FaceCam: Portrait Video Camera Control via Scale-Aware Conditioning

Weijie Lyu[1,2*] Ming-Hsuan Yang[1] Zhixin Shu[2†]
[1]University of California, Merced [2]Adobe Research

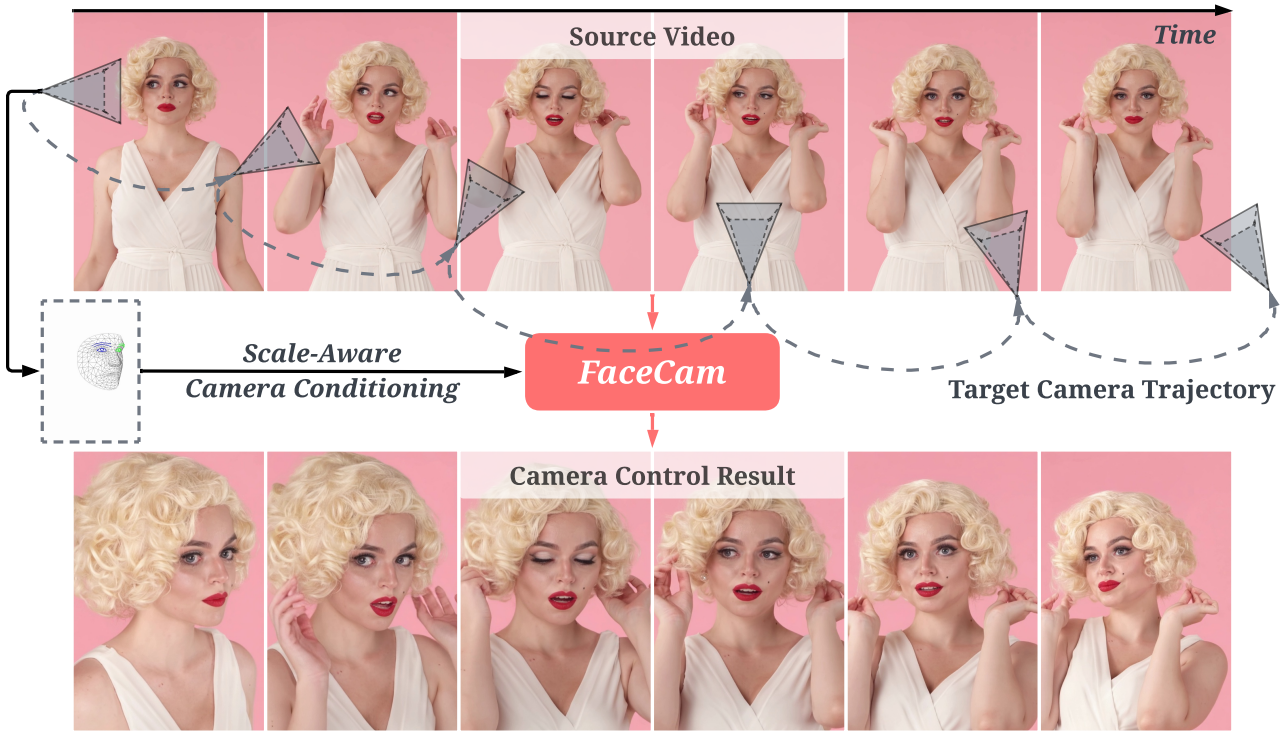


Figure 1. *FaceCam* generates portrait videos with precise camera control from a single input video and a target camera trajectory. We introduce *scale-aware camera conditioning* that represents the target camera via rendered facial landmarks, enabling accurate camera pose control. Our approach preserves subject identity and motion while maintaining high visual quality. Project page: https://weijielyu.github.io/FaceCam.

Abstract

We introduce FaceCam, a system that generates video under customizable camera trajectories for monocular human portrait video input. Recent camera control approaches based on large video-generation models have shown promising progress but often exhibit geometric distortions and visual artifacts on portrait videos due to scale-ambiguous camera representations or 3D reconstruction errors. To overcome these limitations, we propose a face-tailored scale-aware representation for camera transformations that provides deterministic conditioning without relying on 3D priors. We train a video generation model on both multi-view studio captures and in-the-wild monocular videos, and introduce two camera-control data generation strategies: synthetic camera motion and multi-shot stitching, to exploit stationary training cameras while generalizing to dynamic, continuous camera trajectories at inference time. Experiments on Ava-256 dataset and diverse in-the-wild videos demonstrate that FaceCam achieves superior performance in camera controllability, visual quality, identity and motion preservation.

1. Introduction

Controllable video generation [7, 8, 16, 39, 52, 55] has emerged as a central topic in recent research, with camera motion [1, 2, 21, 47, 58] being one of its most criti-

*Work was done when Weijie Lyu was an intern at Adobe Research.
†Corresponding author.

cal control dimensions. Meanwhile, human portrait videos are among the most prevalent video formats, making camera control for portraits a key problem in computer vision and graphics, with applications in social media, post-production, telepresence, and AR/VR. Given a source video, the goal of a camera-control system is to allow users to specify discrete camera positions or continuous camera trajectories and then generate a video of the same scene from those configurations. This defines a dynamic view-synthesis problem, where the model must infer time-varying scene geometry and synthesize unseen pixels from a learned prior. In the context of portrait videos, maintaining accurate facial expressions, verbal articulation, identity consistency, and subtle motions, such as head movement and hair dynamics, is critical for perceptual quality.

Contemporary approaches typically build on large foundation video generation models as strong visual priors. Two main strategies have emerged for specifying camera control. The first [1–3, 21, 47] employs scene-agnostic camera representations, such as intrinsic and extrinsic parameters or image-like encodings of rays (*e.g.*, Plücker rays [26]). The second [53, 55, 57, 58] infers camera motion from scene reconstruction, *e.g.* using depth estimation, thereby tying control directly to the underlying 3D structure.

For human portrait videos, existing approaches face notable challenges. Scene-agnostic camera representations, being unaware of video content, make it difficult to specify the desired camera changes for a portrait and suffer from scale ambiguity: the same parameter change can induce dramatically different visual transformations depending on the object or scene scale, shown in Fig. 2A. Reconstruction-based methods rely on 3D understanding [24, 51] to derive camera motion; small geometric errors in these estimates can amplify into large perceptual artifacts, such as shape distortions or identity drift. These artifacts are especially noticeable due to human sensitivity to facial appearance and facial expressions. The second challenge in training camera control for portrait video generation is data: acquiring paired videos with ground-truth camera annotations that capture the full complexity of human dynamics. Real portrait videos must preserve dynamic facial expressions, natural head movements, and fine-grained details such as realistic hair motion, all of which are notoriously difficult to simulate synthetically at scale. The core difficulty lies in obtaining paired training data where the same dynamic scene is recorded under different camera trajectories.

To address these challenges, we propose *FaceCam*, a portrait video generation system with precise camera control. We overcome the limitations of existing camera-conditioning schemes by introducing a scale-aware camera representation that encodes the relative transformation between source and target poses using image-space pixel correspondences. By explicitly modeling how camera motion acts on a 3D human head, this representation resolves monocular scale ambiguity and allows users to specify camera trajectories in a more direct and interpretable way. To enhance the model's ability to preserve dynamic facial expressions, natural head movements, and fine-grained details, we train our network on NeRSemble [30], a studio-captured multi-view human video dataset. However, this dataset only provides static cameras. To enable continuously moving camera trajectories at inference, we introduce two data generation strategies: synthetic camera motion and multi-shot stitching. We find that the discontinuous camera pose changes produced by multi-shot stitching during training generalize well to continuous camera trajectories at inference. We further incorporate in-the-wild videos augmented with synthetic camera motion to mitigate overfitting to the studio lighting conditions.

By leveraging a large video generation model as backbone and initialization for fine-tuning, we achieve state-of-the-art performance with high fidelity across two key dimensions: precise camera control adherence and faithful preservation of subject dynamics, including facial expressions, identity, head motion, and realistic hair movement. We validate our method on both the studio-captured Ava-256 [40] dataset, which provides ground-truth multi-view static cameras, and challenging in-the-wild portrait videos, demonstrating superior performance in camera control accuracy and video quality compared to existing methods. Our contributions are summarized as follows:

- We propose *FaceCam*, a portrait video camera-control system with a face-tailored, scale-aware camera representation that resolves the scene-scale ambiguity of traditional camera parameterizations and enables intuitive authoring of camera trajectories.
- We develop a data generation and training pipeline that, despite using only static-camera multi-view captures and unlabeled in-the-wild videos for training, supports continuous target camera motion in inference, without relying on any 4D synthetic data.
- Extensive experiments on in-the-wild data validate the effectiveness of our approach, demonstrating precise camera-control adherence and faithful preservation of subject dynamics, highlighting its promise for real-world applications.

2. Related Work

2.1. Human Face View Synthesis

View synthesis for human portraits has progressed from 3D Morphable Model (3DMM)-based [4] mesh reconstruction to NeRF-/Gaussian-based heads and, more recently, diffusion-based generation. Classical 3DMM pipelines estimate per-frame pose and expression on a textured mesh and refine appearance across frames to obtain a drivable avatar [17, 28, 44, 45], but they struggle to capture fine-

scale appearance, complex hair, and full-head coverage. Dynamic NeRF- and Gaussian-based methods further condition on expression codes or FLAME [33] parameters [14, 22, 42, 59, 61, 62], and subsequent variants improve robustness, rendering quality, and articulation for upper-body and full-head avatars. However, monocular pipelines still report inefficiency, difficulty handling large pose changes and rear-head views, and reliance on per-instance optimization over hundreds or thousands of frames, which limits scalability. Recent diffusion-based approaches [6, 10, 27, 50] and foundation-style avatar models [5, 15, 32, 34] instead condition powerful portrait or video diffusion models on audio, text, or sparse motion cues, and scale to multi-identity, multi-character settings with strong lip-sync and expression control. Nevertheless, the primary focus of these works is audio-driven portrait synthesis under limited camera motion; explicitly controlled novel view synthesis and recapturing for portrait videos, especially from a single monocular recording, remains comparatively underexplored.

2.2. Camera-Control Video Generation

Camera control for text/image-conditioned video generation [1, 2, 21, 52] extends large video diffusion models with explicit 3D camera pose or ray-based embeddings to synthesize videos that follow user-specified trajectories from prompts or single images. For dynamic novel view synthesis, GCD [47] introduces a camera-controlled video-to-video translation pipeline trained on synthetic videos from Kubric [18], but it suffers from poor generalization to in-the-wild data due to domain gaps. ReCapture [58] generates an anchor video using multi-view diffusion or point-cloud rendering and then applies masked per-video LoRA [23] fine-tuning to re-angle user-provided videos. Methods such as NVS-Solver [56] and CAT4D [54] repurpose pre-trained video or multi-view video diffusion models as zero-shot or multi-view backbones for static and dynamic novel view synthesis under target camera poses. More recently, ReCamMaster [3] trains a camera-controlled generative re-rendering model on a large synthetic multi-view video dataset rendered with Unreal Engine, and TrajectoryCrafter [57] uses a dual-stream diffusion model that fuses point-cloud renders with the source video to achieve precise trajectory control and generative inpainting of occluded regions. However, these methods still struggle on portrait videos camera control due to ambiguous camera representations and geometric estimation errors.

3. Method

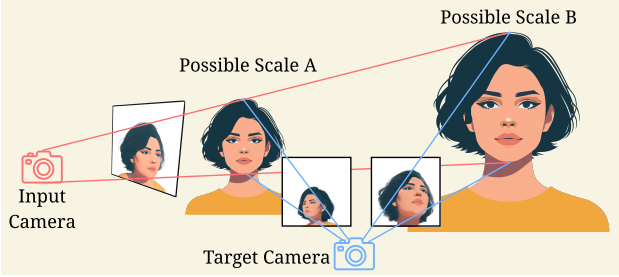
3.1. Problem Setup

Consider a dynamic head as a 4D scene A, a video V of f number of frames $\{I_i\}_{i=1}^{f} \in \mathbb{R}^{f\times h\times w\times c}$ is produced by capturing this scene along a per-frame camera trajectory $C = \{\mathbf{P}_i\}_{i=1}^{f}$, with each camera pose $\mathbf{P}_i = [\mathbf{R}_i \mid \mathbf{t}_i] \in \mathbb{R}^{3\times 4}$. Let $\mathbf{K} \in \mathbb{R}^{3\times 3}$ denote camera intrinsics, we can represent the capture process as rendering the 4D scene A:

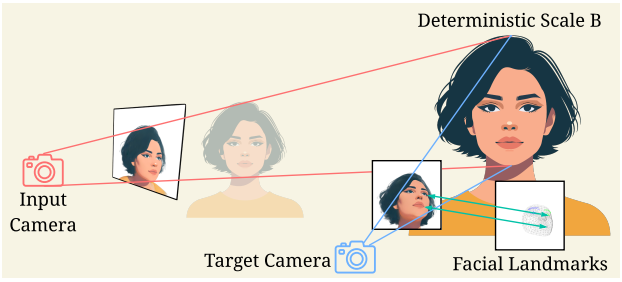
$$V = \mathsf{Render}\big(A;\, C,\, \mathbf{K}\big). \tag{1}$$

Given a source video V^s captured under a camera trajectory C^s, our goal is to generate a target video V^t under a target camera trajectory C^t which captures the same dynamic scene A. In practice, the source camera trajectory C^s is unobtainable, and our system should be able to estimate that. We represent our task as:

$$V^t = \mathit{FaceCam}(V^s, C^t). \tag{2}$$



(A) **Scale-ambiguous camera representation.** Existing camera control methods [1, 3, 21, 47] encode camera using extrinsic parameters. In monocular capture, metric depth is unobservable, the scene is determined only up to a global similarity with unknown scale and translation. Hence, the same image admits infinitely many 3D configurations, making re-rendering from a target pose underdetermined and leading to drift and poor controllability.

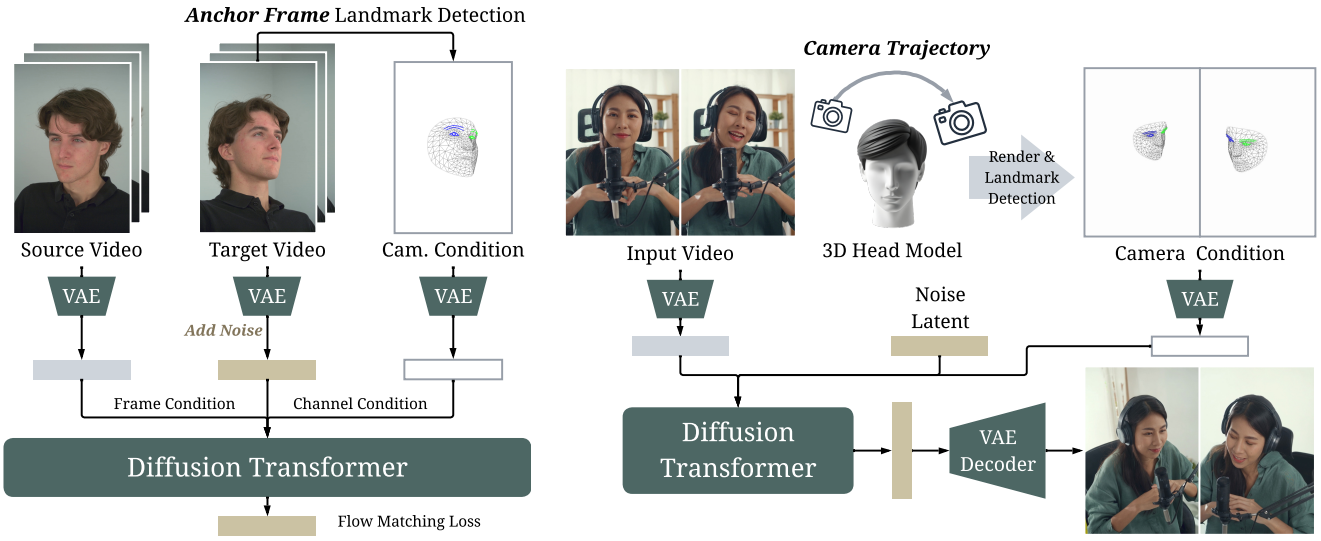


(B) **Scale-aware camera representation.** Instead of extrinsics, we encode the camera via image-space point correspondences. With at least seven 2D correspondences, the fundamental matrix between two uncalibrated views can be estimated, and with known intrinsics the relative pose is recovered up to a global scale. Portrait videos naturally provide such correspondences through facial landmarks, so we use rasterized 2D landmark maps—renderings of 3D facial landmarks from the anchor frame—as the camera representation. This face-tailored, scale-aware encoding is easy to visualize and enables deterministic, high-precision control of the apparent camera pose.

Figure 2. **Camera representation comparison.** We contrast (A) parameter-based representations, which are standard in camera control methods, with (B) image-space point correspondences, which we adopt in *FaceCam* to obtain a scale-aware conditioning that enables precise camera control.

3.2. Camera Representation via Correspondences

Image-space correspondences as sufficient camera representation. Classical multi-view geometry shows that



(A) **Training.** We extract facial landmarks from the anchor frame of the target video as camera condition. Source video, target video, and camera condition are encoded by a VAE into latents, which are passed into the diffusion transformer to predict the target latent, optimized with a flow-matching loss.

(B) **Inference.** We use a 3D head model generated as a generic head, render it along the target camera trajectory, and detect facial landmarks as the camera condition. The output latent from the diffusion transformer is decoded by a VAE decoder to obtain the camera-controlled video. We observe that, although the model is trained only with discontinuous camera pose changes, it generalizes to continuous camera trajectories during inference.

Figure 3. Training and inference pipeline of *FaceCam*.

image-space point correspondences are sufficient to characterize relative camera motion. Given two views and a set of corresponding pixels, one can estimate a fundamental matrix F that satisfies the epipolar constraint for each correspondence [19, 20]. With known intrinsics $\mathbf{K}$, F is upgraded to the essential matrix $E = \mathbf{K}^\top F \mathbf{K}$, from which the relative pose $[\mathbf{R} \mid \mathbf{t}]$ is recovered up to an unknown global scale by decomposing E [19]. Thus, point correspondences encode exactly the observable camera-induced image formation transform (up to this global scale) and are a sufficient representation of camera motion for control. This representation underpins modern SfM pipelines [43]: detect repeatable keypoints (*e.g.*, SIFT [36]), match them, robustly estimate F or E with RANSAC [13], triangulate, and refine with bundle adjustment [46]. Systems like COLMAP [43] implement this workflow end-to-end and demonstrate its effectiveness at scale.

Camera parameters and monocular scale ambiguity. Many camera control methods [1, 3, 21] directly use camera extrinsics as the conditioning signal. Representing a camera by its extrinsics $\mathbf{P} = [\mathbf{R} \mid \mathbf{t}]$ and intrinsics $\mathbf{K}$ exposes an unobservable degree of freedom. Consider the i-th frame of the source video V^s, captured under pose $\mathbf{P}_i^s = [\mathbf{R}_i^s \mid \mathbf{t}_i^s]$ towards a dynamic scene A at timestamp i:

$$V_i^s = \mathsf{Render}\big(A_i;\ [\mathbf{R}_i^s \mid \mathbf{t}_i^s],\ \mathbf{K}\big). \tag{3}$$

A 3D point $\mathbf{x} \in \mathbb{R}^3$ is expressed in camera coordinates as

$$\mathbf{x}_c = \mathbf{R}\mathbf{x} + \mathbf{t} = (x_c, y_c, z_c)^\top, \tag{4}$$

and projected to pixel coordinates

$$u = \frac{f_x x_c}{z_c} + c_x, \qquad v = \frac{f_y y_c}{z_c} + c_y, \tag{5}$$

where f_x, f_y, c_x, and c_y are from the intrinsic matrix $\mathbf{K}$. Monocular image formation is invariant to a global similarity transform: for any $\alpha > 0$, letting $\mathbf{x}' = \alpha\mathbf{x}$ and $\mathbf{t}' = \alpha\mathbf{t}$ yields

$$\mathbf{x}'_c = \mathbf{R}(\alpha\mathbf{x}) + \alpha\mathbf{t} = \alpha(\mathbf{R}\mathbf{x} + \mathbf{t}) = \alpha\mathbf{x}_c, \tag{6}$$

so the perspective ratios $x'_c/z'_c = x_c/z_c$ and $y'_c/z'_c = y_c/z_c$, and hence (u, v), remain unchanged. Given only a single monocular video V^s, absolute metric depth and translation magnitude are therefore not observable. A model conditioned directly on $[\mathbf{R} \mid \mathbf{t}]$ must implicitly choose a metric scale not fixed by the pixels; for a fixed target trajectory C^t, this can lead to variation in the 2D placement of the portrait (Fig. 2A). In contrast, point correspondences reside in pixel space and never expose this unobservable global scale, as they encode exactly what can be observed.

3.3. Scale-Aware Camera Conditioning

Camera conditioning using facial landmarks. Facial landmarks provide reliable correspondences for portrait videos. We use these landmarks to implement our correspondence-based camera representation from Sec. 3.2 (see Fig. 2B). We detect m landmarks in the first frame (anchor frame) of the target video and use them to define a head-centric coordinate system. Let $\mathbf{X} = \{\mathbf{x}_k\}_{k=1}^m$ be the 3D positions of these landmarks (from monocular 3D face

reconstruction), and let $\mathbf{U} = \{\mathbf{u}_k\}_{k=1}^{m}$ be their 2D projections under a desired camera pose $[\mathbf{R}\,|\,\mathbf{t}]$ with intrinsics $\mathbf{K}$:

$$\mathbf{u}_k = (u_k, v_k) = \mathcal{N}\big(\mathbf{K}(\mathbf{R}\mathbf{x}_k + \mathbf{t})\big), \tag{7}$$

where $\mathcal{N}$ performs perspective division: $\mathcal{N}(\mathbf{x}_c) = (x_c/z_c, y_c/z_c)$. We use $\mathbf{U}$ as the camera representation.

Scale invariance. As shown in Sec. 3.2, monocular cameras cannot determine absolute scale. Our landmark representation has this same property. If we scale both the 3D landmarks and translation by any factor $s > 0$ (*i.e.*, $\mathbf{x}'_k = s\mathbf{x}_k$ and $\mathbf{t}' = s\mathbf{t}$), the 2D projections stay the same:

$$\mathbf{u}'_k = \mathcal{N}\big(\mathbf{K}(\mathbf{R}\mathbf{x}'_k + \mathbf{t}')\big) = \mathcal{N}\big(\mathbf{K}s(\mathbf{R}\mathbf{x}_k + \mathbf{t})\big) = \mathbf{u}_k. \tag{8}$$

This shows that $\mathbf{U}$ does not depend on the absolute scale of the scene. Unlike the translation vector $\mathbf{t}$ which is defined in real-world units, $\mathbf{U}$ works directly with what we can observe in pixel space.

Sufficiency for pose control. Our landmark representation contains sufficient information to determine the camera pose. Given 3D landmarks $\mathbf{X}$ and their 2D projections $\mathbf{U}$, a PnP solver can recover the camera rotation $\mathbf{R}$ and translation $\mathbf{t}$ up to a single global scale, since monocular video does not fix absolute distances. This residual scale ambiguity matches our design: both $\mathbf{U}$ and $\mathbf{X}$ are normalized to be scale-invariant. Rather than explicitly solving for pose, we condition the generator directly on rasterized landmark maps. In practice, rather than feeding 2D landmark coordinates directly as numeric inputs, we rasterize the target landmarks into pixel-space channels and use the resulting images as the conditioning signal. This offers a key practical advantage: users can preview and author the desired camera viewpoint simply by inspecting the rendered facial shape, making camera control intuitive to specify.

3.4. Training Data Generation

A major challenge in implementing dynamic portrait video camera control is acquiring suitable training data. Scalable synthetic data acquisition for high-quality, realistic 3D dynamic portraits remains challenging; therefore, we rely exclusively on real captured data. We begin with a multi-view video dataset [30] containing facial performances from 425 subjects captured in a studio environment. Each subject is recorded from 16 synchronized viewpoints with different facial expressions and head movements, yielding approximately 9.4K video sequences. While this dataset provides known camera parameters, it has two critical limitations: first, camera trajectories remain static throughout each sequence, restricting the model to view synthesis between fixed positions. Hence, we develop several data augmentation strategies to synthesize training pairs and enhance model capabilities. Second, all captures share identical studio lighting conditions, limiting the model's ability to generalize to diverse in-the-wild scenarios. To address this limitation, we supplement our training with in-the-wild videos with synthetic camera movement.

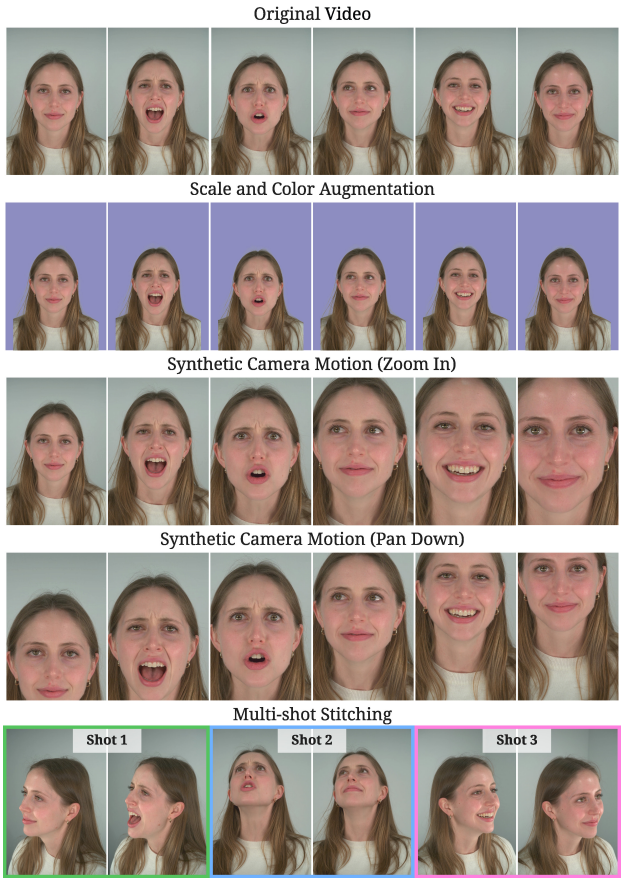


Figure 4. **Training data generation examples.** The source video is applied with scale and color augmentation to increase data diversity, while the target video is augmented with all three types to train the model's camera control capability.

Scale and color augmentation. Since NeRSemble [30] captures faces at uniform scales against fixed studio backgrounds, we introduce variation by (1) randomly scaling each clip with factor $s \in [0.75, 1.25]$, and (2) segmenting the foreground face and replacing the background with random colors (consistent between source and target videos).

Synthetic camera motion. We simulate camera motion on both studio and in-the-wild videos to enable training with dynamic cameras. Specifically, we synthesize two motion types: zoom and pan. For zoom, we sample start and end scale ratios from $[1.0, 1.25]$ and linearly interpolate per frame, producing smooth zoom-in (start $<$ end) or zoom-out (start $>$ end) effects, then restore the original resolution via cropping or padding. For pan, we linearly interpolate cropping or padding offsets across frames, assigning each frame a distinct offset to simulate lateral camera motion parallel to the image plane.

Multi-shot stitching. We can simulate the effect of a moving camera with synthetic camera movement. However, the motion is parallel to the initial image plane and does not include rotation. To introduce camera rotation in model training, we propose a multi-shot stitching technique: for each target video, we randomly select 1–4 clips captured from different camera poses, trim them to different temporal segments, and stitch them together so that a single sequence exhibits changing viewpoints. Although the generated target video only contains discrete camera pose changes, we find through experiments that the model can still perform inference with smooth, continuous camera pose changes.

Adding in-the-wild data. Training our video model exclusively on NeRSemble [30] suffices to support inference with smoothly varying camera trajectories. However, the generated lighting often deviates from the input video, and hands or accessories can appear malformed due to the dataset's limited domain. To improve generalization ability, we collect roughly 800 monocular in-the-wild portrait videos. Because these clips lack a second viewpoint, we apply synthetic camera motion to create target videos with virtual camera movement and pair them with the original clips as source videos, thereby diversifying our training set.

3.5. Inference Pipeline

For inference, we offer a user-friendly procedure to generate the camera condition given a target trajectory, as shown in Fig. 3B. We first take a 3D Gaussian head model produced by FaceLift [38] and render a proxy video along the desired camera trajectory around this head. We then run MediaPipe [37] on each rendered frame to obtain a sequence of facial landmarks, which we use as the camera conditioning signal for video generation. Note that the proxy 3D head can be of any identity and is unrelated to the input video, and we use the same 3D Gaussian head model for all experiments. Also, the sequence of facial landmarks is a representation of camera trajectory, not the actual position of the face in generated video, as further illustrated in Fig. 7.

4. Experiments

4.1. Experimental Setup

Implementation details. We build our system on the open-source video foundation model Wan [48] for conditional video generation. Following [3], we concatenate the source video latent with noise latent through frame condition, and the camera conditioning latent is applied through channel condition following [48]. Detailed architecture and training settings are provided in the supplementary material. We train *FaceCam* on the dataset introduced in Sec. 3.4. This dataset contains 8.9K videos generated from NeRSemble [30], and about 200 in-the-wild videos, in total about 9.1K videos. We follow [3] and fine-tune the 3D attention layers and projection layers of the diffusion model. The entire training takes 3K steps on 24 NVIDIA A100 GPUs, with a constant learning rate of $5e$-5 and a batch size of 24.

Table 1. **Quantitative results on Ava-256.** *FaceCam* outperforms the baselines on both reconstruction metrics and facial identity metric, indicating stronger stationary camera control ability and better preservation of identity and motion.

Method	PSNR ↑	SSIM ↑	LPIPS ↓	ArcFace ↑
ReCamMaster [3]	9.73	0.5570	0.5809	0.7014
TrajectoryCrafter [57]	10.32	0.5462	0.5673	0.5220
*FaceCam**	9.83	0.5816	0.5494	0.8073
FaceCam	**15.85**	**0.7208**	**0.2521**	**0.8574**

Evaluation datasets and metrics. To quantitatively evaluate *FaceCam*, we construct two benchmarks for camera control video generation. (1) *Static camera setting*. We select 10 identities from the studio-captured Ava-256 dataset [40]. For each identity, we construct 10 input–output camera pairs, yielding 100 videos. For the baselines, since Ava-256 provides camera extrinsics for each video, we convert these parameters into each method's camera coordinate system to form the camera-control signal. For *FaceCam*, we detect facial landmarks in the first frame of the target video and use them as the camera conditioning. To ensure a fair comparison when the target video is unavailable, we render a generic 3D Gaussian head under the target camera pose, detect its landmarks, and use them as the conditioning signal; we denote this variant as *FaceCam**. We assess novel view synthesis performance using PSNR, SSIM, and LPIPS [60], and measure identity preservation with ArcFace [11]. (2) *Dynamic camera setting.* We collect 100 in-the-wild portrait videos to evaluate *FaceCam* under dynamic camera trajectories. We apply 10 canonical camera motions (Pan Left / Right / Up / Down, Zoom In / Out, Arc Left / Right / Up / Down), following the basic trajectories in [3]. Each motion is assigned to 10 videos. We evaluate visual quality with VBench [25] and identity preservation with ArcFace [11].

Baselines. We compare *FaceCam* with two baselines, ReCamMaster [3] and TrajectoryCrafter [57], which represent two prevailing strategies for camera control in video generation: a scene-agnostic camera parameters conditioning approach and a reconstruction-based approach. ReCamMaster injects camera extrinsics as conditioning signals into the self-attention layers of DiT [41] blocks. TrajectoryCrafter first estimates a dynamic point cloud, renders it under the target camera trajectory, and then achieves camera control by inpainting the rendered dynamic point cloud.

4.2. Experiments on Ava-256

We report results on the Ava-256 dataset [40] in Tab. 1 and Fig. 5. *FaceCam* achieves stronger camera pose control and better identity and motion preservation than the baselines, demonstrating the benefit of our scale-aware camera representation and curated portrait training data. ReCamMas-

Table 2. **Quantitative results on In-the-wild videos.** *FaceCam* demonstrates superior identity preservation and camera trajectory correctness. It also generates videos with better visual quality and consistency as evidenced by VBench [25] scores.

Method	Camera Correctness	ArcFace Similarity	Imaging Quality	Aesthetic Quality	Subject Consistency	Background Consistency	Motion Smoothness	Dynamic Degree
ReCamMaster [3]	83.00	78.92	69.05	55.85	93.26	93.02	**99.30**	90.00
TrajectoryCrafter [57]	99.00	49.79	71.37	55.76	92.23	92.25	98.97	**97.00**
FaceCam (w/o In-the-wild Videos)	**100.00**	77.73	70.71	55.73	94.52	**95.16**	99.23	89.00
FaceCam	97.00	**83.94**	**73.49**	**59.91**	**94.77**	94.98	99.05	96.00

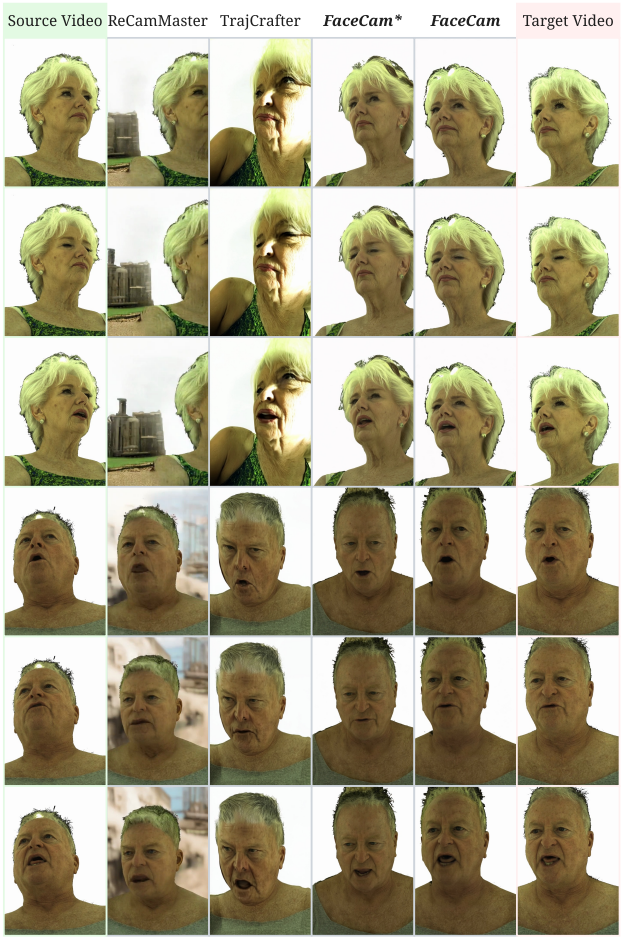


Figure 5. **Qualitative results on Ava-256.** *FaceCam* produces more realistic, ground-truth-aligned novel views than baselines. ReCamMaster [3] often fails under large pose changes, pushing the head out of frame, while TrajectoryCrafter [57] frequently shows facial distortions from dynamic point-cloud errors.

ter [3] often fails under large pose changes due to scale ambiguity, producing hallucinated backgrounds, and can only generate videos whose first frame matches the source. TrajectoryCrafter [57] exhibits facial distortions due to errors in estimating and warping the dynamic point cloud, leading to lower identity preservation score.

4.3. Experiments on In-the-wild Portrait Videos

Since ground-truth videos with known camera paths are unavailable, we evaluate camera-following accuracy via head-pose change. We use MediaPipe [37] to detect facial landmarks in the last frame of the generated and input videos, then estimate their head-pose difference. This pose change serves as a proxy for camera motion: for example, under a Pan Left target, the generated landmarks should shift right relative to the source; under an Arc Left target, the final head should face right relative to the source. We assign a binary correctness label to each video based on whether the measured pose shift matches the intended trajectory.

We present results in Tab. 2 and Fig. 6. *FaceCam* achieves high camera-motion correctness without explicit 3D geometry and attains stronger identity preservation than the baselines, highlighting the effectiveness of our scale-aware camera-conditioning design. ReCamMaster [3] shows weaker camera control and often produces blur under zoom-in motions. TrajectoryCrafter [57] relies on point-cloud estimation and lacks precise portrait-geometry reasoning, leading to lower ArcFace scores. Both baselines also struggle in outpainting (examples 2), whereas *FaceCam* exhibits more robust portrait scene understanding.

We provide an ablation study on the training data in Tab. 2. Training solely on NeRSemble [30] without in-the-wild videos yields almost perfect camera movement correctness, but leads to lower identity preservation and image quality. Our full model achieves better identity preservation and visual quality, while maintaining high camera control adherence. More qualitative comparison and ablation studies are provided in the supplementary material.

We further showcase *FaceCam* under diverse, randomly sampled camera trajectories in Fig. 7, with varying azimuths, elevations, and FOVs. The results show robust performance across a wide range of motions and scenes, with strong preservation of facial features and expressions, consistent handling of human structure (*e.g.*, hands and hair), and accurate synthesis of common co-occurring objects, indicating practical applicability in real-world settings.

5. Conclusion

We introduce *FaceCam*, a portrait video camera-control system that replaces scene-agnostic extrinsic camera representations with a face-tailored, scale-aware landmark representation. This conditioning resolves monocular scale ambiguity while providing intuitive, precise control over viewpoint. We further propose a data-generation pipeline that bootstraps from static multi-view studio captures and unlabeled in-the-wild videos via synthetic camera motion

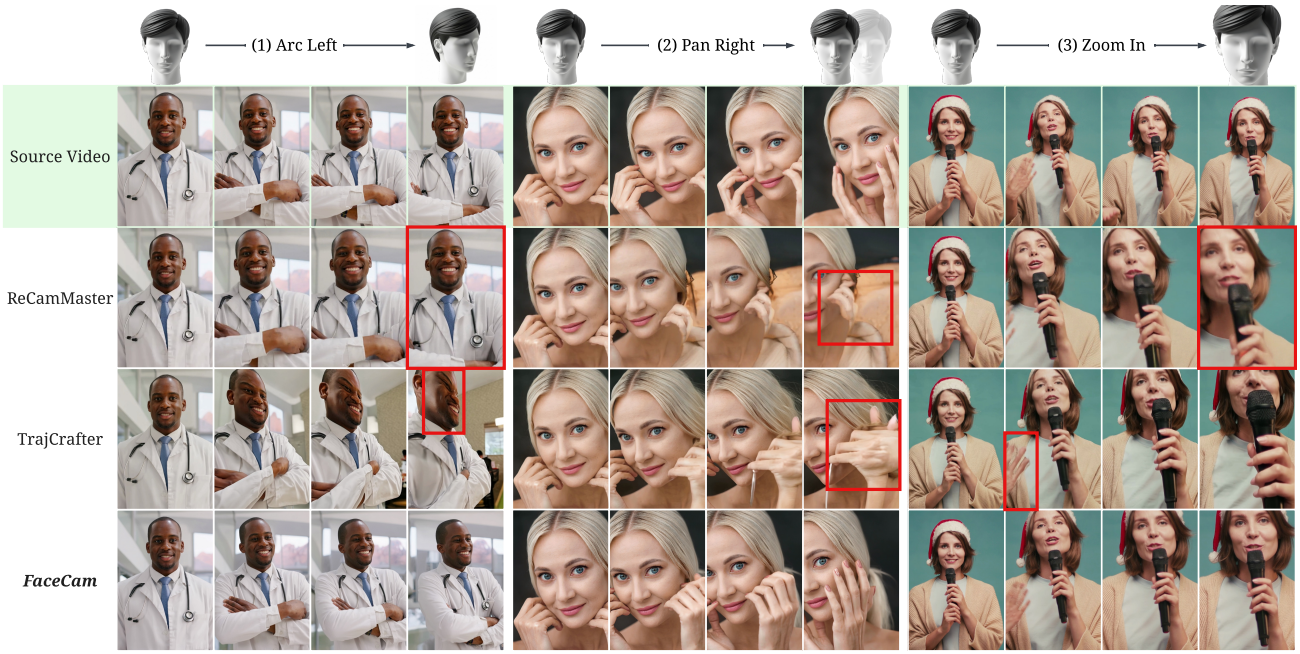


Figure 6. **Qualitative results on in-the-wild videos.** We present three camera motions: (1) Arc Left, (2) Pan Right, and (3) Zoom In. ReCamMaster [3] often loses camera control in angle changes (panel 1) and produces blurry outputs under zoom in (panel 3). TrajectoryCrafter [57] yields flattened faces with weak facial texture (panel 2). *FaceCam* delivers higher visual quality and trajectory correctness, and more faithfully captures human geometry, including hands, hair, and facial features.

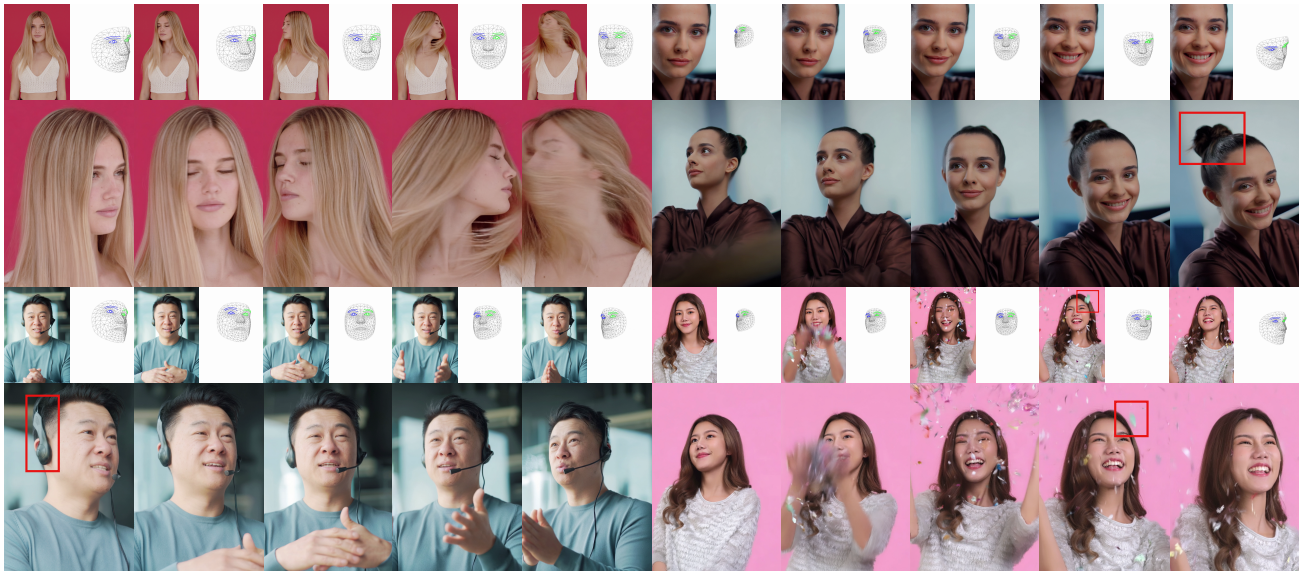


Figure 7. **In-the-wild results under diverse camera trajectories.** For each example, the first row shows the source video and the target camera, and the second row shows generated video. *FaceCam* closely follows the specified trajectories and robustly handles everyday portrait scenarios: it synthesizes realistic outpainted regions when needed (*e.g.*, bun hairstyles), preserves identity, expressions, and dynamic hair motion, and faithfully renders common co-occurring objects (*e.g.*, headset). The example also highlights strong 3D understanding (*e.g.*, flowing confetti). The first example further illustrates that the facial-landmark conditioning is not merely a representation of face location, but instead encodes camera pose and scale disentangled from head motion. ***Zoom in for higher resolution and details.***

and multi-shot stitching, enabling continuous camera trajectories at inference without explicit 3D supervision. Experiments on Ava-256 [40] and diverse in-the-wild videos demonstrate state-of-the-art camera controllability, stronger identity and motion preservation, and improved visual quality, validating both our representation and data strategy.

References

[1] Sherwin Bahmani, Ivan Skorokhodov, Guocheng Qian, Aliaksandr Siarohin, Willi Menapace, Andrea Tagliasacchi, David B Lindell, and Sergey Tulyakov. AC3D: Analyzing and improving 3D camera control in video diffusion transformers. In *CVPR*, 2025. 1, 2, 3, 4

[2] Sherwin Bahmani, Ivan Skorokhodov, Aliaksandr Siarohin, Willi Menapace, Guocheng Qian, Michael Vasilkovsky, Hsin-Ying Lee, Chaoyang Wang, Jiaxu Zou, Andrea Tagliasacchi, David B. Lindell, and Sergey Tulyakov. VD3D: Taming large video diffusion transformers for 3D camera control. In *ICLR*, 2025. 1, 3

[3] Jianhong Bai, Menghan Xia, Xiao Fu, Xintao Wang, Lianrui Mu, Jinwen Cao, Zuozhu Liu, Haoji Hu, Xiang Bai, Pengfei Wan, et al. ReCamMaster: Camera-controlled generative rendering from a single video. In *ICCV*, 2025. 2, 3, 4, 6, 7, 8, 13

[4] Volker Blanz and Thomas Vetter. A morphable model for the synthesis of 3D faces. In *SIGGRAPH*, 1999. 2

[5] Yalong Chen, Shiyu Liang, Ziyu Zhou, Ziqi Huang, Yichen Ma, Jie Tang, Qi Lin, Yao Zhou, and Qifeng Lu. HunyuanVideo-Avatar: High-fidelity audio-driven human animation for multiple characters. *arXiv preprint arXiv:2505.20156*, 2025. 3

[6] Zhao Chen, Jiale Cao, Zhe Chen, Yang Li, and Chong Ma. EchoMimic: Lifelike audio-driven portrait animations through editable landmark conditions. In *AAAI*, 2025. 3

[7] Gang Cheng, Xin Gao, Li Hu, Siqi Hu, Mingyang Huang, Chaonan Ji, Ju Li, Dechao Meng, Jinwei Qi, Penchong Qiao, et al. Wan-Animate: Unified character animation and replacement with holistic replication. *arXiv preprint arXiv:2509.14055*, 2025. 1

[8] Ruihang Chu, Yefei He, Zhekai Chen, Shiwei Zhang, Xiaogang Xu, Bin Xia, Dingdong Wang, Hongwei Yi, Xihui Liu, Hengshuang Zhao, et al. Wan-Move: Motion-controllable video generation via latent trajectory guidance. In *NeurIPS*, 2025. 1

[9] Hyung Won Chung, Noah Constant, Xavier Garcia, Adam Roberts, Yi Tay, Sharan Narang, and Orhan Firat. Unimax: Fairer and more effective language sampling for large-scale multilingual pretraining. *arXiv preprint arXiv:2304.09151*, 2023. 14

[10] Junliang Cui, Huabin Li, Yi Zhan, Haoxin Shang, Kang Cheng, Yifan Ma, Shuo Mu, Hang Zhou, Jingdong Wang, and Shi-Min Zhu. Hallo3: Highly dynamic and realistic portrait image animation with video diffusion transformer. *arXiv preprint arXiv:2412.00733*, 2024. 3

[11] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *CVPR*, 2019. 6

[12] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *ICML*, 2024. 14

[13] Martin A. Fischler and Robert C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 1981. 4

[14] Guy Gafni, Justus Thies, Michael Zollhöfer, and Matthias Nießner. Dynamic neural radiance fields for monocular 4D facial avatar reconstruction. In *CVPR*, 2021. 3

[15] Qijun Gan, Ruizi Yang, Jianke Zhu, Shaofei Xue, and Steven Hoi. OmniAvatar: Efficient audio-driven avatar video generation with adaptive body animation. *arXiv preprint arXiv:2506.18866*, 2025. 3

[16] Daniel Geng, Charles Herrmann, Junhwa Hur, Forrester Cole, Serena Zhang, Tobias Pfaff, Tatiana Lopez-Guevara, Yusuf Aytar, Michael Rubinstein, Chen Sun, et al. Motion prompting: Controlling video generation with motion trajectories. In *CVPR*, 2025. 1

[17] Philip-William Grassal, Malte Prinzler, Titus Leistner, Carsten Rother, Matthias Nießner, and Justus Thies. Neural head avatars from monocular rgb videos. In *CVPR*, 2022. 2

[18] Klaus Greff, Francois Belletti, Lucas Beyer, Carl Doersch, Yilun Du, Daniel Duckworth, David J Fleet, Dan Gnanapragasam, Florian Golemo, Charles Herrmann, et al. Kubric: A scalable dataset generator. In *CVPR*, 2022. 3

[19] Richard Hartley and Andrew Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2004. 4

[20] Richard I. Hartley. In defense of the eight-point algorithm. *IJCV*, 1997. 4

[21] Hao He, Yinghao Xu, Yuwei Guo, Gordon Wetzstein, Bo Dai, Hongsheng Li, and Ceyuan Yang. CameraCtrl: Enabling camera control for video diffusion models. In *ICLR*, 2025. 1, 2, 3, 4

[22] Yang Hong, Bo Peng, Haiyao Xiao, Ligang Liu, and Juyong Zhang. HeadNeRF: A real-time NeRF-based parametric head model. In *CVPR*, 2022. 3

[23] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. LoRA: Low-rank adaptation of large language models. In *ICLR*, 2022. 3

[24] Wenbo Hu, Xiangjun Gao, Xiaoyu Li, Sijie Zhao, Xiaodong Cun, Yong Zhang, Long Quan, and Ying Shan. DepthCrafter: Generating consistent long depth sequences for open-world videos. In *CVPR*, 2025. 2

[25] Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, et al. VBench: Comprehensive benchmark suite for video generative models. In *CVPR*, 2024. 6, 7

[26] Yan-Bin Jia. Plücker coordinates for lines in the space. Com S 477/577 Course Handout, Iowa State University, 2020. 2

[27] Jiawei Jiang, Chen Liang, Jing Yang, Guangting Lin, Tianyu Zhong, and Yujun Zheng. Loopy: Taming audio-driven portrait avatar with long-term motion dependency. *arXiv preprint arXiv:2409.02634*, 2024. 3

[28] Hyeongwoo Kim, Pablo Garrido, Ayush Tewari, Weipeng Xu, Justus Thies, Matthias Niessner, Patrick Pérez, Christian Richardt, Michael Zollhöfer, and Christian Theobalt. Deep video portraits. *ACM TOG*, 2018. 2

[29] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 14

[30] Tobias Kirschstein, Shenhan Qian, Simon Giebenhain, Tim Walter, and Matthias Nießner. NeRSemble: Multi-view radiance field reconstruction of human heads. *ACM TOG*, 2023. 2, 5, 6, 7, 12

[31] Tobias Kirschstein, Javier Romero, Artem Sevastopolsky, Matthias Nießner, and Shunsuke Saito. Avat3r: Large animatable gaussian reconstruction model for high-fidelity 3D head avatars. In *ICCV*, 2025. 12

[32] Zhihao Kong, Fei Gao, Yuxuan Zhang, Zhibo Kang, Xinyu Wei, Xin Cai, Guangyao Chen, and Wenhan Luo. Let Them Talk: Audio-driven multi-person conversational video generation. *arXiv preprint arXiv:2505.22647*, 2025. 3

[33] Tianye Li, Timo Bolkart, Michael J. Black, Hao Li, and Javier Romero. Learning a model of facial shape and expression from 4D scans. *ACM Transactions on Graphics, (Proc. SIGGRAPH Asia)*, 2017. 3

[34] Gaojie Lin, Jianwen Jiang, Jiaqi Yang, Zerong Zheng, Chao Liang, Yuan Zhang, and Jingtuo Liu. Omnihuman-1: Rethinking the scaling-up of one-stage conditioned human animation models. In *ICCV*, 2025. 3

[35] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *ICLR*, 2024. 14

[36] David G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 2004. 4

[37] Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, Chuo-Ling Chang, Ming Yong, Juhyun Lee, Wan-Teh Chang, Wei Hua, Manfred Georg, and Matthias Grundmann. MediaPipe: A framework for perceiving and processing reality. In *CVPRW*, 2019. 6, 7, 15

[38] Weijie Lyu, Yi Zhou, Ming-Hsuan Yang, and Zhixin Shu. FaceLift: Learning generalizable single image 3D face reconstruction from synthetic heads. In *ICCV*, 2025. 6, 12

[39] Yue Ma, Kunyu Feng, Zhongyuan Hu, Xinyu Wang, Yucheng Wang, Mingzhe Zheng, Xuanhua He, Chenyang Zhu, Hongyu Liu, Yingqing He, et al. Controllable video generation: A survey. *arXiv preprint arXiv:2507.16869*, 2025. 1

[40] Julieta Martinez, Emily Kim, Javier Romero, Timur Bagautdinov, Shunsuke Saito, Shoou-I Yu, Stuart Anderson, Michael Zollhöfer, Te-Li Wang, Shaojie Bai, Chenghui Li, Shih-En Wei, Rohan Joshi, Wyatt Borsos, Tomas Simon, Jason Saragih, Paul Theodosis, Alexander Greene, Anjani Josyula, Silvio Mano Maeta, Andrew I. Jewett, Simon Venshtain, Christopher Heilman, Yueh-Tung Chen, Sidi Fu, Mohamed Ezzeldin A. Elshaer, Tingfang Du, Longhua Wu, Shen-Chi Chen, Kai Kang, Michael Wu, Youssef Emad, Steven Longay, Ashley Brewer, Hitesh Shah, James Booth, Taylor Koska, Kayla Haidle, Matt Andromalos, Joanna Hsu, Thomas Dauer, Peter Selednik, Tim Godisart, Scott Ardisson, Matthew Cipperly, Ben Humberston, Lon Farr, Bob Hansen, Peihong Guo, Dave Braun, Steven Krenn, He Wen, Lucas Evans, Natalia Fadeeva, Matthew Stewart, Gabriel Schwartz, Divam Gupta, Gyeongsik Moon, Kaiwen Guo, Yuan Dong, Yichen Xu, Takaaki Shiratori, Fabian Prada, Bernardo R. Pires, Bo Peng, Julia Buffalini, Autumn Trimble, Kevyn McPhail, Melissa Schoeller, and Yaser Sheikh. Codec Avatar Studio: Paired human captures for complete, driveable, and generalizable avatars. In *NeurIPS*, 2024. 2, 6, 8, 13

[41] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *ICCV*, 2023. 6, 14

[42] Shenhan Qian, Tobias Kirschstein, Liam Schoneveld, Davide Davoli, Simon Giebenhain, and Matthias Nießner. GaussianAvatars: Photorealistic head avatars with rigged 3D gaussians. In *CVPR*, 2024. 3

[43] Johannes L. Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *CVPR*, 2016. 4

[44] Justus Thies, Michael Zollhofer, Marc Stamminger, Christian Theobalt, and Matthias Nießner. Face2face: Real-time face capture and reenactment of rgb videos. In *CVPR*, 2016. 2

[45] Justus Thies, Michael Zollhöfer, and Matthias Nießner. Deferred neural rendering: Image synthesis using neural textures. *ACM TOG*, 2019. 2

[46] Bill Triggs, Philip F. McLauchlan, Richard I. Hartley, and Andrew W. Fitzgibbon. Bundle adjustment – a modern synthesis. In *Vision Algorithms: Theory and Practice*, 2000. 4

[47] Basile Van Hoorick, Rundi Wu, Ege Ozguroglu, Kyle Sargent, Ruoshi Liu, Pavel Tokmakov, Achal Dave, Changxi Zheng, and Carl Vondrick. Generative camera dolly: Extreme monocular dynamic novel view synthesis. In *ECCV*, 2024. 1, 2, 3

[48] Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, Jiayu Wang, Jingfeng Zhang, Jingren Zhou, Jinkai Wang, Jixuan Chen, Kai Zhu, Kang Zhao, Keyu Yan, Lianghua Huang, Mengyang Feng, Ningyi Zhang, Pandeng Li, Pingyu Wu, Ruihang Chu, Ruili Feng, Shiwei Zhang, Siyang Sun, Tao Fang, Tianxing Wang, Tianyi Gui, Tingyu Weng, Tong Shen, Wei Lin, Wei Wang, Wei Wang, Wenmeng Zhou, Wente Wang, Wenting Shen, Wenyuan Yu, Xianzhong Shi, Xiaoming Huang, Xin Xu, Yan Kou, Yangyu Lv, Yifei Li, Yijing Liu, Yiming Wang, Yingya Zhang, Yitong Huang, Yong Li, You Wu, Yu Liu, Yulin Pan, Yun Zheng, Yuntao Hong, Yupeng Shi, Yutong Feng, Zeyinzi Jiang, Zhen Han, Zhi-Fan Wu, and Ziyu Liu. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025. 6, 14

[49] Wan-AI. Wan2.2: Open-source mixture-of-experts video generation models. https://github.com/Wan-Video/Wan2.2, 2025. Mixture-of-Experts (MoE) video diffusion architecture. 14

[50] Chen Wang, Keren Tian, Jicheng Zhang, Yubao Guan, Fei Luo, Fumin Shen, Zhe Jiang, Qing Gu, Xiaojuan Han, and Wenhan Yang. V-Express: Conditional dropout for progressive training of portrait video generation. *arXiv preprint arXiv:2406.02511*, 2024. 3

[51] Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. VGGT: Visual geometry grounded transformer. In *CVPR*, 2025. 2

[52] 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 *ACM SIGGRAPH 2024 Conference Papers*, 2024. 1, 3

[53] Haoyu Wu, Diankun Wu, Tianyu He, Junliang Guo, Yang Ye, Yueqi Duan, and Jiang Bian. Geometry forcing: Marrying video diffusion and 3D representation for consistent world modeling. *arXiv preprint arXiv:2507.07982*, 2025. 2

[54] Rundi Wu, Ruiqi Gao, Ben Poole, Alex Trevithick, Changxi Zheng, Jonathan T. Barron, and Aleksander Holynski. CAT4D: Create anything in 4D with multi-view video diffusion models. In *CVPR*, 2025. 3

[55] Zeqi Xiao, Wenqi Ouyang, Yifan Zhou, Shuai Yang, Lei Yang, Jianlou Si, and Xingang Pan. Trajectory attention for fine-grained video motion control. *arXiv preprint arXiv:2411.19324*, 2024. 1, 2

[56] Meng You, Zhiyu Zhu, Hui Liu, and Junhui Hou. NVS-Solver: Video diffusion model as zero-shot novel view synthesizer. In *ICLR*, 2025. 3

[57] Mark Yu, Wenbo Hu, Jinbo Xing, and Ying Shan. TrajectoryCrafter: Redirecting camera trajectory for monocular videos via diffusion models. In *ICCV*, 2025. 2, 3, 6, 7, 8, 13

[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. In *CVPR*, 2025. 1, 2, 3

[59] Jiawei Zhang, Zijian Wu, Zhiyang Liang, Yicheng Gong, Dongfang Hu, Yao Yao, Xun Cao, and Hao Zhu. FATE: Full-head gaussian avatar with textural editing from monocular video. In *CVPR*, 2025. 3

[60] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018. 6

[61] Yufeng Zheng, Victoria Fernández Abrevaya, Marcel C. Bühler, Xu Chen, Michael J. Black, and Otmar Hilliges. I M Avatar: Implicit morphable head avatars from videos. In *CVPR*, 2022. 3

[62] Wojciech Zielonka, Timo Bolkart, and Justus Thies. Instant volumetric head avatars. In *CVPR*, 2023. 3

FaceCam: Portrait Video Camera Control via Scale-Aware Conditioning

Supplementary Material

1. Overview

In this supplementary material, we first present a video (contains audio) overview of *FaceCam* and its visual results. In Sec. 2, we provide ablation studies on training data generation and proxy head selection. Additional qualitative results are shown in Sec. 3. Implementation details are given in Sec. 4. We include relevant preliminaries in Sec. 5, and discuss limitations and future work in Sec. 6.

2. Ablation Study

2.1. Training Data Generation

We provide an ablation study on training data generation in Tab. 3 and Fig. 9 to analyze the impact of each strategy on the final results. We compare three ablated variants and our full model:

- *FaceCam (w/o Synthetic Camera Motion)*: applies only Multi-shot Stitching to NeRSemble [30] videos.
- *FaceCam (w/o Multi-shot Stitching)*: applies only Synthetic Camera Motion to NeRSemble videos.
- *FaceCam (w/o In-the-wild Videos)*: applies both Synthetic Camera Motion and Multi-shot Stitching to NeRSemble videos, without using any in-the-wild videos.
- *FaceCam*: our full model, which adds in-the-wild videos with Synthetic Camera Motion (since multi-view videos are not available) on top of the third baseline.

Through these experiments, we observe that Synthetic Camera Motion enables the model to learn zoom and pan motions and to produce smooth trajectories without sudden camera pose changes. Multi-shot Stitching further teaches the model to follow camera angle changes along the target trajectory, and together these two strategies yield accurate camera control. Incorporating in-the-wild video data improves generalization to diverse real-world lighting conditions and objects, leading to better appearance consistency with the input video.

2.2. Proxy 3D Head Selection

For all experiments reported in the main paper, we use a single generic 3D Gaussian head as a proxy during inference to render videos and extract facial landmarks. This proxy head can be generated by any 3D head generation methods [31, 38]. To verify that the specific choice of proxy head does not influence performance, we select two additional proxy heads (Fig. 8) and evaluate *FaceCam* on in-the-wild videos. The results in Tab. 3 show only minor differences across all three proxies, indicating that our approach is largely insensitive to the particular proxy head. This supports our design choice that the landmarks serve purely as a camera-conditioning signal, rather than conveying identity or expression information. The identity and expression in the generated videos come solely from the source video.

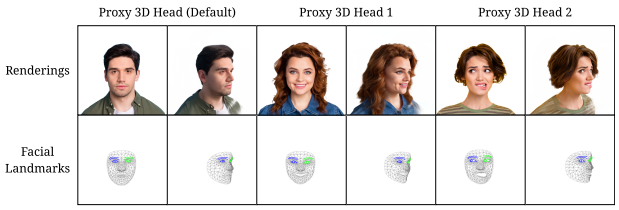


Figure 8. **Different choices of proxy 3D head**. We select two additional proxy 3D heads with different identities and corresponding facial landmark detections, and conduct an ablation study showing that the proxy's identity and expression do not affect the final generation results.

3. Experimental Results

We conduct extensive experiments on in-the-wild videos with diverse camera trajectories to assess *FaceCam* under both challenging synthetic settings and realistic use cases. The results are shown in Fig. 10 and Fig. 11. Across a wide range of inputs, the model tracks intricate head and hair dynamics, responds smoothly to varied facial expressions, and respects motion-dependent artifacts such as blur from rapid body movement. When the target trajectory places the virtual camera farther from the subject, *FaceCam* plausibly completes missing regions by synthesizing coherent clothing and background content. On real footage, it recreates studio and streaming scenes with stable identity and layout, while reliably retaining fine-grained accessories and props (*e.g.*, cosmetics, jewelry, headbands, microphones, and glasses). Notably, the same pipeline extends to stylized inputs such as cartoon characters, indicating strong generalization beyond the distribution of the training data.

4. Implementation Details

4.1. Training Data Generation

We provide pseudo-code for three training data generation procedures. *Scale and Color Augmentation* (Algorithm 1) is applied to both source and target videos to increase data diversity, including variations in head size and background appearance. *Synthetic Camera Motion* (Algorithm 2) is applied to the target video to create continuous camera trajectories with zoom and pan effects, which is essential for achieving smooth, temporally coherent generated videos.

Table 3. **Ablation study.** We conduct ablation studies to quantify the impact of different training data components on the final performance of our model. We also vary the choice of proxy head and show that this selection has negligible effect on the generated results.

Method	Camera Correctness	ArcFace Similarity	Imaging Quality	Aesthetic Quality	Subject Consistency	Background Consistency	Motion Smoothness	Dynamic Degree
FaceCam (w/o Synthetic Camera Motion)	96.00	81.19	72.03	58.02	94.27	94.91	99.29	83.00
FaceCam (w/o Multi-shot Stitching)	86.00	76.38	70.73	55.10	94.56	95.12	**99.30**	80.00
FaceCam (w/o In-the-wild Videos)	**100.00**	77.73	70.71	55.73	94.52	**95.16**	99.23	89.00
FaceCam	97.00	**83.94**	**73.49**	**59.91**	**94.77**	94.98	99.05	**96.00**
FaceCam (Proxy 3D Head 1)	97.00	84.45	73.48	59.85	94.80	94.89	99.02	95.00
FaceCam (Proxy 3D Head 2)	97.00	84.74	73.47	59.89	94.74	94.89	99.03	94.00

Algorithm 1 Scale and Color Augmentation

Require: Source clip $V^s = \{I_i^s\}_{i=1}^{T_s}$, target clip $V^t = \{I_i^t\}_{i=1}^{T_t}$
Ensure: Augmented clips $\tilde{V}^s, \tilde{V}^t$

Sample scale factors $s^s, s^t \sim \mathcal{U}(0.75, 1.25)$
Sample a background color $c \sim$ UniformColor() ▷ shared between source and target
for $i = 1$ to T_s **do** ▷ augment source clip
  $J_i^s \leftarrow \text{Resize}(I_i^s, s^s)$
  $M_i^s \leftarrow \text{FaceSeg}(J_i^s)$ ▷ $M_i^s \in \{0,1\}^{H \times W}$
  $B_i^s \leftarrow c \cdot (\mathbf{1} - M_i^s)$
  $\tilde{I}_i^s \leftarrow M_i^s \odot J_i^s + B_i^s$
end for
$\tilde{V}^s \leftarrow \{\tilde{I}_i^s\}_{i=1}^{T_s}$
for $i = 1$ to T_t **do** ▷ augment target clip with same background color
  $J_i^t \leftarrow \text{Resize}(I_i^t, s^t)$
  $M_i^t \leftarrow \text{FaceSeg}(J_i^t)$
  $B_i^t \leftarrow c \cdot (\mathbf{1} - M_i^t)$
  $\tilde{I}_i^t \leftarrow M_i^t \odot J_i^t + B_i^t$
end for
$\tilde{V}^t \leftarrow \{\tilde{I}_i^t\}_{i=1}^{T_t}$

Algorithm 2 Synthetic Camera Motion (Zoom and Pan)

Require: Input clip $V = \{I_i\}_{i=1}^{T}$, motion type $m \in \{\text{zoom}, \text{pan}\}$, image resolution (H, W)
Ensure: Motion-augmented clip $\tilde{V}$

if $m =$ zoom **then**
  Sample $s_{\text{start}}, s_{\text{end}} \sim \mathcal{U}(1.0, 1.25)$
  for $i = 1$ to T **do**
    $\alpha \leftarrow \frac{i-1}{\max(T-1,1)}$
    $s_i \leftarrow (1-\alpha) \cdot s_{\text{start}} + \alpha \cdot s_{\text{end}}$
    $J_i \leftarrow \text{Resize}(I_i, s_i)$
    $\tilde{I}_i \leftarrow \text{CenterCropOrPad}(J_i, H, W)$
  end for
else if $m =$ pan **then**
  Choose maximum offset δ_x, δ_y relative to (H, W)
  Sample offsets $\mathbf{o}_{\text{start}}, \mathbf{o}_{\text{end}} \sim [-\delta_x, \delta_x] \times [-\delta_y, \delta_y]$
  for $i = 1$ to T **do**
    $\alpha \leftarrow \frac{i-1}{\max(T-1,1)}$
    $\mathbf{o}_i \leftarrow (1-\alpha)\,\mathbf{o}_{\text{start}} + \alpha\,\mathbf{o}_{\text{end}}$
    $\tilde{I}_i \leftarrow \text{CropOrPadWithOffset}(I_i, \mathbf{o}_i, H, W)$
  end for
end if
$\tilde{V} \leftarrow \{\tilde{I}_i\}_{i=1}^{T}$

Finally, *Multi-shot Stitching* (Algorithm 3) is applied to the target video to introduce discrete camera pose changes, enabling the model to handle viewpoint transitions in the generated outputs.

4.2. Experimental Details

All experiments are conducted at a resolution of 704×480, with generated videos of length 81 frames. We use the same text prompt for all experiments: "A portrait of a person." TrajectoryCrafter [57] can generate at most 49 frames in the general setting, and only 29 frames when the first frame of the generated video does not coincide with the first frame of the source video. To ensure a fair comparison, we therefore evaluate on the first 29 frames in the static-camera setting and on the first 49 frames in the dynamic-camera setting for all baselines. ReCamMaster [3] produces camera-controlled results only when the target camera pose for the first frame has an identity rotation; otherwise, the generated video degenerates to the source video. In the static-

Table 4. Source and target camera pairs used in experiment on Ava-256 [40].

ID	Source Camera	Target Camera
1	cam_400944	cam_401031
2	cam_400944	cam_401410
3	cam_400981	cam_401045
4	cam_400981	cam_401292
5	cam_401163	cam_401031
6	cam_401163	cam_401458
7	cam_401168	cam_401045
8	cam_401168	cam_401292
9	cam_401316	cam_401410
10	cam_401316	cam_401458

Algorithm 3 Multi-shot Stitching

Require: Set of clips for a target video $\mathcal{C} = \{V^{(k)}\}_{k=1}^{N}$, $V^{(k)} = \{I_i^{(k)}\}_{i=1}^{T_k}$, maximum shots $K_{\max} = 4$
Ensure: Stitched target clip $\tilde{V}$

```
Sample number of shots K ~ Uniform{1, 2, ..., K_max}
Sample K distinct indices {i_1, ..., i_K} from {1, ..., N}   ▷ different camera poses
Initialize stitched sequence Ṽ ← ∅
for j = 1 to K do
    V^(j) ← V^(i_j), length T^(j)
    Sample start index a_j ~ {1, ..., T^(j) − 1}
    Sample end index b_j ~ {a_j + 1, ..., T^(j)}
    Define segment S_j ← {I_i^(j) | i = a_j, ..., b_j}
    Ṽ ← Concat(Ṽ, S_j)
end for
return Ṽ
```

camera experiments, we thus enforce an identity rotation as the first-frame camera condition for this baseline to obtain valid results. All baselines are run with their official configurations and released pre-trained weights.

For the static-camera setting on the Ava-256 dataset, we select 10 identities, each with 10 source–target camera pairs, yielding a total of 100 videos. The selected identities are `KDA058`, `XJT672`, `LAS440`, `IFG774`, `EID363`, `NRE683`, `PAK800`, `MCR809`, `SKB942`, `KJJ701`. The source and target cameras are summarized in Tab. 4.

5. Preliminary

5.1. Conditional Video Generation

We build our system on the open-source video foundation model Wan [48] for conditional video generation. Wan is a latent video diffusion model comprising a 3D Variational Autoencoder (VAE) [29], a text prompt encoder [9], and two transformer-based diffusion models (DiT) [41] specialized for the high and low noise stages. The model adopts Rectified Flow framework [12] for the noise schedule and denoising process. Detailed architecture and training settings are provided in the supplementary material.

During training, a pre-trained 3D VAE encodes a video $V \in \mathbb{R}^{f\times h\times w\times c}$ into latent space: $z_0 = \mathcal{E}(V)$. Then in the forward diffusion process, the DiT injects Gaussian noise ϵ into z_0 to create a noisy latent. The forward process is defined as straight paths between data distribution and a standard normal distribution.

$$z_t = (1-t)z_0 + t\epsilon, \qquad \varepsilon \sim \mathcal{N}(0, I), \tag{9}$$

where t denotes the iterative timestep. z_t is then patchified, concatenated with text tokens encoded by the text prompt encoder and other additional conditioning signals, and fed into DiT blocks. To solve the reverse denoising process, Conditional Flow Matching (CFM) [35] learns a time-dependent velocity field $v_\theta(z, t, \mathbf{c})$ that defines an ordinary differential equation (ODE):

$$\frac{dz_t}{dt} = v_\theta(z_t, t, \mathbf{c}), \qquad t \in [0, 1], \tag{10}$$

transporting samples from the base (standard Gaussian) to the data distribution under conditioning $\mathbf{c}$. With the rectified interpolant, the target velocity along the path is constant:

$$u^\star(z_t, t|z_0, \varepsilon) = \frac{dz_t}{dt} = \varepsilon - z_0. \tag{11}$$

CFM trains v_θ by regressing to this target with MSE loss:

$$\mathcal{L}_{\text{CFM}} = \mathbb{E}_{t,z_0,\epsilon}\big|\big|v_\theta(z_t, t, \mathbf{c}) - (\varepsilon - z_0)\big|\big|_2^2. \tag{12}$$

At inference, we integrate the learned ODE deterministically from noise to data by marching from $t = 1$ to $t = 0$:

$$z_{t-\Delta t} = z_t - \Delta t\, v_\theta(z_t, t, \mathbf{c}), \tag{13}$$

yielding the final latent z_0 consistent with the conditioning $\mathbf{c}$. z_0 is then decoded by the pre-trained VAE decoder and outputs the generated video: $V = \mathcal{D}(z_0)$.

5.2. Wan2.2 and MoE Video Diffusion

Wan2.2 [48] is a family of large-scale latent video diffusion models. It supports multi-modal conditioning (text-to-video, image-to-video, text–image-to-video, and specialized speech/animation variants) and generates high-fidelity videos up to 720p at 24 fps using a high-compression video VAE [29] and a DiT-style [41] diffusion backbone.

To scale model capacity without increasing inference cost, Wan2.2 replaces a single denoising network with a Mixture-of-Experts (MoE) [49] architecture. A set of expert denoisers is specialized for different noise regimes (*e.g.*, high-noise early steps *vs*. low-noise late steps), and a routing scheme based on the diffusion timestep selects which expert to apply at each step. This design enlarges the total parameter count and improves motion, semantic, and aesthetic fidelity, while keeping per-step FLOPs comparable to a dense model.

More generally, an MoE layer consists of a collection of experts $\{E_k\}_{k=1}^{K}$ and a gating function $g(x)$ that selects a sparse subset of experts for each input x, often via top-k routing. Only the selected experts are evaluated and their outputs are combined, for example

$$y = \sum_{k\in\mathcal{S}(x)} g_k(x)\, E_k(x), \tag{14}$$

where $\mathcal{S}(x)$ is a small set of active experts and $g_k(x)$ are normalized routing weights. By activating only a few experts per input, MoE architectures enable models with billions of parameters to operate at roughly the same compute cost as much smaller dense networks, a property that Wan2.2 leverages to scale video generation quality and controllability.

5.3. MediaPipe Facial Landmark Detection

We use Google's MediaPipe Face Mesh [37] as an off-the-shelf module to obtain dense 2D/3D facial keypoints from monocular RGB inputs. MediaPipe Face Mesh predicts a set of $K = 468$ landmarks in real time, even on mobile devices, by applying a lightweight neural network to a cropped face region and regressing per-vertex coordinates that approximate the full facial surface. The model operates on a single RGB camera without requiring depth sensors and is optimized for GPU acceleration, making it suitable for large-scale video processing and interactive applications.

Concretely, given an input frame I_i, the detector returns a landmark set

$$\mathbf{U}_i = \{\mathbf{u}_{i,k}\}_{k=1}^{K}, \quad K = 468, \tag{15}$$

where each

$$\mathbf{u}_{i,k} = (x_{i,k}, y_{i,k}, z_{i,k}) \tag{16}$$

encodes normalized image coordinates $(x_{i,k}, y_{i,k})$ and a relative depth value $z_{i,k}$. In practice, MediaPipe adopts a two-stage pipeline: a BlazeFace-style face detector first produces a tight region of interest, and a dedicated mesh regressor then predicts the dense landmark configuration within that region. These landmarks are widely used in AR and avatar applications to recover facial geometry and pose from video streams; in our work, we reuse them as a compact, robust representation for conditioning and camera control.

6. Limitations and Future Work

Despite the accurate camera control and high-quality results achieved, *FaceCam* still has several limitations. First, because facial landmarks can only be detected when facial features are visible in the input video, *FaceCam* cannot handle views where the camera rotates to the back of the head. For the same reason, although *FaceCam* can generalize to cartoon characters, it does not extend to general scenes in which a facial landmark detector is inapplicable. Building on the same idea of using image-space correspondences as a camera representation, but redefining how these correspondences are encoded, could help address this limitation. Second, background generation is not the focus of this work, partly due to data limitations. Incorporating synthetic data with multi-view-consistent backgrounds could further improve the model's ability to synthesize background content behind the subject. Third, due to the limitations of the underlying video generation model, *FaceCam* remains relatively slow at inference and is not yet suitable for real-time applications. Distilling the model or adopting a more efficient video generation backbone are promising directions.

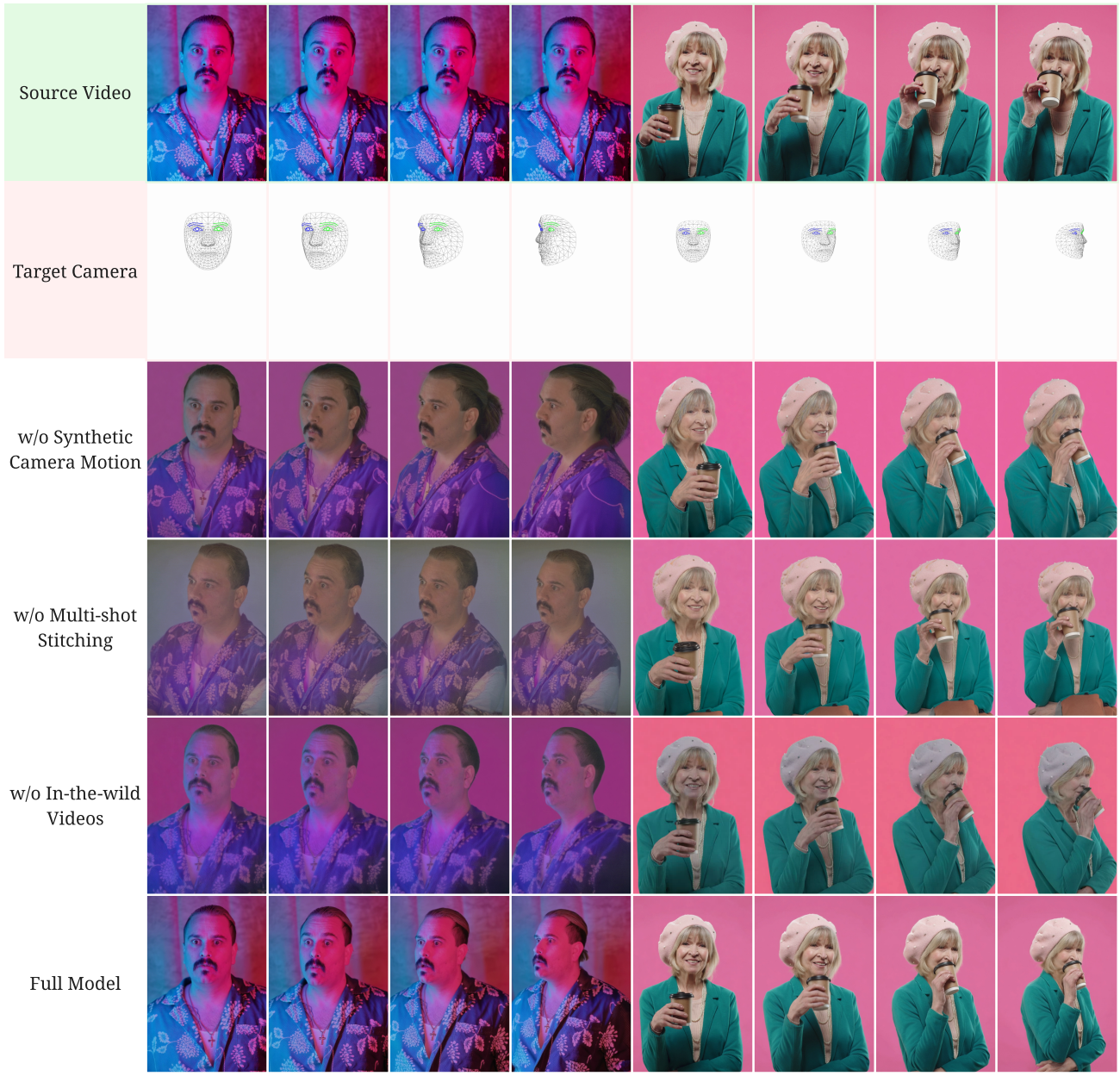


Figure 9. **Ablation study on training data generation.** Without Synthetic Camera Motion, the model often produces inaccurate camera trajectories with discontinuous or abrupt changes. Without Multi-shot Stitching, the model cannot learn to change camera angles along a trajectory. With both strategies applied but without in-the-wild videos (*w/o In-the-wild Videos*), the model generates correct camera motion and angle changes, but the lighting remains tied to the training distribution and fails to generalize to real-world illumination, leading to inconsistencies with the source video. Our full model provides accurate camera control and high image quality with lighting and appearance consistent with the source video.

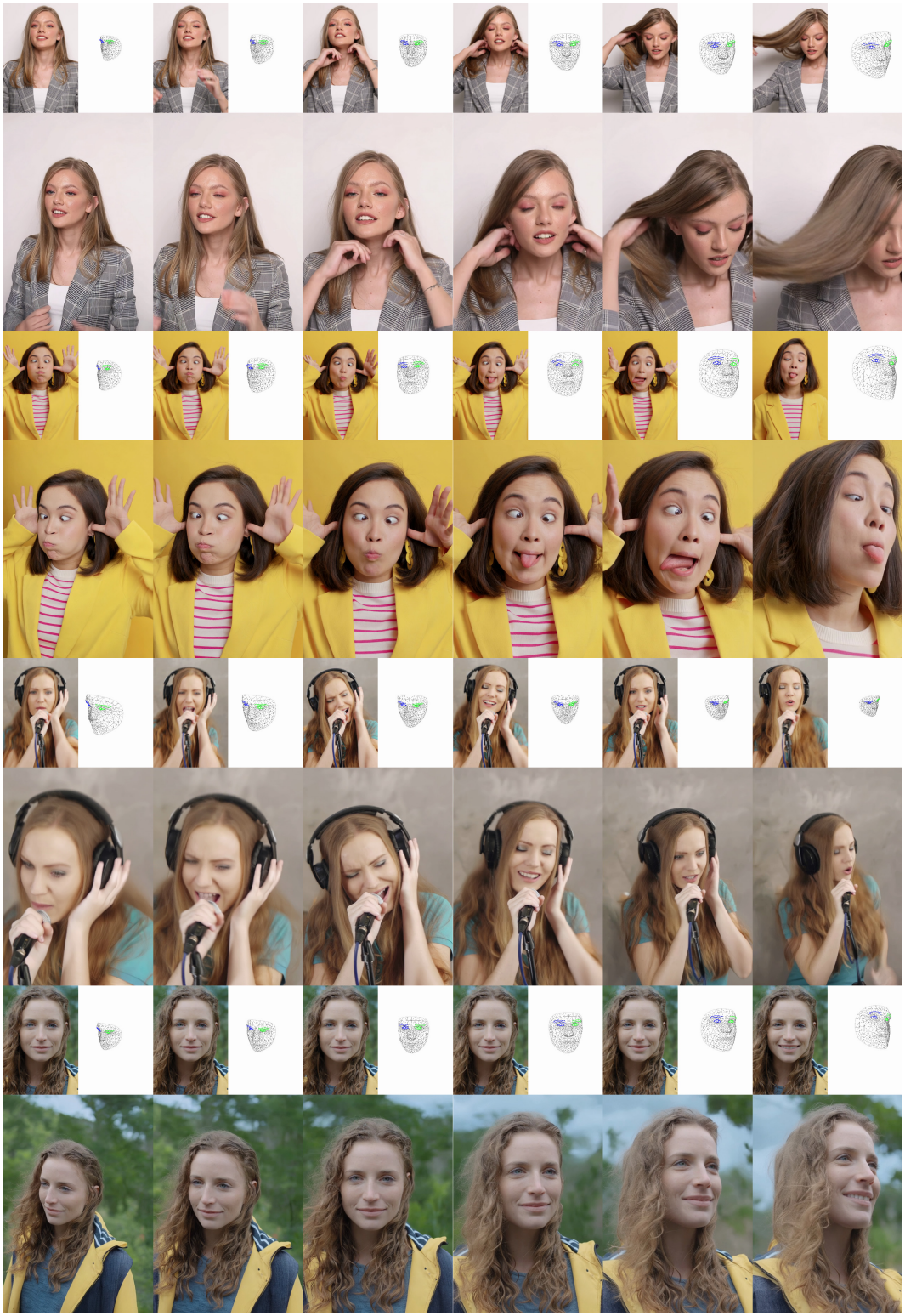


Figure 10. *FaceCam* performs robustly across diverse, challenging scenarios: it retains head and hair motion from the input video (example 1), captures a wide range of facial expressions (example 2), maintains motion-induced blur from fast body movements (example 3), and plausibly outpaints clothing and background when the generated video contains a smaller face region than the input (example 4).

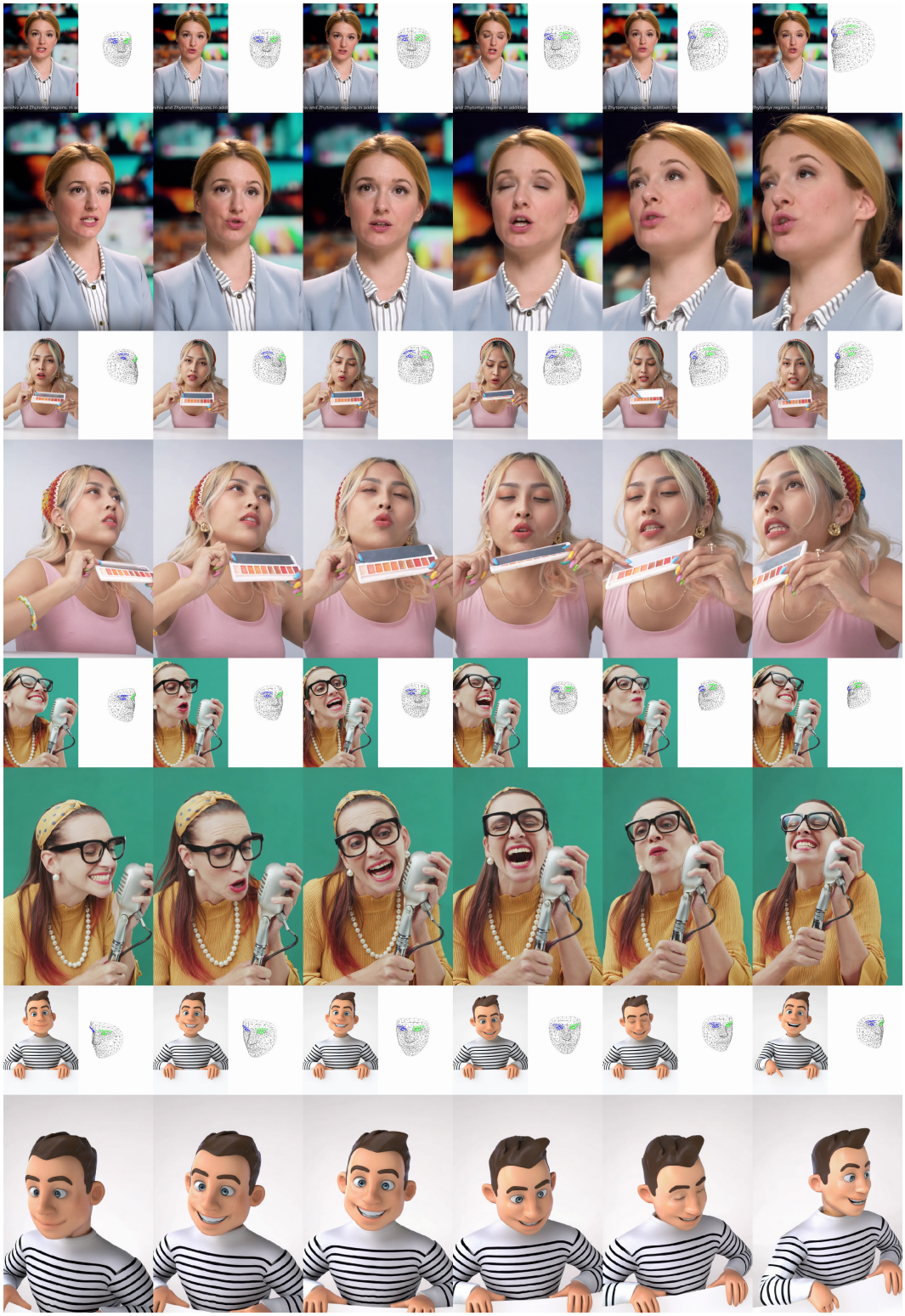


Figure 11. *FaceCam* in real-world scenarios. It recaptures a newscaster with detailed facial texture while keeping the studio background consistent (example 1). It further re-synthesizes an e-commerce streamer (example 2) and a singer (example 3) under novel camera angles, accurately maintaining co-occurring objects such as an eyeshadow palette, earrings, headband, necklace, microphone, and glasses, *etc*. The model even generalizes to cartoon characters (example 4), despite never having seen such content during training.