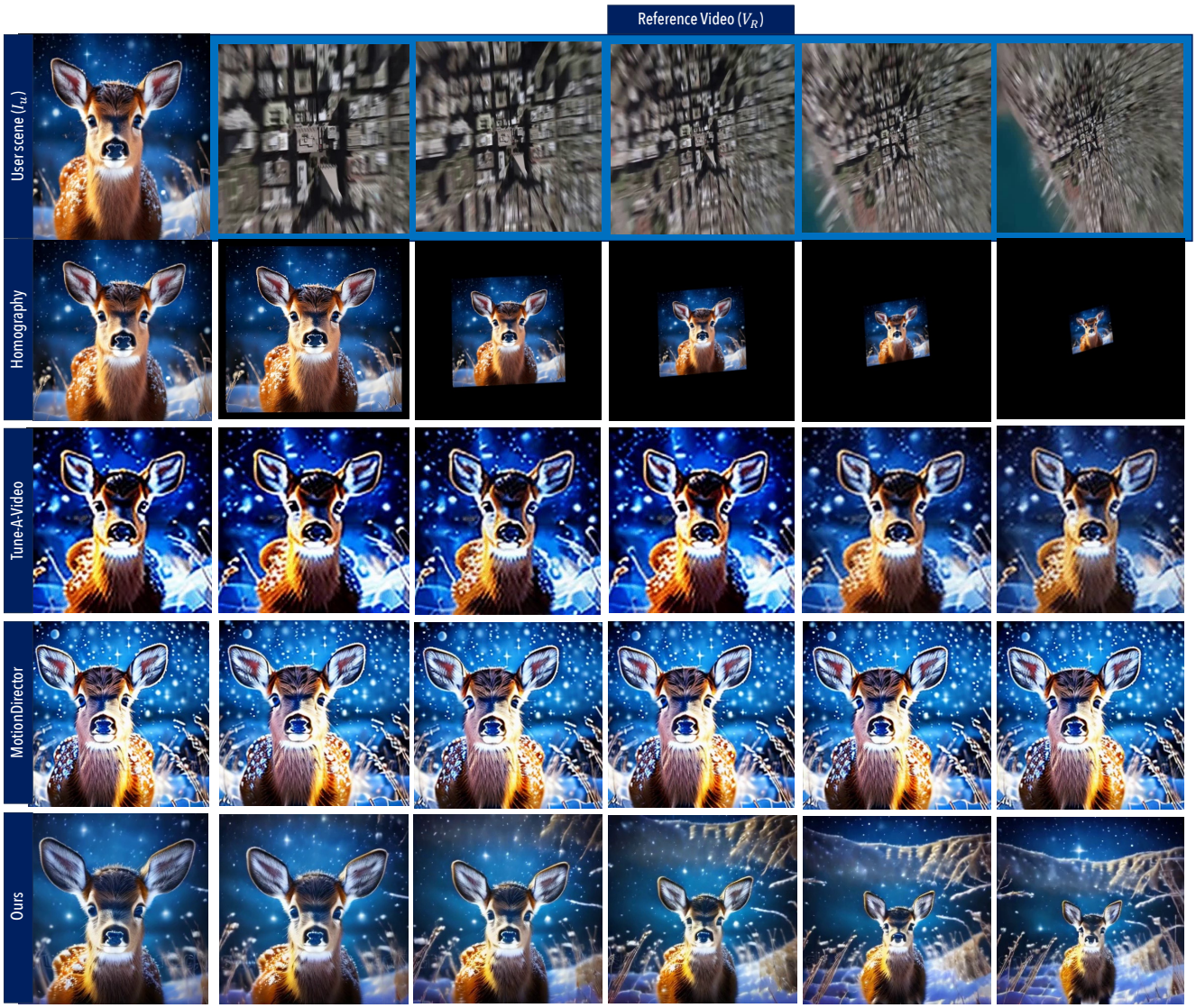

Figure 6. Qualitative results of our method, demonstrating clear improvements over prior work in transferring camera motion, while preserving scene content

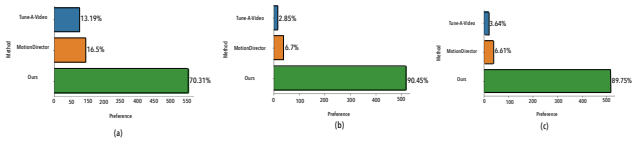

Figure 7. Graph (a) shows participants evaluating the similarity between the generated video scene and the user-provided scene. Graph (b) depicts participants' preferences for the output video with the desired camera motion among three methods. Graph (c) indicates participants' choice of the overall best output among the different methods.

user image, camera motion fidelity to the reference video, and overall best. Fig. 7 shows the results obtained from the study. We observe that our method was preferred across all three criteria by an overwhelming majority.

5. Conclusion, Limitations and Future Work

We present one of the first methods for transferring camera motion from a reference video to a user scene in a zero-shot manner without additional data. Qualitative and quantitative evaluations, including a new metric for camera motion transfer, demonstrate the superiority of our method over prior work. Our approach is highly generalizable, applying to any source image and reference video using the pre-trained model without extra information. Our method

has a few limitations that suggest future directions: (i) The homography matrix may have errors with moving objects in the reference video; using an LLM-based object detector could help recognize these objects. (ii) Extending our method to user videos with moving objects instead of static images would be interesting. (iii) As new video foundational models emerge, CamMimic can serve as a plug-and-play method to generate higher quality videos with more complex camera motions and scene content.

References

[1] Zeroscope, 2023. 3, 6

[2] Camera movement terms: Everything you need to know, 2024. 2

[3] Movie gen: A cast of media foundation models, 2024. 2

[4] How to use movement to tell a story in your videography, 2024. 2

[5] Videographer tips: How to direct your audience with camera movement, 2024. 2

[6] The challenges of shooting photography and videography, 2025. 2

[7] Yuval Alaluf, Elad Richardson, Gal Metzer, and Daniel Cohen-Or. A neural space-time representation for text-to-image personalization. *ACM Transactions on Graphics (TOG)*, 42(6):1–10, 2023. 3

[8] Dawit Mureja Argaw, Fabian Caba Heilbron, Joon-Young Lee, Markus Woodson, and In So Kweon. The anatomy of video editing: A dataset and benchmark suite for ai-assisted video editing. In *European Conference on Computer Vision*, pages 201–218. Springer, 2022. 1

[9] Nader Asadi, Mahdi Beitollahi, Yasser Khalil, Yinchuan Li, Guojun Zhang, and Xi Chen. Does combining parameter-efficient modules improve few-shot transfer accuracy? *arXiv preprint arXiv:2402.15414*, 2024. 5

[10] Steven S. Beauchemin and John L. Barron. The computation of optical flow. *ACM computing surveys (CSUR)*, 27(3):433–466, 1995. 5

[11] 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. 3

[12] Xi Chen, Lianghua Huang, Yu Liu, Yujun Shen, Deli Zhao, and Hengshuang Zhao. Anydoor: Zero-shot object-level image customization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6593–6602, 2024. 3

[13] Konstantinos G Derpanis. Overview of the ransac algorithm. *Image Rochester NY*, 4(1):2–3, 2010. 5

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

[15] 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. 1, 2, 3

[16] 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 preprint arXiv:2404.02101*, 2024. 3

[17] Zecheng He, Bo Sun, Felix Juefei-Xu, Haoyu Ma, Ankit Ramchandani, Vincent Cheung, Siddharth Shah, Anmol Kalia, Harihar Subramanyam, Alireza Zareian, et al. Imagine yourself: Tuning-free personalized image generation. *arXiv preprint arXiv:2409.13346*, 2024. 1

[18] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 5

[19] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021. 3, 4

[20] Teng Hu, Jiangning Zhang, Ran Yi, Yating Wang, Hongrui Huang, Jieyu Weng, Yabiao Wang, and Lizhuang Ma. Motionmaster: Training-free camera motion transfer for video generation. *arXiv preprint arXiv:2404.15789*, 2024. 3

[21] Teng Hu, Jiangning Zhang, Ran Yi, Yating Wang, Jieyu Weng, Hongrui Huang, Yabiao Wang, and Lizhuang Ma. Comd: Training-free video motion transfer with camera-object motion disentanglement. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 3459–3468, 2024. 2

[22] Sagar Imambi, Kolla Bhanu Prakash, and GR Kanagachidambaresan. Pytorch. *Programming with TensorFlow: Solution for Edge Computing Applications*, pages 87–104, 2021. 6

[23] Hyeonho Jeong, Geon Yeong Park, and Jong Chul Ye. Vmc: Video motion customization using temporal attention adaption for text-to-video diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9212–9221, 2024. 2, 3

[24] Hongda Jiang, Xi Wang, Marc Christie, Libin Liu, and Baoquan Chen. Cinematographic camera diffusion model. In *Computer Graphics Forum*, page e15055. Wiley Online Library, 2024. 3

[25] Divya Kothandaraman, Tianyi Zhou, Ming Lin, and Dinesh Manocha. Hawki: Homography & mutual information guidance for 3d-free single image to aerial view. *arXiv preprint arXiv:2311.15478*, 2023. 5

[26] Divya Kothandaraman, Kihyuk Sohn, Ruben Villegas, Paul Voigtlaender, Dinesh Manocha, and Mohammad Babaeizadeh. Text prompting for multi-concept video customization by autoregressive generation. *arXiv preprint arXiv:2405.13951*, 2024. 3

[27] Pengzhi Li, Qiang Nie, Ying Chen, Xi Jiang, Kai Wu, Yuhuan Lin, Yong Liu, Jinlong Peng, Chengjie Wang, and Feng Zheng. Tuning-free image customization with image and text guidance. In *European Conference on Computer Vision*, pages 233–250. Springer, 2024. 2

[28] Pengzhi Li, Qiang Nie, Ying Chen, Xi Jiang, Kai Wu, Yuhuan Lin, Yong Liu, Jinlong Peng, Chengjie Wang, and Feng Zheng. Tuning-free image customization with image and text guidance. In *European Conference on Computer Vision*, pages 233–250. Springer, 2025. 3

[29] Zhen Li, Mingdeng Cao, Xintao Wang, Zhongang Qi, Ming-Ming Cheng, and Ying Shan. Photomaker: Customizing realistic human photos via stacked id embedding. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8640–8650, 2024. 1

[30] Minghua Liu, Chao Xu, Haian Jin, Linghao Chen, Mukund Varma T, Zexiang Xu, and Hao Su. One-2-3-45: Any single image to 3d mesh in 45 seconds without per-shape optimization. *Advances in Neural Information Processing Systems*, 36, 2024. 3

[31] G Lowe. Sift-the scale invariant feature transform. *Int. J*, 2 (91-110):2, 2004. 5

[32] Liqian Ma, Xu Jia, Qianru Sun, Bernt Schiele, Tinne Tuytelaars, and Luc Van Gool. Pose guided person image generation. *Advances in neural information processing systems*, 30, 2017. 3

[33] Yue Ma, Yingqing He, Xiaodong Cun, Xintao Wang, Siran Chen, Xiu Li, and Qifeng Chen. Follow your pose: Pose-guided text-to-video generation using pose-free videos. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 4117–4125, 2024. 2

[34] Joanna Materzynska, Josef Sivic, Eli Shechtman, Antonio Torralba, Richard Zhang, and Bryan Russell. Customizing motion in text-to-video diffusion models. *arXiv preprint arXiv:2312.04966*, 2023. 3

[35] Van Nguyen Nguyen, Thibault Groueix, Georgy Ponimatkin, Yinlin Hu, Renaud Marlet, Mathieu Salzmann, and Vincent Lepetit. Nope: Novel object pose estimation from a single image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17923–17932, 2024. 3

[36] 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. 6

[37] 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. 3

[38] Yixuan Ren, Yang Zhou, Jimei Yang, Jing Shi, Difan Liu, Feng Liu, Mingi Kwon, and Abhinav Shrivastava. Customize-a-video: One-shot motion customization of text-to-video diffusion models. *arXiv preprint arXiv:2402.14780*, 2024. 3

[39] 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. 3

[40] 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. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 22500–22510, 2023. 6, 1, 3

[41] 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. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6527–6536, 2024. 3

[42] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 5

[43] Jing Shi, Wei Xiong, Zhe Lin, and Hyun Joon Jung. Instantbooth: Personalized text-to-image generation without test-time finetuning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8543–8552, 2024. 1

[44] Ruoxi Shi, Hansheng Chen, Zhuoyang Zhang, Minghua Liu, Chao Xu, Xinyue Wei, Linghao Chen, Chong Zeng, and Hao Su. Zero123++: a single image to consistent multi-view diffusion base model. *arXiv preprint arXiv:2310.15110*, 2023. 3

[45] Richard Szeliski. *Computer vision: algorithms and applications*. Springer Nature, 2022. 5

[46] Yoad Tewel, Rinon Gal, Gal Chechik, and Yuval Atzmon. Key-locked rank one editing for text-to-image personalization. In *ACM SIGGRAPH 2023 Conference Proceedings*, pages 1–11, 2023. 3

[47] Ketan Totlani. The evolution of generative ai: Implications for the media and film industry. *International Journal for Research in Applied Science and Engineering Technology*, 11(10):973–980, 2023. 2

[48] Veo-Team, :, Agrim Gupta, Ali Razavi, Andeep Toor, Ankush Gupta, Dumitru Erhan, Eleni Shaw, Eric Lau, Frank Belletti, Gabe Barth-Maron, Gregory Shaw, Hakan Erdogan, Hakim Sidahmed, Henna Nandwani, Hernan Moraldo, Hyunjik Kim, Irina Blok, Jeff Donahue, José Lezama, Kory Mathewson, Kurtis David, Matthieu Kim Lorrain, Marc van Zee, Medhini Narasimhan, Miaosen Wang, Mohammad Babaeizadeh, Nelly Papalampidi, Nick Pezzotti, Nilpa Jha, Parker Barnes, Pieter-Jan Kindermans, Rachel Hornung, Ruben Villegas, Ryan Poplin, Salah Zaiem, Sander Dieleman, Sayna Ebrahimi, Scott Wisdom, Serena Zhang, Shlomi Fruchter, Signe Nørly, Weizhe Hua, Xinchen Yan, Yuqing Du, and Yutian Chen. Veo 2. 2024. 2

[49] 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

[50] Zhao Wang, Aoxue Li, Lingting Zhu, Yong Guo, Qi Dou, and Zhenguo Li. Customvideo: Customizing text-to-video generation with multiple subjects. *arXiv preprint arXiv:2401.09962*, 2024. 3

[51] Yujie Wei, Shiwei Zhang, Zhiwu Qing, Hangjie Yuan, Zhiheng Liu, Yu Liu, Yingya Zhang, Jingren Zhou, and Hongming Shan. Dreamvideo: Composing your dream videos with customized subject and motion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6537–6549, 2024. 2

[52] Jianzong Wu, Xiangtai Li, Yanhong Zeng, Jiangning Zhang, Qianyu Zhou, Yining Li, Yunhai Tong, and Kai Chen. Mo-

tionbooth: Motion-aware customized text-to-video generation. *arXiv preprint arXiv:2406.17758*, 2024. 2

[53] 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. 3, 6

[54] Jinbo Xing, Menghan Xia, Yong Zhang, Haoxin Chen, Wangbo Yu, Hanyuan Liu, Gongye Liu, Xintao Wang, Ying Shan, and Tien-Tsin Wong. Dynamicrafter: Animating open-domain images with video diffusion priors. In *European Conference on Computer Vision*, pages 399–417. Springer, 2024. 1

[55] Hu Xu, Gargi Ghosh, Po-Yao Huang, Dmytro Okhonko, Armen Aghajanyan, Florian Metze, Luke Zettlemoyer, and Christoph Feichtenhofer. Videoclip: Contrastive pre-training for zero-shot video-text understanding. *arXiv preprint arXiv:2109.14084*, 2021. 6

[56] Yifei Xu, Xiaolong Xu, Honghao Gao, and Fu Xiao. Sgdm: An adaptive style-guided diffusion model for personalized text to image generation. *IEEE Transactions on Multimedia*, 2024. 3

[57] Ziyang Yuan, Mingdeng Cao, Xintao Wang, Zhongang Qi, Chun Yuan, and Ying Shan. Customnet: Zero-shot object customization with variable-viewpoints in text-to-image diffusion models. *arXiv preprint arXiv:2310.19784*, 2023. 3

[58] David Junhao Zhang, Roni Paiss, Shiran Zada, Nikhil Karnad, David E Jacobs, Yael Pritch, Inbar Mosseri, Mike Zheng Shou, Neal Wadhwa, and Nataniel Ruiz. Recapture: Generative video camera controls for user-provided videos using masked video fine-tuning. *arXiv preprint arXiv:2411.05003*, 2024. 3

[59] 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. 3

[60] Yuxin Zhang, Fan Tang, Nisha Huang, Haibin Huang, Chongyang Ma, Weiming Dong, and Changsheng Xu. Motioncrafter: One-shot motion customization of diffusion models. *arXiv preprint arXiv:2312.05288*, 2023. 3

[61] Yiming Zhang, Zhening Xing, Yanhong Zeng, Youqing Fang, and Kai Chen. Pia: Your personalized image animator via plug-and-play modules in text-to-image models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7747–7756, 2024. 2

[62] Rui Zhao, Yuchao Gu, Jay Zhangjie Wu, David Junhao Zhang, Jia-Wei Liu, Weijia Wu, Jussi Keppo, and Mike Zheng Shou. Motiondirector: Motion customization of text-to-video diffusion models. In *European Conference on Computer Vision*, pages 273–290. Springer, 2025. 6, 1, 3

[63] Zhan Zhuang, Xiequn Wang, Yulong Zhang, Wei Li, Yu Zhang, and Ying Wei. Copra: A progressive lora training strategy. *arXiv preprint arXiv:2410.22911*, 2024. 4

CamMimic: Zero-Shot Image To Camera Motion Personalized Video Generation Using Diffusion Models

Supplementary Material

6. Prior Work

Naive Homography. A fundamental approach to understanding scene changes due to camera motion is based on homography. This involves computing homography matrices from transitions observed in the reference video V_R and applying the same sequence to the target image I_u. Specifically, we define:

$$V_u(t) = \mathcal{H}(V_R, t) I_u(t)$$

where $\mathcal{H}(V_R, t)$ represents the homography matrix computed at time step t between consecutive frames of V_R, which is then applied to $I_u(t)$, the corresponding state of I_u. Given the central role of homography in our method, we also investigate the impact of integrating diffusion models for this task. This experiment evaluates the effect of excluding diffusion models on the generated outputs.

Tune-A-Video (Dreambooth-modified). In our method, we have used a video diffusion model. However, these are part of very recent developments within the diffusion literature. Text-to-image models have improved multi-fold since their inception. Tune-A-Video showcased extending these models to generated videos based on the text prompt provided. This method cannot be directly applied to our task as we need the motion to be applied on a specific scene. We, therefore, modify the method to incorporate a Dreambooth-style [40] training mechanism. This involves finetuning the Tune-A-Video 3D UNet with the video and its corresponding text, followed by finetuning it for the user image I_u. We have used the publicly available implementations of this work and DreamBooth to implement and test this case.

MotionDirector [62]. In contrast to Tune-A-Video, this work does the same task of transferring object motions observed in a reference video but using a video diffusion model. For scene customization, they suggest finetuning two separate UNets - one on the reference video and one on the desired scene. Following this, they replace the spatial LoRA layers in the UNet finetuned on the reference video with the spatial LoRAs of the UNet finetuned on the image. We use the publicly available implementation for this work.

7. CameraScore

In Section 3.3, we introduced *CameraScore*, a novel metric designed to evaluate the quality of camera motion generated by a method in comparison to the motion in the reference video. A *lower score* indicates better performance. We deliberately avoid assessing camera motion using the estimated 3D trajectory typically obtained via COLMAP, as it proves unreliable or infeasible under specific conditions. For example, when the camera achieves a zoom effect solely by adjusting the focal length without physical movement, or when the scene structure is predominantly flat (i.e., coplanar), COLMAP struggles to generate accurate trajectories.

Figure 8 illustrates this limitation. In Case 1, the zoom-in effect is achieved through the physical movement of the camera toward the subjects, enabling COLMAP to generate a 3D trajectory for the reference video. In contrast, Case 2 demonstrates a zoom-in effect achieved by altering the focal length, with no actual camera movement. Here, COLMAP fails to compute a trajectory. However, in both cases, homography matrices can still be computed. This highlights the advantage of using homography over COLMAP, making it a more robust and versatile metric for evaluating camera motion.

Consider the optical flow maps generated for the reference and generated videos in both Case-1 and Case-2 (Figure 8). While the observed motion in both the reference and generated videos is similar, directly comparing their optical flow maps is challenging due to structural differences between the reference scene and the desired scene. Additionally, optical flow often exhibits significant errors near boundaries, making it unreliable for accurately evaluating the observed camera motions.

8. Dataset

Our method facilitates the transfer of camera motion from a reference video to any user-provided scene. To rigorously evaluate our approach, we required a dataset that encompasses (1) videos featuring a wide variety of camera motion types and (2) videos created across different contexts, such as movies, vlogs, and animations. This diversity is essential for assessing the method's performance across various video content and textures.

Existing datasets, such as the Anatomy of Video Editing dataset [8], primarily focus on movie clips and include annotations for camera motion, shot angles, and shot types. However, these datasets do not offer a broad enough range of video content. To address this gap, we curated a custom database of publicly available videos that span multiple content categories. The dataset includes videos from a variety of sources, such as short films, social media vlogs,

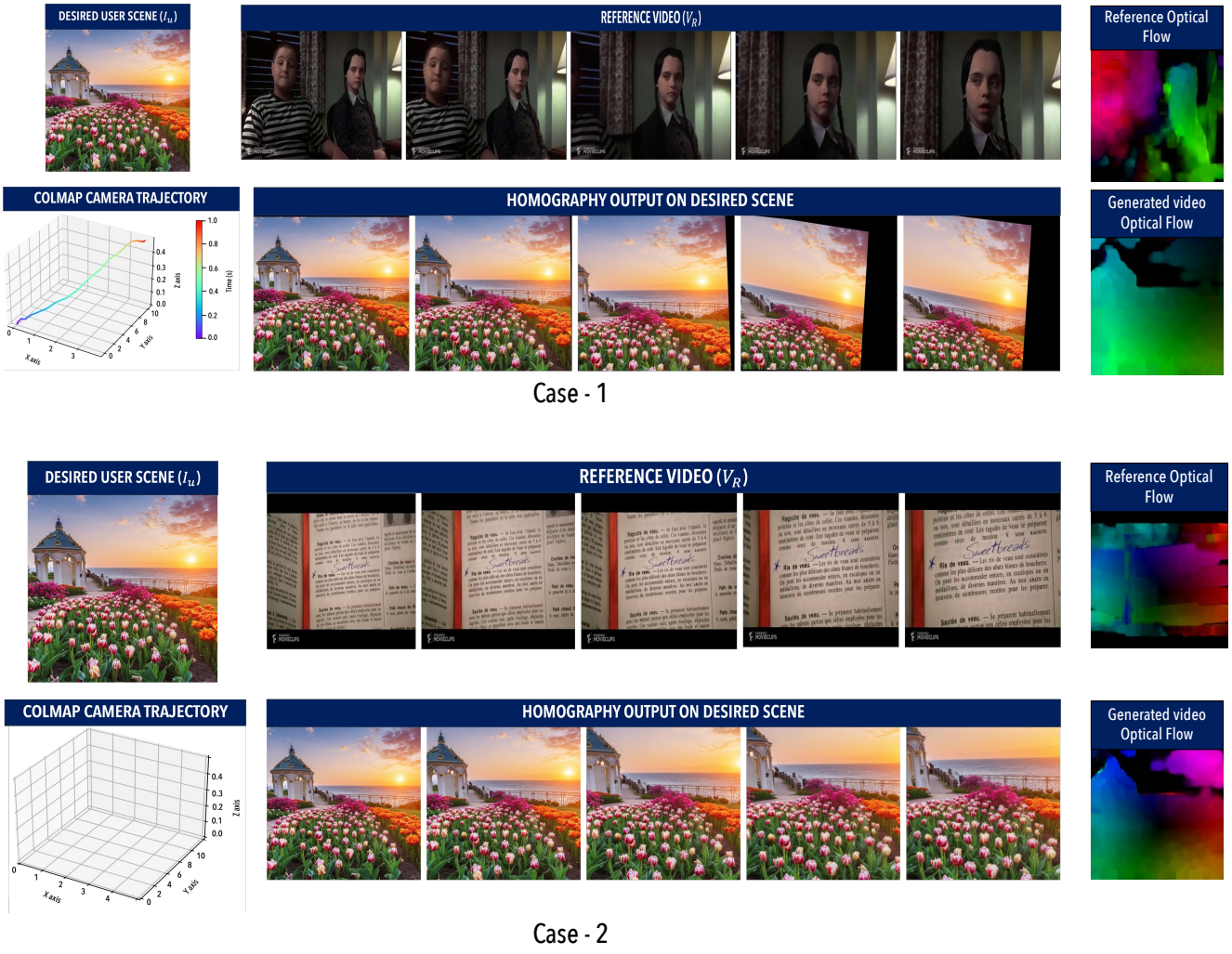

Figure 8. COLMAP and optical flow are not reliable metrics for comparing the camera motions observed in the reference and the generated videos. COLMAP fails to generate the camera trajectory in Case-2. Comparing the optical flows would be challenging because of structural differences. Therefore, we define *CamerScore*.

documentaries, and animated sequences, providing a comprehensive set of real-world and synthetic video data.

Each video in the dataset is approximately 2-4 seconds in length, providing a sufficient amount of motion information for our method to analyze and replicate. The videos are categorized based on four primary types of camera motion: zooming (in and out), panning (left to right, right to left), and tilting (up to down, down to up). These four categories represent the most commonly observed camera motions and allow for detailed testing across a range of typical shot compositions.

In addition to the four primary motion categories, the dataset includes variations in motion speed, with some clips featuring slow, subtle movements, and others with fast, dynamic shifts. The collection also incorporates diverse scene compositions and textures, ranging from minimalistic environments to more complex, cluttered scenes. This variation ensures that our method is tested on a broad spectrum of video content, enabling us to evaluate its robustness and flexibility in adapting to different types of footage.

Furthermore, we used a text-to-image model to generate a variety of user-provided scenes, ensuring that the scenes tested with the camera motion transfer method encompass a wide range of structures and complexities. This combination of diverse reference videos and custom-generated scenes allows for comprehensive testing of our approach under different conditions, ensuring its practical applicability across various real-world scenarios. Considering the different reference video-image pairs, we have a total of 408 test samples in the dataset. This is comparable to other

works in image and video synthesis such as [40].

9. Qualitative Results

We showcase some qualitative results obtained using CamMimic in Fig.9-15. In each example shown, we show the video generated using our method as well as baselines. The same video backbone (zeroscope [1]) has been used in our CamMimic and MotionDirector [62] to generate the video output observed. Stable Diffusion v2 [39] has been used as the image generation backbone for generating Tune-A-Video [53] outputs. *The supplementary video shows additional results.*

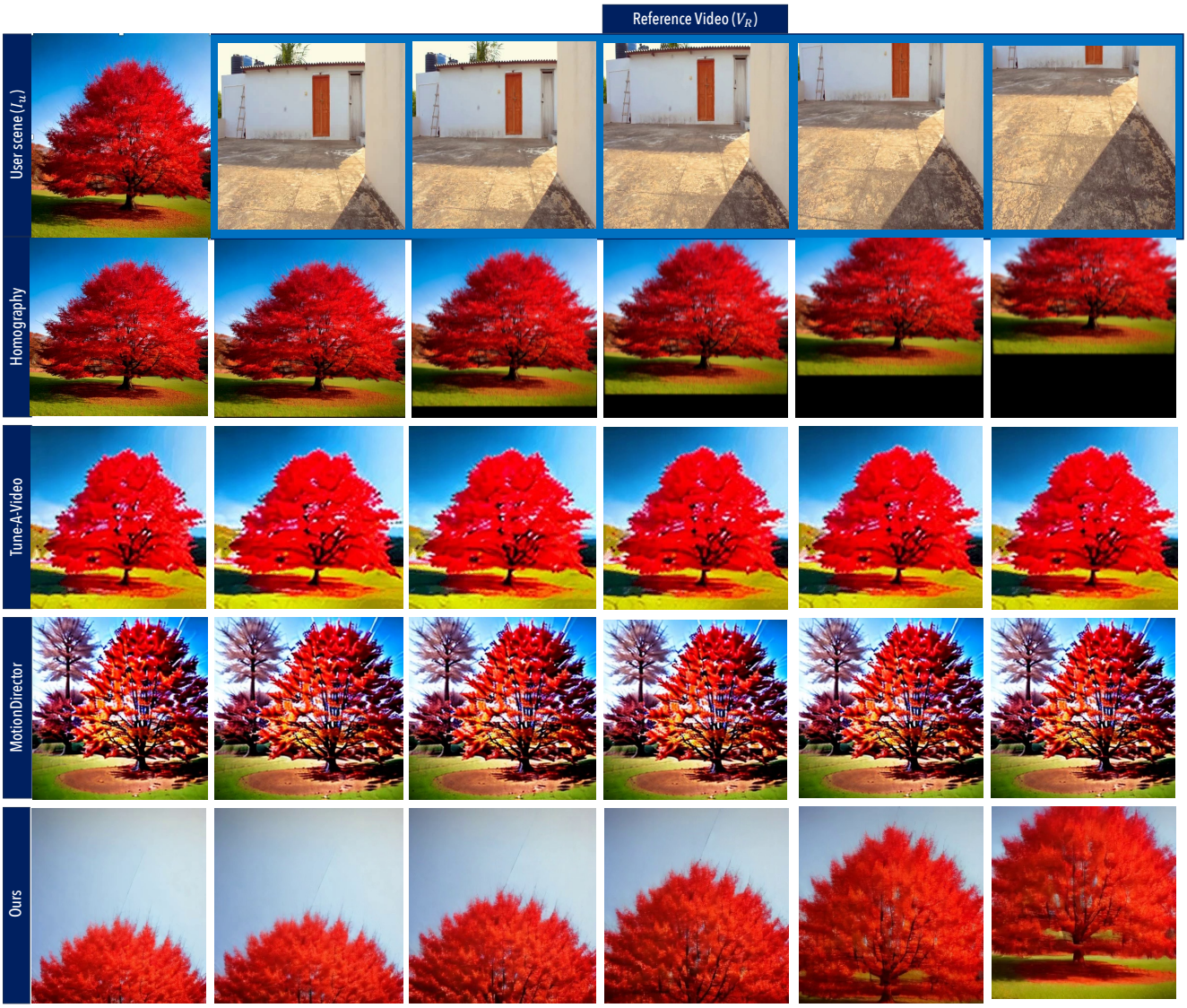

Figure 9. The first row shows the user-provided scene and the reference video containing camera motion. In contrast to the baselines (Tune-A-Video and MotionDirector), CamMimic (ours) more effectively captures the underlying camera motion from the reference video while also better preserving the details of the user-provided scene.

CameraScore

Figure 10. The first row shows the user-provided scene and the reference video containing camera motion. In contrast to the baselines (Tune-A-Video and MotionDirector), CamMimic (ours)more effectively captures the underlying camera motion from the reference video while also better preserving the details of the user-provided scene.

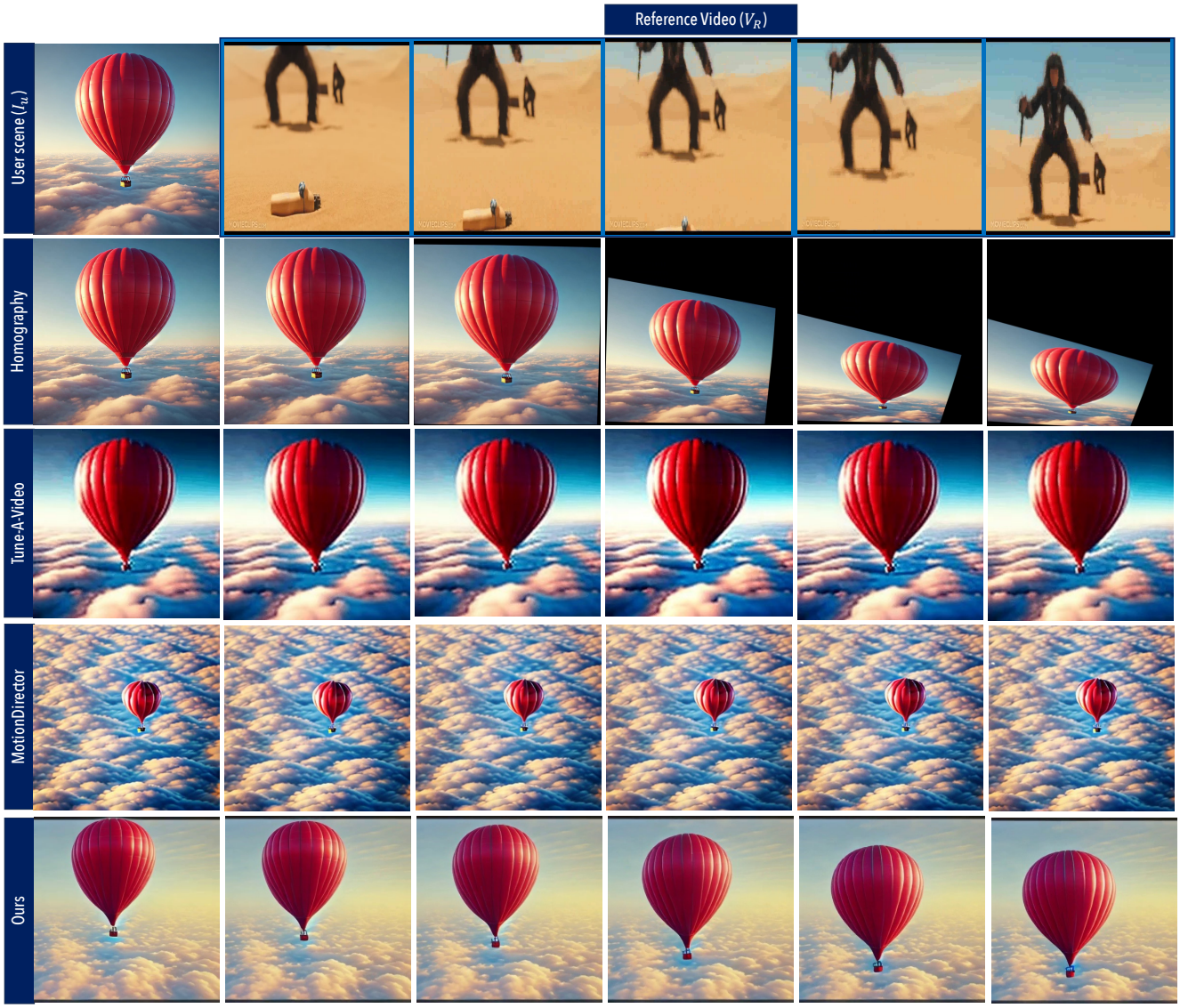

Figure 11. The first row shows the inputs, i.e., the user-provided scene and the reference video containing camera motion. In contrast to the baselines, the video output generated by CamMimic (ours) more effectively captures the underlying camera motion from the reference video while also better preserving the details of the user-provided scene.

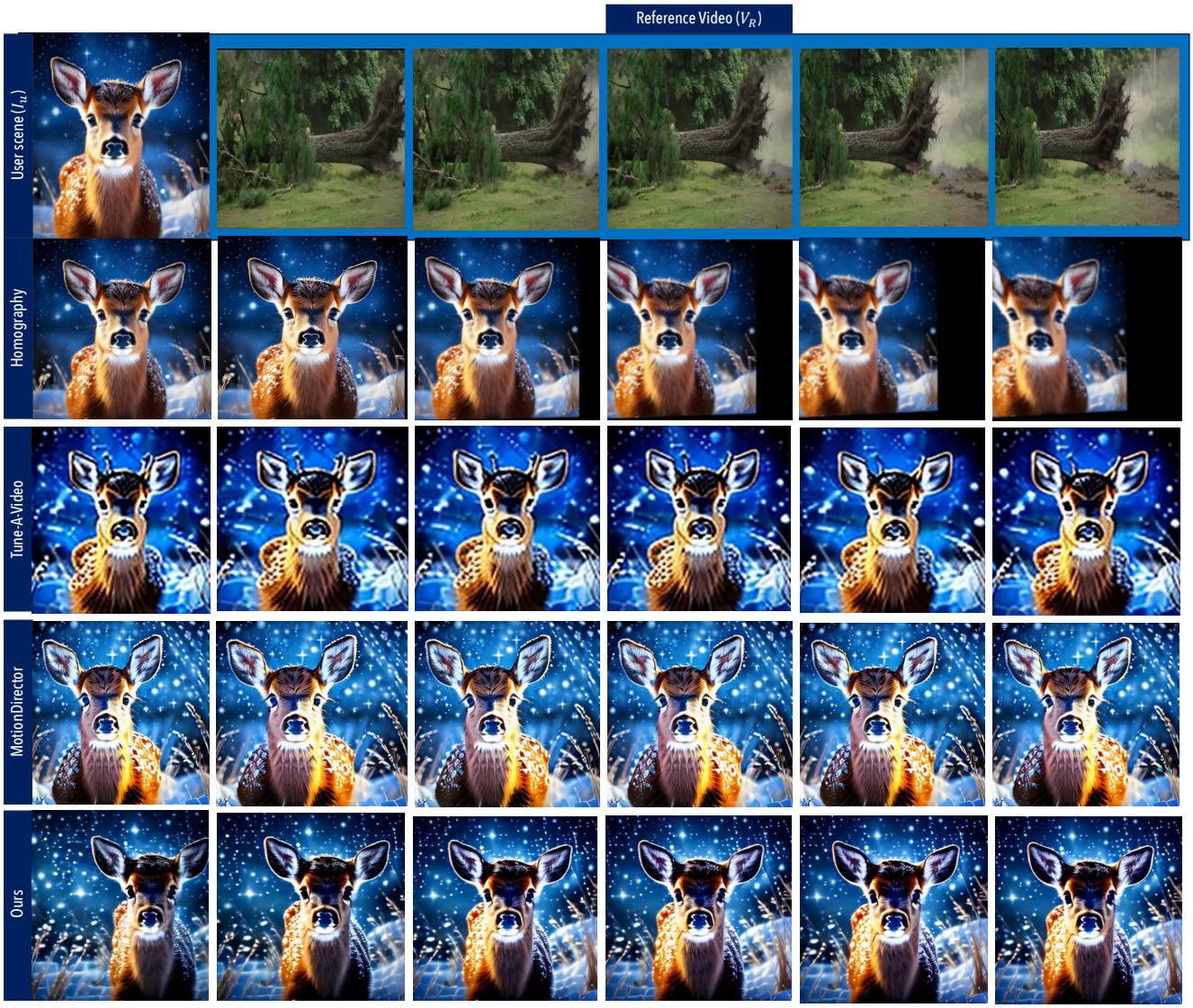

Figure 12. The first row shows the inputs, i.e. the user-provided scene and the reference video containing camera motion. In contrast to the baselines, the video output generated by CamMimic (ours) more effectively captures the underlying camera motion from the reference video while also better preserving the details of the user-provided scene.

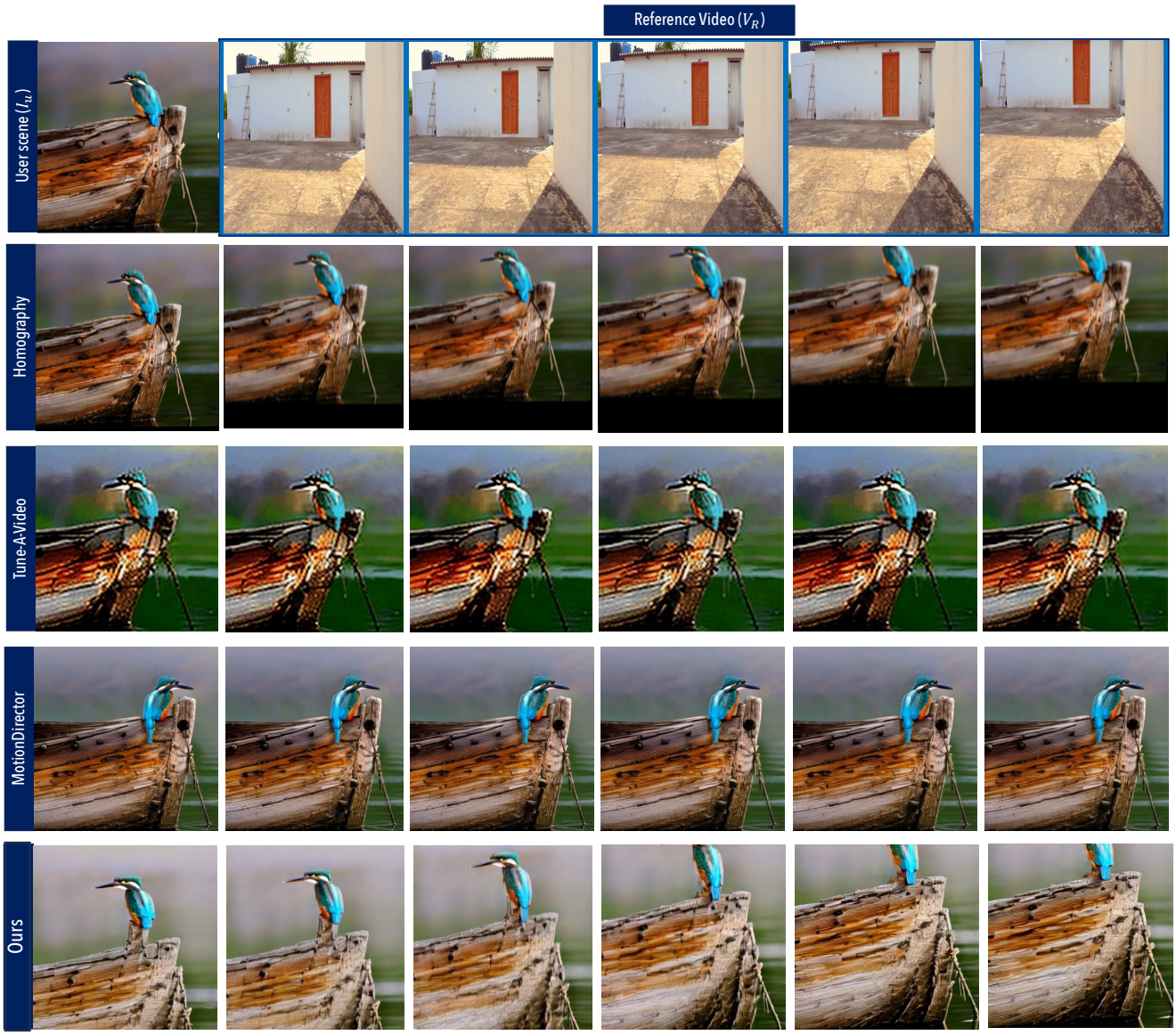

Figure 13. The first row shows the inputs, i.e., the user-provided scene and the reference video containing camera motion. In contrast to the baselines, the video output generated by CamMimic (ours) more effectively captures the underlying camera motion from the reference video while also better preserving the details of the user-provided scene.

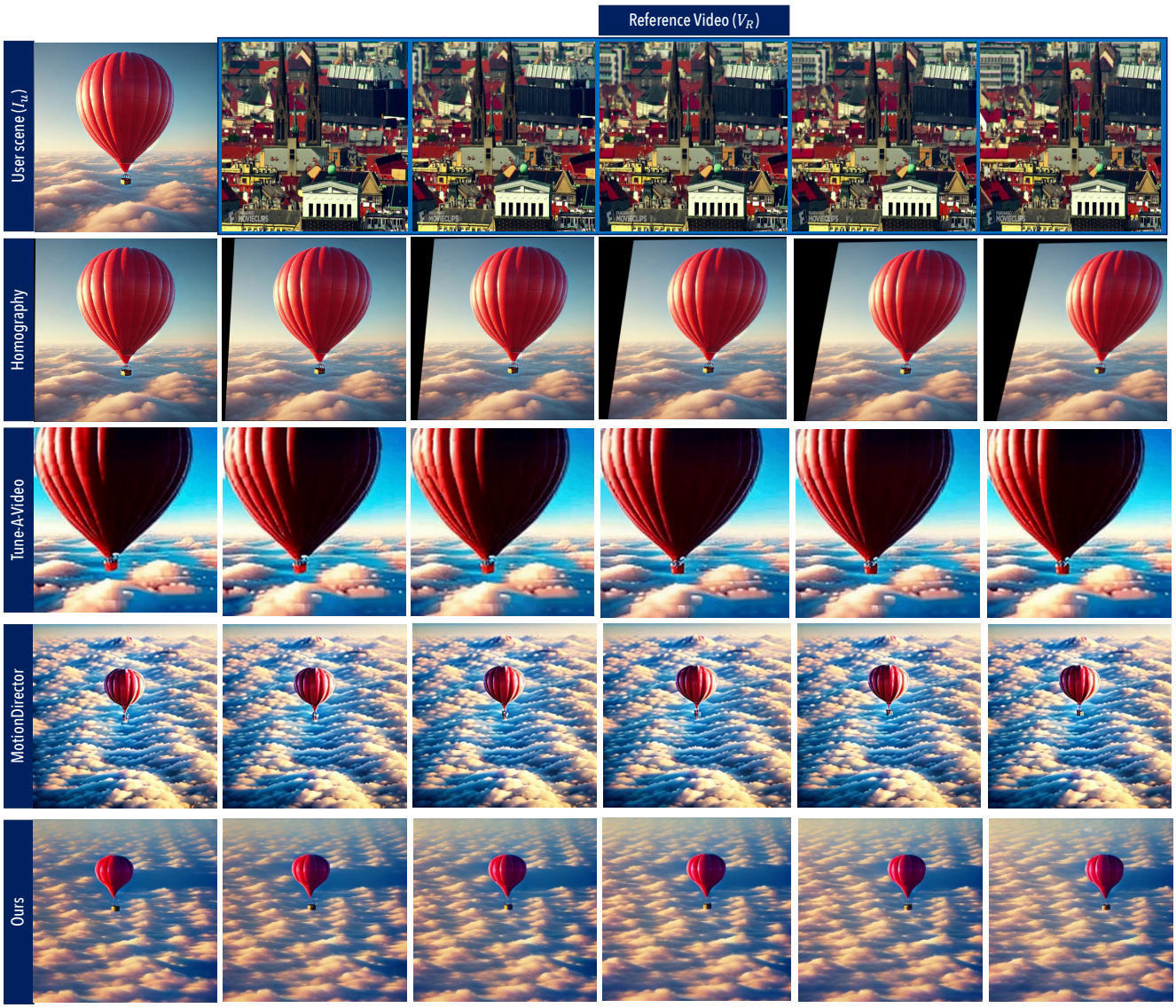

Figure 14. The first row shows the inputs, i.e., the user-provided scene and the reference video containing camera motion. In contrast to the baselines, the video output generated by CamMimic (ours) more effectively captures the underlying camera motion from the reference video while also better preserving the details of the user-provided scene.

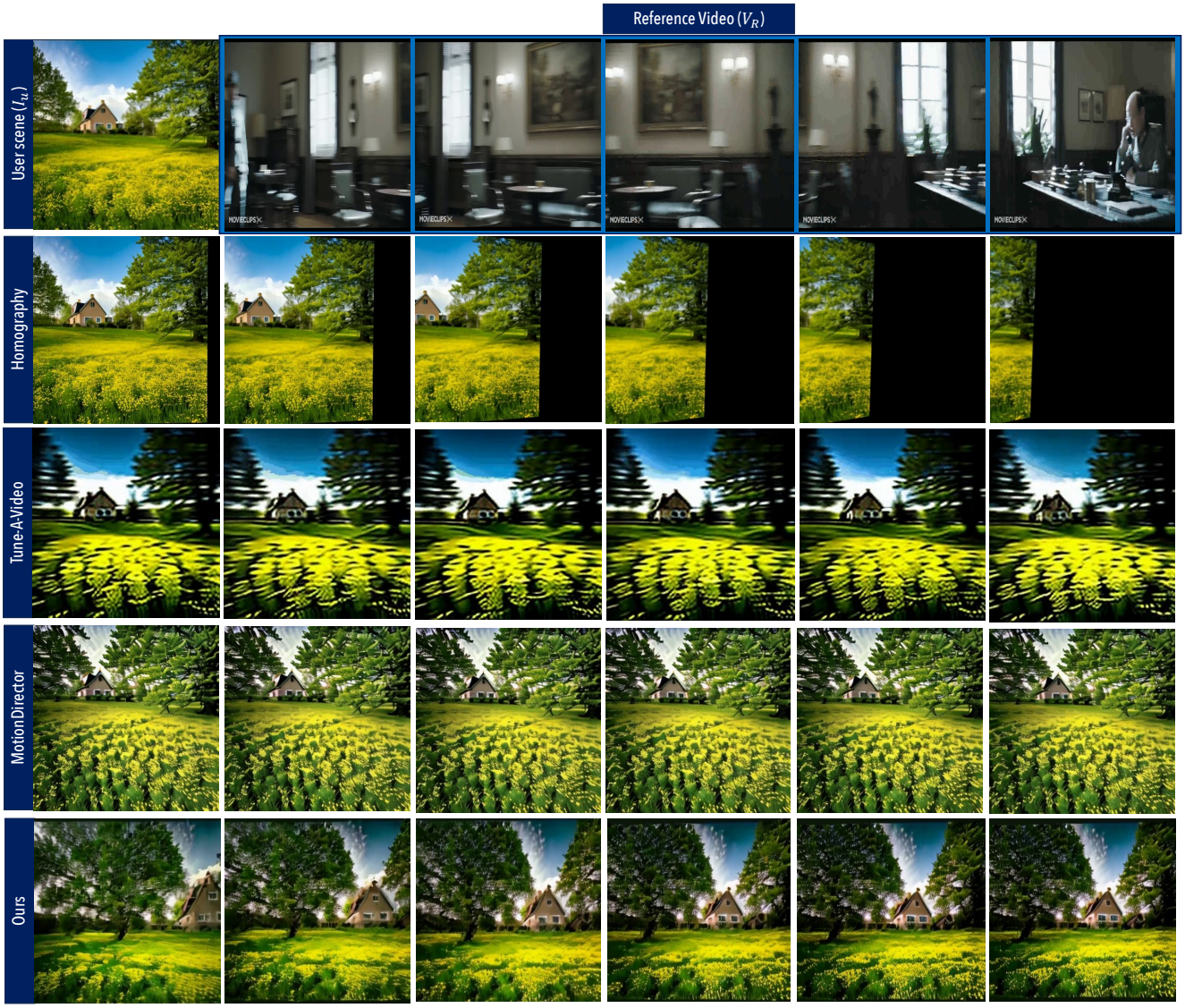

Figure 15. The first row shows the inputs, i.e., the user-provided scene and the reference video containing camera motion. In contrast to the baselines, the video output generated by CamMimic (ours) more effectively captures the underlying camera motion from the reference video while also better preserving the details of the user-provided scene.