EPIPE: EPIPOLAR POSITIONAL ENCODING FOR CAMERA-CONTROLLED VIDEO GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Camera positional encodings provide an efficient way to condition video diffusion transformers on prescribed viewpoints. However, encodings based on homogeneous camera transformations do not directly express the epipolar compatibility of viewing rays, a geometric prior available before the scene content is synthesized. We introduce *Epipolar Positional Encoding (EpiPE)*, a relative camera positional encoding that incorporates this prior into learned attention scores. EpiPE applies the classical Plücker line transformation to learned features, exposing the coupling between translation and viewing direction that underlies epipolar geometry. This places epipolar coupling within a consistent coordinate transformation whose composition structure allows relative camera interactions to factor into per-token operations. The resulting encoding integrates epipolar geometry into standard attention without restricting which tokens may interact. Experiments on PanShot and RealEstate10K show improved camera-pose accuracy with comparable visual quality.

1 INTRODUCTION

Camera-controlled video generation supports broad applications in world action models Bar et al. (2025), autonomous driving, and virtual reality He et al. (2025b). As a conditional generation task, it requires synthesized videos to follow prescribed camera trajectories. The camera conditions thus determine viewing rays at each image location, imposing geometric constraints across views. Classical multi-view geometry (Hartley & Zisserman, 2004) relates these viewing rays through epipolar constraints, restricting which rays could observe the same static scene point. These constraints therefore provide a geometric prior for cross-view attention before the scene geometry is synthesized.

Existing epipolar attention methods exploit this prior by restricting cross-view interactions to candidates satisfying epipolar constraints (Zheng et al., 2024), but aggregating features along irregular epipolar lines incurs computational and memory overhead (Li et al., 2024). Alternatively, camera positional encodings retain efficient attention computation by encoding relative viewing geometry through per-token feature transformations (Miyato et al., 2024; Li et al., 2025; Zhang et al., 2026). However, their formulations based on homogeneous transformations do not explicitly capture the epipolar coupling between viewing rays (Miyato et al., 2024; Zhang et al., 2026). This raises our central question: *can we incorporate epipolar geometry into efficient camera positional encodings?*

Addressing this question requires considering both the geometric information encoded in attention and how it is computed. We therefore identify three design goals for camera positional encodings. **(1) Explicit epipolar coupling.** The encoding should expose the epipolar coupling, allowing the model to exploit this geometric prior. **(2) Efficient attention integration.** Relative geometric interactions should factor into independent per-token feature transformations, preserving efficient computation within standard attention. **(3) Consistency with camera geometry.** The encoded relationships should be invariant to rigid changes of the world coordinate system and retain information about relative rotation and translation. In particular, they should preserve rotation information when camera centers coincide, where the epipolar constraint becomes uninformative.

Our key insight is to associate each image token with a viewing line and use line transformations (Pless, 2003) to structure cross-view feature interactions. Under camera motion, the Plücker line transformation explicitly contains the translation–rotation coupling underlying epipolar geometry while retaining rotation information when translation vanishes. Building on this structure, we

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

introduce *EpiPE*, a camera positional encoding that applies Plücker line transformation operators to learned features. The resulting relative interactions thus factor into independent per-token transformations, enabling efficient attention computation. Unlike prior methods that use Plücker ray coordinates as conditioning inputs (He et al., 2025a), EpiPE uses their transformation law to encode relative viewing geometry within standard attention.

We integrate EpiPE into a pretrained Wan text-to-video diffusion transformer (Wan et al., 2025) and evaluate it on PanShot (Zhang et al., 2026) and RealEstate10K (Zhou et al., 2018) for camera-controlled video generation. EpiPE achieves lower relative camera-pose errors than existing camera positional encodings while maintaining comparable video quality. These improvements extend to RealEstate10K without further fine-tuning, demonstrating cross-dataset generalization. These results support incorporating epipolar geometry into camera positional encodings for controllable video generation.

Our contributions are threefold. (1) We introduce EpiPE, a camera positional encoding that incorporates epipolar coupling through Plücker line transformations while retaining efficient per-token computation. (2) We improve camera control in video generation, achieving lower camera-pose errors on PanShot and demonstrating transferability to RealEstate10K without fine-tuning, while maintaining comparable video quality. (3) We further demonstrate EpiPE's applicability beyond video generation and ease of integration by incorporating it into an existing stereo depth estimation model and improving its accuracy.

2 Related Work

Video generation and camera control. Video diffusion models have developed appearance and motion priors for video synthesis (Ho et al., 2022; Guo et al., 2024; Blattmann et al., 2023), with latent diffusion transformers extending these capabilities to large-scale text- and image-conditioned generation (Yang et al., 2025; Kong et al., 2024; Wan et al., 2025). Camera-conditioned methods adapt these generators to prescribed viewpoints and trajectories, including MotionCtrl (Wang et al., 2024), CameraCtrl (He et al., 2025a;b), VD3D (Bahmani et al., 2025b), and AC3D (Bahmani et al., 2025a). Source videos (Bai et al., 2025) and reconstructed scene geometry (Yu et al., 2025) provide additional guidance for novel views. Our setting is text-to-video generation conditioned on a camera trajectory, where the camera rays are available before the scene appearance and depth are generated.

Geometric constraints and supervision. Multi-view geometry provides constraints on the views generated along a camera trajectory. Epipolar attention uses these constraints to restrict cross-view token interactions (Zheng et al., 2024). Geometry can also inform training through output-level evaluation: epipolar consistency provides rankings for preference optimization (Kupyn et al., 2025), while agreement between estimated relative poses in generated and reference videos provides a training reward (Wang et al., 2026). These methods use geometry to define candidate interactions or assess generated samples. EpiPE incorporates the direction–translation coupling underlying the epipolar constraint into the camera encoding, where it contributes to learned attention scores.

Camera positional encoding. Camera encodings associate tokens with camera parameters or viewing rays. Plücker ray embeddings describe each ray by its direction and moment (He et al., 2025a). Geometric feature transformations provide a way to encode relative camera relationships: GTA applies geometric transformations to attention features (Miyato et al., 2024), PRoPE represents relative camera frustums (Li et al., 2025), and UCPE uses local ray frames to support different projection models (Zhang et al., 2026). SCoPE adds sightline coordinates to queries and keys, introducing content–ray and ray–ray interactions into the attention score (Yin et al., 2026). Our formulation builds on local ray frames and examines the transformation between them. By associating each image token with one viewing line, EpiPE uses the transformation law of Plücker lines to define relative positional encodings, explicitly incorporating epipolar coupling.

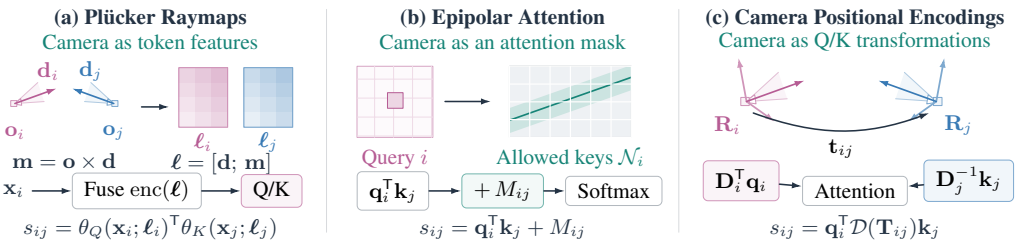

Figure 1: **Camera geometry in token interactions.** (a) **Pixel-aligned encodings** encode Plücker raymaps and fuse the resulting features with video features before query and key projection. (b) **Epipolar attention** uses the epipolar constraint to select candidate keys $\mathcal{N}_i$, masking interactions outside this set. (c) **Relative camera positional encodings** transform queries and keys independently so that their dot product depends on the relative camera transformation $\mathbf{T}_{ij}$. Scores are shown before scaling and softmax.

3 Method

3.1 Preliminaries

Camera-controlled video generation. Given a text prompt c and a camera sequence $\mathcal{C} = \{(\mathbf{R}_t, \mathbf{o}_t, \pi_t)\}_{t=0}^{T-1}$, a video diffusion transformer generates latent video tokens conditioned on these inputs. A decoder then maps the tokens to RGB frames. Here, $\mathbf{R}_t$ is the camera-to-world rotation, $\mathbf{o}_t$ is the camera center, and π_t is the camera projection, including intrinsics and distortion. For a central camera, $\pi_t^{-1}(\mathbf{u})$ gives the unit viewing direction corresponding to image location $\mathbf{u}$, expressed in camera coordinates. The corresponding viewing ray originates at $\mathbf{o}_t$ and has world-space direction $\mathbf{d}_{t,\mathbf{u}} = \mathbf{R}_t \pi_t^{-1}(\mathbf{u})$.

Within each attention head of the transformer, a token feature $\mathbf{x}_i$ is projected into a query $\mathbf{q}_i = \mathbf{W}_Q \mathbf{x}_i$, a key $\mathbf{k}_i = \mathbf{W}_K \mathbf{x}_i$, and a value $\mathbf{v}_i = \mathbf{W}_V \mathbf{x}_i$. Scaled dot-product attention computes

$$s_{ij} = \mathbf{q}_i^\top \mathbf{k}_j, \qquad a_{ij} = \frac{\exp(s_{ij}/\sqrt{d_h})}{\sum_r \exp(s_{ir}/\sqrt{d_h})}, \qquad \mathbf{y}_i = \sum_j a_{ij} \mathbf{v}_j, \tag{1}$$

where d_h is the query/key dimension and s_{ij} denotes the score before scaling and softmax. Positional encodings allow these scores to depend on where tokens are located. For video tokens, these positions span space and time, while camera geometry further describes the origins and directions of their viewing rays across frames.

Pixel-aligned camera encodings. Camera geometry can be represented by a raymap that associates each image location with its viewing ray. Using Plücker coordinates, each ray is described by its direction and moment, $\boldsymbol{\ell} = [\mathbf{d}; \mathbf{m}]$, where $\mathbf{m} = \mathbf{o} \times \mathbf{d}$ and $\mathbf{o}$ is the camera center. A raymap encoder extracts geometric features and fuses them with video features before query and key projection (He et al., 2025a), as illustrated in Figure 1(a). Writing the combined encoding, fusion, and projection operations schematically as θ_Q and θ_K, the attention score becomes

$$s_{ij}^{\text{raymap}} = \theta_Q(\mathbf{x}_i; \boldsymbol{\ell}_i)^\top \theta_K(\mathbf{x}_j; \boldsymbol{\ell}_j). \tag{2}$$

Raymaps provide a straightforward way to encode camera intrinsics and extrinsics. However, their coordinates depend on the chosen reference frame, leaving the model to learn consistent interactions across different coordinate descriptions of the same geometry (Li et al., 2025).

Epipolar attention. Camera geometry can also guide the selection of token interactions. Rays observing the same static scene point are coplanar with the baseline between their camera centers. Let $\mathbf{b}_i$ and $\mathbf{b}_j$ denote their unit viewing directions in their respective camera coordinate frames, and let $(\mathbf{R}_{ij}, \mathbf{t}_{ij})$ map coordinates from frame j to frame i. The epipolar constraint expresses this correspondence condition as

$$\mathbf{b}_i^\top \mathbf{E}_{ij} \mathbf{b}_j = 0, \qquad \mathbf{E}_{ij} = [\mathbf{t}_{ij}]_\times \mathbf{R}_{ij}, \tag{3}$$

where $[\mathbf{t}]_\times$ is the skew-symmetric matrix representing the cross product, such that $[\mathbf{t}]_\times \mathbf{u} = \mathbf{t} \times \mathbf{u}$. For a query ray, the constraint restricts possible correspondences in another view to an epipolar line (or a curve for distorted cameras). Epipolar attention selects candidate keys along this line or curve to form $\mathcal{N}_i$ (Zheng et al., 2024) (Figure 1(b)), giving the masked score

$$s_{ij}^{\text{epi}} = \mathbf{q}_i^\mathsf{T} \mathbf{k}_j + M_{ij}, \qquad M_{ij} = \begin{cases} 0, & j \in \mathcal{N}_i, \\ -\infty, & \text{otherwise.} \end{cases} \tag{4}$$

However, under general camera configurations, epipolar lines are not aligned with image grids, and sampling and aggregating features along these irregular lines can incur additional computation and memory overhead (Li et al., 2024).

Relative camera positional encodings. Relative camera positional encodings transform queries and keys so that their dot product depends on the geometry between tokens (Miyato et al., 2024; Li et al., 2025; Zhang et al., 2026). Following UCPE (Zhang et al., 2026), each token is associated with a local ray frame centered at its camera, with the z-axis aligned with its viewing ray and the remaining axes determined using the camera's y-axis. Let $\mathbf{T}_i$ map world coordinates into this frame. Then $\mathbf{T}_{ij} = \mathbf{T}_i \mathbf{T}_j^{-1} = [\mathbf{R}_{ij}, \mathbf{t}_{ij}] \in \mathrm{SE}(3)$ maps coordinates from token j's frame to token i's frame.

Following the geometric attention formulation of GTA (Miyato et al., 2024), each token is associated with a feature transformation $\mathbf{D} = \mathbf{I}_{d/4} \otimes \mathbf{T}$, where $\otimes$ denotes the Kronecker product, d is the feature dimension, and $\mathbf{I}$ is the identity matrix. Transforming queries by $\mathbf{D}_i^\mathsf{T}$ and keys by $\mathbf{D}_j^{-1}$ gives

$$s_{ij} = (\mathbf{D}_i^\mathsf{T} \mathbf{q}_i)^\mathsf{T} (\mathbf{D}_j^{-1} \mathbf{k}_j) = \mathbf{q}_i^\mathsf{T} \big(\mathbf{I}_{d/4} \otimes \mathbf{T}_{ij}\big) \mathbf{k}_j, \tag{5}$$

where $\mathbf{T}_{ij} = \mathbf{T}_i \mathbf{T}_j^{-1}$ is the relative transformation between the two local ray frames.

These homogeneous camera encodings use $\mathbf{T}_{ij}$ itself to transform four-channel feature blocks (Miyato et al., 2024; Zhang et al., 2026), which do not explicitly expose the epipolar coupling $[\mathbf{t}_{ij}]_\times \mathbf{R}_{ij}$ in the feature interaction.

3.2 EpiPE: Relative Positional Encodings with Epipolar Coupling

We aim to incorporate epipolar coupling directly into learned attention scores while retaining the per-token computation of relative positional encoding. By associating each token position with a Plücker line, their relative line transformation contains the epipolar coupling: transforming the line moment couples translation with the rotated line direction. This observation motivates *Epipolar Positional Encoding (EpiPE)*, which uses the line transformation to encode relative camera geometry in learned feature interactions (Figure 2).

Viewing line transformation. Consider the same viewing line expressed in two coordinate frames related by $\mathbf{p}' = \mathbf{R}\mathbf{p} + \mathbf{t}$. Its Plücker coordinates are $[\mathbf{d}; \mathbf{m}]$, where $\mathbf{m} = \mathbf{p} \times \mathbf{d}$ for any point $\mathbf{p}$ on the line. The direction and moment transform in-between these two coordinate frames as

$$\mathbf{d}' = \mathbf{R}\mathbf{d}, \qquad \mathbf{m}' = (\mathbf{R}\mathbf{p} + \mathbf{t}) \times \mathbf{R}\mathbf{d} = \mathbf{R}\mathbf{m} + [\mathbf{t}]_\times \mathbf{R}\mathbf{d}. \tag{6}$$

For EpiPE, we choose this classical Plücker line transformation (Pless, 2003; Miraldo & Araujo, 2015) as our representation, denoted by $\mathcal{D}$ in the remainder of this section. For $\mathbf{T} = [\mathbf{R}, \mathbf{t}]$,

$$\mathcal{D}(\mathbf{T}) = \begin{bmatrix} \mathbf{R} & \mathbf{0} \\ [\mathbf{t}]_\times \mathbf{R} & \mathbf{R} \end{bmatrix}, \qquad \begin{bmatrix} \mathbf{d}' \\ \mathbf{m}' \end{bmatrix} = \mathcal{D}(\mathbf{T}) \begin{bmatrix} \mathbf{d} \\ \mathbf{m} \end{bmatrix}. \tag{7}$$

For the relative transformation $\mathbf{T}_{ij}$, the lower-left block is precisely the essential matrix $\mathbf{E}_{ij}$. The transformation thus contains epipolar coupling alongside the rotation of direction and moment. To use this geometric action as a relative positional encoding, we also need its relative interactions to factor into per-token feature transformations.

From line transformations to relative positional encoding. The viewing line transformation preserves composition and inverse:

$$\begin{aligned} \mathcal{D}(\mathbf{T}_1 \mathbf{T}_2) &= \mathcal{D}(\mathbf{T}_1) \mathcal{D}(\mathbf{T}_2), \\ \mathcal{D}(\mathbf{T}^{-1}) &= \mathcal{D}(\mathbf{T})^{-1}. \end{aligned} \tag{8}$$

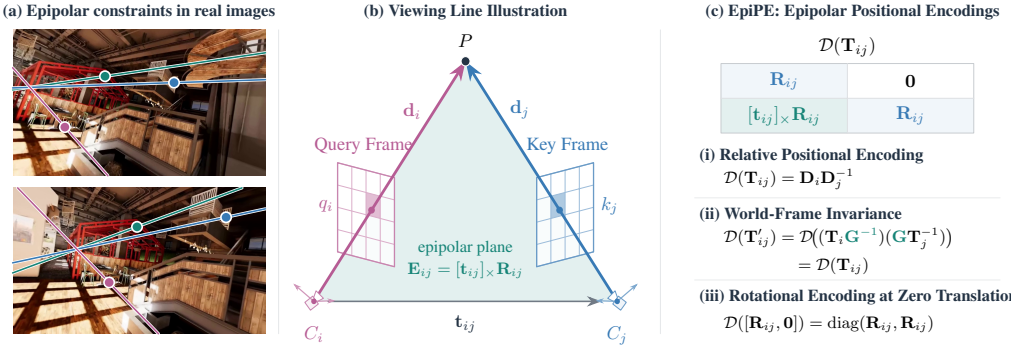

Figure 2: **Viewing lines and their relative transformation.** (a) Three matched point pairs and their pose-derived epipolar lines in real reference frames; matching colors identify correspondences. (b) Viewing lines through the highlighted query and key patches intersect at P, defining an epipolar plane with the camera baseline. (c) The Plücker line transformation contains epipolar coupling and is consistent with camera geometry. Here $\mathbf{D}_i = \mathcal{D}(\mathbf{T}_i)$, $\mathbf{T}_i$ maps world coordinates to token i's ray frame, and $\mathbf{G} \in \mathrm{SE}(3)$ is a shared change of world coordinates, with $\mathbf{T}'_{ij} = \mathbf{T}'_i(\mathbf{T}'_j)^{-1}$.

Consequently, its relative action factors into per-token transformations:

$$\mathcal{D}(\mathbf{T}_{ij}) = \mathcal{D}(\mathbf{T}_i)\mathcal{D}(\mathbf{T}_j)^{-1} = \mathbf{D}_i \mathbf{D}_j^{-1}, \tag{9}$$

where $\mathbf{D}_i = \mathcal{D}(\mathbf{T}_i)$. This relative action is also invariant to a shared rigid change of world coordinates. For such a change $\mathbf{G} \in \mathrm{SE}(3)$, the world-to-ray maps become $\mathbf{T}'_i = \mathbf{T}_i \mathbf{G}^{-1}$, giving

$$\mathcal{D}\big(\mathbf{T}'_i(\mathbf{T}'_j)^{-1}\big) = \mathcal{D}\big((\mathbf{T}_i\mathbf{G}^{-1})(\mathbf{T}_j\mathbf{G}^{-1})^{-1}\big) = \mathcal{D}(\mathbf{T}_i\mathbf{T}_j^{-1}). \tag{10}$$

The factorization in Eq. (9) lets us compute relative interactions through independent query and key transformations. For each six-channel feature block, we use

$$\widehat{\mathbf{q}}_i = \mathbf{D}_i^\mathsf{T} \mathbf{q}_i, \qquad \widehat{\mathbf{k}}_j = \mathbf{D}_j^{-1} \mathbf{k}_j. \tag{11}$$

The transpose and inverse used in these transformations have the explicit forms

$$\mathcal{D}(\mathbf{T})^\mathsf{T} = \begin{bmatrix} \mathbf{R}^\mathsf{T} & -\mathbf{R}^\mathsf{T}[\mathbf{t}]_\times \\ \mathbf{0} & \mathbf{R}^\mathsf{T} \end{bmatrix}, \qquad \mathcal{D}(\mathbf{T})^{-1} = \begin{bmatrix} \mathbf{R}^\mathsf{T} & \mathbf{0} \\ -\mathbf{R}^\mathsf{T}[\mathbf{t}]_\times & \mathbf{R}^\mathsf{T} \end{bmatrix}. \tag{12}$$

Taking the dot product of the transformed query and key yields

$$\widehat{\mathbf{q}}_i^\mathsf{T} \widehat{\mathbf{k}}_j = \mathbf{q}_i^\mathsf{T} \mathbf{D}_i \mathbf{D}_j^{-1} \mathbf{k}_j = \mathbf{q}_i^\mathsf{T} \mathcal{D}(\mathbf{T}_{ij}) \mathbf{k}_j, \tag{13}$$

placing the relative viewing line transformation inside the attention score. Whereas raymap conditioning supplies Plücker coordinates as inputs, EpiPE applies their transformation law to learned query and key features.

3.3 Properties of EpiPE

Epipolar coupling in attention. EpiPE makes the coupling underlying the epipolar constraint explicit in learned feature interactions. Writing each six-channel block as $\mathbf{q}_i = [\mathbf{q}_{i,d}; \mathbf{q}_{i,m}]$ and $\mathbf{k}_j = [\mathbf{k}_{j,d}; \mathbf{k}_{j,m}]$, the attention score expands as

$$s_{ij} = \mathbf{q}_i^\mathsf{T} \mathcal{D}(\mathbf{T}_{ij}) \mathbf{k}_j = \mathbf{q}_{i,d}^\mathsf{T} \mathbf{R}_{ij} \mathbf{k}_{j,d} + \mathbf{q}_{i,m}^\mathsf{T} \mathbf{R}_{ij} \mathbf{k}_{j,m} + \underbrace{\mathbf{q}_{i,m}^\mathsf{T} \mathbf{E}_{ij} \mathbf{k}_{j,d}}_{\text{epipolar coupling}}. \tag{14}$$

The final term couples the learned triplets through $\mathbf{E}_{ij} = [\mathbf{t}_{ij}]_\times \mathbf{R}_{ij}$. Thus, this translation–direction coupling directly participates in the score.

Directional interactions at zero translation. Using the essential coupling alone would remove the geometric interaction when camera centers coincide, since $\mathbf{t}_{ij} = \mathbf{0}$ implies $\mathbf{E}_{ij} = \mathbf{0}$. Viewing directions can still differ in this case, both during camera rotation and between rays within the same frame. The viewing line transformation retains these relationships through its rotational blocks:

$$\mathcal{D}([\mathbf{R}_{ij}, \mathbf{0}]) = \mathrm{diag}(\mathbf{R}_{ij}, \mathbf{R}_{ij}). \tag{15}$$

The first two terms in Eq. (14) can therefore encode directional relationships even when the epipolar term vanishes.

Efficient computation. EpiPE encodes geometry by transforming queries and keys independently before attention. Each transformed feature is reused across all pairwise interactions, as in relative positional encoding. This allows epipolar coupling to enter standard attention without constructing or sampling epipolar lines. The geometric encoding therefore adds only linear overhead in the number of tokens. The same factorization extends to value aggregation (Miyato et al., 2024), where values are transformed before attention and outputs afterward:

$$\widehat{\mathbf{v}}_j = \mathbf{D}_j^{-1}\mathbf{v}_j, \qquad \mathbf{y}_i = \mathbf{D}_i \sum_j a_{ij}\widehat{\mathbf{v}}_j = \sum_j a_{ij}\mathcal{D}(\mathbf{T}_{ij})\mathbf{v}_j. \tag{16}$$

3.4 Application to Camera-Controlled Text-to-Video Generation

Constructing token transformations. We construct the world-to-ray maps $\mathbf{T}_i$ from the prescribed cameras using UCPE's ray frames (Zhang et al., 2026), supporting unified camera models. Specifically, for a token associated with image location $\mathbf{u}$ and source frame t, the ray-frame origin is the camera center $\mathbf{o}_t$, and its z-axis is the unit viewing direction $\mathbf{d}_{t,\mathbf{u}}$. Using the camera's world-space y-axis $\mathbf{c}_t^y = \mathbf{R}_t(0,1,0)^\mathsf{T}$ to fix the remaining axes gives

$$\mathbf{e}_i^z = \mathbf{d}_{t,\mathbf{u}}, \qquad \mathbf{e}_i^x = \frac{\mathbf{c}_t^y \times \mathbf{e}_i^z}{\|\mathbf{c}_t^y \times \mathbf{e}_i^z\|}, \qquad \mathbf{e}_i^y = \mathbf{e}_i^z \times \mathbf{e}_i^x. \tag{17}$$

The world-to-ray rotation and full transformation are

$$\mathbf{R}_i^{w2r} = \begin{bmatrix} (\mathbf{e}_i^x)^\mathsf{T} \\ (\mathbf{e}_i^y)^\mathsf{T} \\ (\mathbf{e}_i^z)^\mathsf{T} \end{bmatrix}, \qquad \mathbf{T}_i = \begin{bmatrix} \mathbf{R}_i^{w2r} & -\mathbf{R}_i^{w2r}\mathbf{o}_t \\ \mathbf{0}^\mathsf{T} & 1 \end{bmatrix}, \qquad \mathbf{D}_i = \mathcal{D}(\mathbf{T}_i). \tag{18}$$

Integration with the video transformer. We adopt UCPE's parallel two-branch design (Zhang et al., 2026): the pretrained self-attention branch is retained, and a trainable camera-attention branch uses EpiPE to condition token interactions on the prescribed cameras. The camera branch combines spatial 2D RoPE with EpiPE on separate feature channels and applies the Q/K and V/O transformations described above. The camera-branch output passes through a zero-initialized projection before being added to the pretrained self-attention output. At initialization, the model therefore reproduces the backbone output. We freeze the Wan backbone and train the camera adapter (Wan et al., 2025).

Relative geometry describes relationships between viewpoints but does not specify their orientation relative to a world-up direction. We therefore retain UCPE's absolute orientation conditioning in the camera-branch. Camera preprocessing removes the first-frame yaw while retaining pitch and roll. This absolute conditioning complements EpiPE's relative geometry; the world-coordinate invariance in Section 3.2 concerns the relative operator. More details are given in Appendix A.

4 Experiments

Overview. We evaluate whether EpiPE improves camera control for video generation and ablate the epipolar coupling component under varied camera movements. We also examine the applicability of EpiPE by incorporating it into an existing stereo depth estimation model. Specifically, we first compare camera control and visual quality on PanShot Zhang et al. (2026) and zero-shot transfer to RealEstate10K Zhou et al. (2018), then analyze the epipolar coupling through an encoding ablation across different camera baselines. We use the UniMatch Xu et al. (2023) for the stereo depth estimation.

4.1 Camera-controlled Video Generation

4.1.1 Experimental setting

Experimental settings. We train a 35.64M-parameter camera adapter on a frozen Wan2.1-T2V-1.3B backbone (Wan et al., 2025), using 47,920 PanShot clips for 5,000 steps with global batch size 16. We evaluate within the training-data distribution on 272 held-out PanShot conditions and assess zero-shot transfer on 100 RealEstate10K conditions without further fine-tuning (Zhou et al., 2018).

Table 1: **Camera control and inference time.** All methods use the same trajectory-estimation protocol. Time is measured as the mean seconds per 81-frame video over three PanShot inputs on one H200, at 480×832 resolution with 50 sampling steps. Best and second best in each metric.

	PanShot Within-data distribution			RealEstate10K Zero-shot transfer			Inference Single H200
Method	RotErr	TransErr	CamMC	RotErr	TransErr	CamMC	Time (s)↓
ReCamMaster (T2V)	11.027	34.224	40.109	1.357	7.073	7.601	**91.97**
Epipolar attention	10.537	51.959	57.243	1.644	11.777	12.231	187.61
SCoPE (Wan2.1)	9.150	22.312	27.511	1.333	2.722	3.593	96.35
UCPE	6.088	17.197	20.666	0.750	1.427	1.923	102.15
EpiPE	**4.526**	**15.702**	**18.147**	**0.595**	**1.291**	**1.664**	104.31

Table 2: **Video quality and training configurations.** Trainable parameter counts (M) refer to the evaluated Wan2.1 adaptations. ✓ denotes a fully frozen backbone; ✗ denotes fine-tuning of selected backbone layers. FVD/FID use PanShot references; CLIP measures text alignment. IQ, IA, and VQ are normalized Q-Align image quality, image aesthetics, and video quality.

	Training		PanShot Within-data distribution			RealEstate10K Zero-shot transfer		
Method	Trainable params. (M)	Frozen backbone	FVD↓	FID↓	CLIP↑	IQ↑	IA↑	VQ↑
ReCamMaster (T2V)	354.91	✗	508.03	64.39	25.10	0.8035	0.3961	0.8534
Epipolar attention	35.82	✓	529.65	65.18	25.08	**0.9622**	**0.4237**	**0.9751**
SCoPE (Wan2.1)	1133.33	✗	**434.79**	63.08	**25.25**	0.8613	0.4216	0.9074
UCPE	35.64	✓	552.19	**62.55**	25.17	0.8809	0.4147	0.9203
EpiPE	35.64	✓	504.93	65.01	25.14	0.8886	0.4025	0.9254

Each condition specifies a text prompt and camera trajectory, with no input image. All methods generate 81-frame videos at 480×832 resolution using shared prompts, camera conditions, and a generation seed. Appendix A provides complete training and inference settings.

Baselines and Metrics. UCPE (Zhang et al., 2026) provides the closest comparison, using homogeneous transformations in the same ray frames, with the same adapter capacity, spatial RoPE, and absolute orientation conditioning. We also compare ReCamMaster (Bai et al., 2025), epipolar attention based on CamI2V (Zheng et al., 2024), and SCoPE (Yin et al., 2026), adapted to Wan text-to-video generation and unified cameras. We use ViPE (Huang et al., 2025) to estimate camera trajectories from generated videos after rectification to pinhole views, and evaluate them against the target trajectories using rotation error (RotErr), translation error (TransErr), and camera motion consistency (CamMC). Following Zhang et al. (2026), estimated and target trajectories are aligned to their respective first frames, and translations are independently normalized by maximum displacement. We assess visual quality with FVD, FID, and CLIP on PanShot (Unterthiner et al., 2018; Heusel et al., 2017; Radford et al., 2021), and Q-Align on RealEstate10K (Wu et al., 2024).

4.1.2 Comparisons

Camera control. EpiPE follows the prescribed camera motion more accurately across both rotation and normalized translation on PanShot, achieving the lowest errors among all evaluated methods (Table 1). Its advantage over ReCamMaster, SCoPE, and epipolar attention spans all three control metrics, indicating better agreement in both viewing direction and camera path. The comparison with UCPE further shows that this improvement is attainable at the same adapter capacity: rotation and translation errors decrease by 25.7% and 8.7%, respectively.

Transfer to RealEstate10K. The control advantage persists when the PanShot-trained adapters are applied to RealEstate10K without fine-tuning (Table 1). EpiPE again achieves the lowest errors in all three control metrics, ahead of both the alternative conditioning methods and UCPE. This consistent ranking across datasets supports the transfer of learned camera conditioning beyond the training domain, with gains in both viewing direction and normalized camera path.

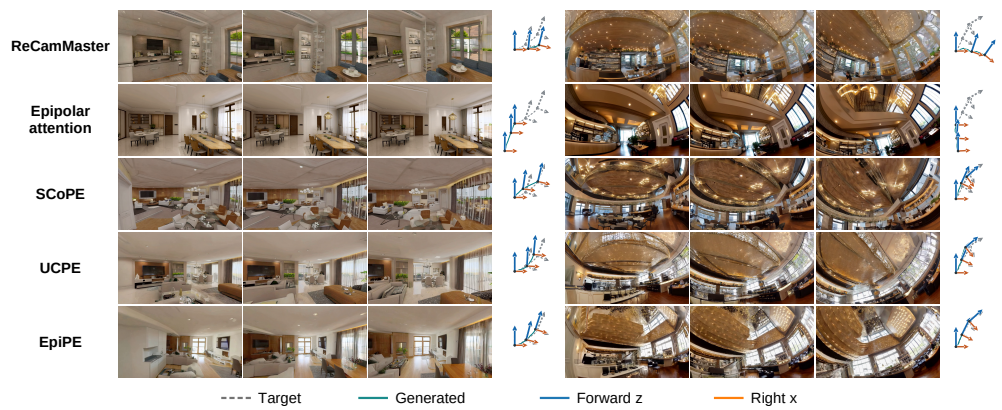

Figure 3: **Qualitative camera control on RealEstate10K (left) and PanShot (right).** Rows compare five methods under matched prompts and target camera trajectories; frames are sampled at the beginning, middle, and end of each video. Trajectory plots overlay the target (gray dashed) and the estimate (teal), with camera-right (orange) and forward (blue) axes.

Visual quality. The control gains coexist with competitive visual quality (Table 2). We would like to clarify that, similar to UCPE, our method does not fine-tune the video generation backbone, with modest training parameters. Relative to UCPE, EpiPE lowers PanShot FVD from 552.19 to 504.93, with similar CLIP alignment and a higher FID. Perceptual scores should be interpreted alongside motion accuracy: epipolar attention achieves the highest Q-Align scores on RealEstate10K, yet it performs worst for the camera control. This case illustrates that visually convincing videos can still under-respond to the requested camera motion. EpiPE's main advantage is improved camera control while retaining comparable visual quality.

Computational efficiency. Table 1 reports inference time on a single H200 under shared generation settings. EpiPE takes roughly 100 seconds per video, similar to UCPE. Relative to epipolar attention, EpiPE reduces inference time by 44.4%. These measurements show that EpiPE retains generation speed close to UCPE while incorporating epipolar coupling through per-token transformations. Timing averages three shared PanShot inputs after warmup, detailed in Appendix A.

Qualitative Results. Figure 3 compares two matched examples from RealEstate10K and PanShot. In the living-room example (left), EpiPE follows the rotations more closely than baselines. In the wide-angle interior (right), EpiPE preserves distorted frames as the viewpoint changes, and its recovered trajectory follows the target more closely than baselines. The visualized trajectories show normalized translation paths and projected camera axes (top-down view), making position and orientation differences visible together.

4.1.3 Ablations and Analysis

Contribution of epipolar coupling. To examine the epipolar coupling part of EpiPE, we train a rotation-only encoding that replaces $\mathcal{D}(\mathbf{T})$ with $\mathrm{diag}(\mathbf{R}, \mathbf{R})$ and retains other parts. Across the full evaluation sets, removing the coupling increases TransErr, but the corresponding changes in RotErr are moderate. These results support the contribution of the epipolar coupling to camera control.

Dependence on camera baseline. We further group the same conditions by their **input target camera baseline**, defined as the maximum camera-center displacement from the initial frame before per-trajectory normalization. Near-static conditions have baseline below 0.05 in the dataset's camera-coordinate scale; the remaining conditions are divided into small, medium, and large groups using within-dataset tertiles.We report only RotErr for the near-static group because independent translation normalization can amplify estimation jitter when the target displacement is close to zero. Implementation and evaluation details are given in Appendix A.

Table 3: **Quantitative results across varied target camera baselines.** Near-static conditions report RotErr only. Parentheses indicate group sizes.

	PanShot									
	Near-static (39)	Small (78)			Medium (77)			Large (78)		
Method	RotErr	RotErr	TransErr	CamMC	RotErr	TransErr	CamMC	RotErr	TransErr	CamMC
UCPE	5.632	**3.535**	15.932	17.290	7.714	9.379	14.669	7.262	10.751	15.522
Rotation-only	4.271	4.145	58.657	59.364	**5.103**	55.206	56.611	6.009	56.240	58.118
EpiPE	**3.804**	3.594	**13.520**	**15.006**	5.525	**8.541**	**12.108**	**4.834**	**8.375**	**11.529**
	RealEstate10K									
	Near-static (0)	Small (33)			Medium (33)			Large (34)		
Method	RotErr	RotErr	TransErr	CamMC	RotErr	TransErr	CamMC	RotErr	TransErr	CamMC
UCPE	–	0.738	**1.721**	2.183	0.529	1.221	1.504	0.978	1.340	2.078
Rotation-only	–	0.797	11.830	11.956	**0.481**	11.473	11.524	0.892	12.071	12.243
EpiPE	–	**0.490**	1.777	**2.008**	0.510	**1.097**	**1.374**	**0.778**	**1.008**	**1.612**

Table 4: **Stereo depth estimation using UniMatch.**

	AbsRel↓			RMSE↓		
Encoding	RGBD	SUN3D	Scenes11	RGBD	SUN3D	Scenes11
UniMatch	0.1889	0.1836	0.1353	0.9558	0.5262	1.0117
+PRoPE	−28.58%	−26.21%	−32.62%	−21.78%	−24.58%	−32.32%
+UCPE	−31.19%	−26.85%	−38.52%	−26.35%	−24.39%	−32.86%
+EpiPE	−32.37%	−28.35%	−41.74%	−26.40%	−25.10%	−35.12%

The rotation-only encoding has substantially higher TransErr and CamMC than both translation-aware encodings in every moving group. Its rotation errors remain competitive, including lower RotErr than full EpiPE in the medium-baseline groups. On the near-static conditions, EpiPE has lower RotErr than both UCPE and the rotation-only encoding (Table 3). This contrast indicates that retaining rotational information is insufficient for following camera translations and how to inject the translation influences the overall performance. We observe that the benefits of epipolar coupling become more obvious when the camera baseline becomes larger, aligning with the characteristics of the epipolar geometry.

4.2 Task Generalization: Depth Estimation

Experimental setting. We train UniMatch (Xu et al., 2023) on DeMoN from scratch for 50k steps with global batch size 20, comparing its original positional encoding with PRoPE (Li et al., 2025), UCPE, and EpiPE at equal trainable parameter counts. We evaluate absolute relative depth error (AbsRel) and root mean squared error (RMSE) on three subsets. Table 4 reports absolute errors for UniMatch and relative changes for the other encodings; lower is better. Appendix A.1 provides architecture, training, and evaluation details.

Results. EpiPE achieves the lowest reported errors on all three subsets, reducing AbsRel by 28.35–41.74% and RMSE by 25.10–35.12% relative to UniMatch. These results demonstrate that our EpiPE could be applied to two-view depth estimation with better performance.

5 Conclusion

We introduced EpiPE, a camera positional encoding that exposes epipolar coupling through Plücker line transformations while retaining efficient per-token computation. Experiments demonstrate improved camera control with comparable video quality, with additional gains in two-view depth estimation. These results support incorporating explicit geometric structure into learned attention interactions.

Limitations and future work. As a positional encoding, EpiPE depends on its backbone's capabilities and assumes known camera poses. Future work will explore stronger backbones and generalization to broader tasks and camera motions.

AI USE STATEMENT

In this work, we used generative AI tools for helping create scientific figures, assist implementation, improve readability. We have reviewed all AI-assisted work. We have not used generative AI tools for other tasks with required disclosure, and the rest of the required disclosure tasks are not applicable to this work. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

We have included the architecture and implementation details in the main paper and supplementary. We will release codes in the final version.

REFERENCES

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*, pp. 22875–22889, 2025a.

Sherwin Bahmani, Ivan Skorokhodov, Aliaksandr Siarohin, Willi Menapace, Guocheng Qian, Michael Vasilkovsky, Hsin-Ying Lee, Chaoyang Wang, Jiaxu Zou, Andrea Tagliasacchi, et al. Vd3d: Taming large video diffusion transformers for 3d camera control. In *ICLR*, 2025b.

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.

Amir Bar, Gaoyue Zhou, Danny Tran, Trevor Darrell, and Yann LeCun. Navigation world models. In *CVPR*, pp. 15791–15801, 2025.

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.

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. In *ICLR*, 2024.

R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, ISBN: 0521540518, second edition, 2004.

Hao He, Yinghao Xu, Yuwei Guo, Gordon Wetzstein, Bo Dai, Hongsheng Li, and Ceyuan Yang. Cameractrl: Enabling camera control for video diffusion models. In *The Thirteenth International Conference on Learning Representations*, 2025a. URL https://openreview.net/forum?id=Z4evOUYrk7.

Hao He, Ceyuan Yang, Shanchuan Lin, Yinghao Xu, Meng Wei, Liangke Gui, Qi Zhao, Gordon Wetzstein, Lu Jiang, and Hongsheng Li. Cameractrl ii: Dynamic scene exploration via camera-controlled video diffusion models. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 13416–13426. IEEE, 2025b.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *NeurIPS*, 30, 2017.

Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *Advances in neural information processing systems*, 35:8633–8646, 2022.

Jiahui Huang, Qunjie Zhou, Hesam Rabeti, Aleksandr Korovko, Huan Ling, Xuanchi Ren, Tianchang Shen, Jun Gao, Dmitry Slepichev, Chen-Hsuan Lin, et al. Vipe: Video pose engine for 3d geometric perception. *arXiv preprint arXiv:2508.10934*, 2025.

Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, Bo Wu, Jianwei Zhang, et al. Hunyuanvideo: A systematic framework for large video generative models. *arXiv preprint arXiv:2412.03603*, 2024.

Orest Kupyn, Fabian Manhardt, Federico Tombari, and Christian Rupprecht. Epipolar geometry improves video generation models. *arXiv preprint arXiv:2510.21615*, 2025. URL https://arxiv.org/abs/2510.21615.

Peng Li, Yuan Liu, Xiaoxiao Long, Feihu Zhang, Cheng Lin, Mengfei Li, Xingqun Qi, Shanghang Zhang, Wenhan Luo, Ping Tan, et al. Era3d: High-resolution multiview diffusion using efficient row-wise attention. *Advances in Neural Information Processing Systems*, 37:55975–56000, 2024.

Ruilong Li, Brent Yi, Junchen Liu, Hang Gao, Yi Ma, and Angjoo Kanazawa. Cameras as relative positional encoding. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=kK8cFKu1U7.

Christopher Mei and Patrick Rives. Single view point omnidirectional camera calibration from planar grids. In *ICRA*, pp. 3945–3950. IEEE, 2007.

Pedro Miraldo and Helder Araujo. Generalized essential matrix: Properties of the singular value decomposition. *Image and Vision Computing*, 34:45–50, 2015. doi: 10.1016/j.imavis.2014.11.003.

Takeru Miyato, Bernhard Jaeger, Max Welling, and Andreas Geiger. Gta: A geometry-aware attention mechanism for multi-view transformers. In *ICLR*, 2024.

Robert Pless. Using many cameras as one. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2003. doi: 10.1109/CVPR.2003.1211520.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, pp. 8748–8763. PmLR, 2021.

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

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, et al. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

Zhaoqing Wang, Xiaobo Xia, Zhuolin Bie, Jinlin Liu, Dongdong Yu, Jia-Wang Bian, and Changhu Wang. Taming camera-controlled video generation with verifiable geometry reward. In *European Conference on Computer Vision*, pp. 625–642. Springer, 2026.

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 Conf. Papers*, pp. 1–11, 2024.

Haoning Wu, Zicheng Zhang, Weixia Zhang, Chaofeng Chen, Liang Liao, Chunyi Li, Yixuan Gao, Annan Wang, Erli Zhang, Wenxiu Sun, et al. Q-align: teaching lmms for visual scoring via discrete text-defined levels. In *ICML*, pp. 54015–54029, 2024.

Haofei Xu, Jing Zhang, Jianfei Cai, Hamid Rezatofighi, Fisher Yu, Dacheng Tao, and Andreas Geiger. Unifying flow, stereo and depth estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. Cogvideox: Text-to-video diffusion models with an expert transformer. In *ICLR*, 2025.

Minghao Yin, Jiahao Lu, Wenbo Hu, Wang Zhao, Ying Shan, and Kai Han. Scope: Sightline-coordinate positional encoding for video diffusion transformers. *arXiv preprint arXiv:2606.27345*, 2026.

Wangbo Yu, Jinbo Xing, Li Yuan, Wenbo Hu, Xiaoyu Li, Zhipeng Huang, Xiangjun Gao, Tien-Tsin Wong, Ying Shan, and Yonghong Tian. Viewcrafter: Taming video diffusion models for high-fidelity novel view synthesis. *IEEE TPAMI*, 2025.

Cheng Zhang, Boying Li, Meng Wei, Yan-Pei Cao, Camilo Gambardella, Dinh Phung, and Jianfei Cai. Unified camera positional encoding for controlled video generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 38027–38037, 2026.

Guangcong Zheng, Teng Li, Rui Jiang, Yehao Lu, Tao Wu, and Xi Li. Cami2v: Camera-controlled image-to-video diffusion model. *arXiv preprint arXiv:2410.15957*, 2024.

Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely. Stereo magnification: learning view synthesis using multiplane images. *ACM TOG*, 37(4):1–12, 2018.

A Implementation and Evaluation Details

Architecture and training. We follow UCPE's parallel camera-adapter design with a zero-initialized output projection, keeping the Wan backbone frozen (Zhang et al., 2026; Wan et al., 2025). The initial model therefore reproduces the backbone output. For clean latent $\mathbf{z}_0$, Gaussian noise $\boldsymbol{\epsilon}$, and diffusion noise level σ, let $\mathbf{z}_\sigma = (1-\sigma)\mathbf{z}_0 + \sigma\boldsymbol{\epsilon}$. We use the backbone's weighted flow-matching objective,

$$\mathcal{L}_{\text{train}} = \mathbb{E}\Big[w(\sigma)\,\|v_{\theta,\phi}(\mathbf{z}_\sigma, \sigma, c, \mathcal{C}) - (\boldsymbol{\epsilon} - \mathbf{z}_0)\|_2^2\Big], \tag{19}$$

where θ is frozen and ϕ denotes the trainable adapter parameters.

Attention configuration. We use the unified camera model for distortion (Mei & Rives, 2007). Following UCPE, each 192-channel head allocates 96 channels to spatial RoPE and 96 to geometry. The latter accommodate 24 homogeneous four-channel blocks or 16 six-channel blocks for EpiPE, keeping the feature budget fixed. Both the baseline and EpiPE use endpoint camera poses for temporally compressed tokens. We retain UCPE's separate absolute orientation map, using the camera-space up direction $\mathbf{R}_t^\mathsf{T}\mathbf{g}$, $\mathbf{g} = (0, -1, 0)^\mathsf{T}$, after removing only the first-frame yaw.

Training and inference. The camera branch uses attention compression factor 8, and gradient clipping at norm 1.0. Each global batch contains four clips on each of four GPUs, and evaluation uses the final 5,000-step checkpoint. Generation uses the same seed with shared prompts, camera conditions, output dimensions, and sampling settings.

Inference-time measurement. We benchmark the five Wan2.1-T2V-1.3B adaptations sequentially on the same NVIDIA H200 using the same configuration. All methods generate 81 frames at 480×832 with 50 denoising steps and the same seed 0. All weights remain on the GPU, with no CPU offloading and untiled VAE decoding. We select three shared inputs. For each method and input, one full warmup generation precedes one timed generation; Table 1 reports the arithmetic mean of the three timed runs. CUDA-synchronized wall time includes text encoding, camera conditioning, denoising, VAE decoding, and conversion to output frames, but excludes model loading, input loading and transfer, and video encoding or disk I/O.

Baseline implementations. For ReCamMaster (Bai et al., 2025), we use the reimplementation provided by UCPE (Zhang et al., 2026). For SCoPE (Yin et al., 2026), we replace its backbone with Wan2.1-T2V-1.3B, the same backbone used by the other baselines.

Relative pose metrics. We follow UCPE (Zhang et al., 2026) to evaluate camera-to-world trajectories. Estimated and target trajectories are each aligned to their own initial frame, and each translation sequence is divided by its maximum displacement norm plus 10^{-9}. RotErr sums geodesic rotation angles over the selected frames; TransErr sums translation-vector distances; CamMC sums Euclidean distances between the flattened first three rows of the relative poses. Reported results average these per-video sums over the test set.

Rotation-only encoding ablation. The rotation-only variant replaces the six-dimensional line action with $\mathrm{diag}(\mathbf{R}, \mathbf{R})$. It retains the same trainable parameters, the absolute orientation map, spatial RoPE, and endpoint camera timestamps. We evaluate the trained model on the same 272 PanShot and 100 RealEstate10K conditions as the main comparison.

Camera-baseline groups. We define $b = \max_{0 \le t \le 80} \|\mathbf{c}_t - \mathbf{c}_0\|_2$ from the target camera centers in the dataset's conditioning scale, before the independent trajectory normalization used for TransErr and CamMC. Near-static conditions satisfy $b < 0.05$; this selects 39 PanShot clips and no RealEstate10K clips. We divide the remaining conditions at the linearly interpolated one-third and two-thirds quantiles: 0.94 and 1.40 on PanShot, and 0.80 and 1.52 on RealEstate10K. Using these thresholds gives group sizes of 78/77/78 and 33/33/34, respectively. The thresholds are dataset-specific and are not metric distances. Group membership is determined entirely by the target trajectory and is shared across methods; the same pose metrics are then averaged within each group.

A.1 Depth Estimation

Architecture and positional encodings. For Table 4, all four variants use a single-scale UniMatch (Xu et al., 2023) model with 192 feature channels and six transformer layers. The original UniMatch baseline retains additive 2D sinusoidal positional encoding. PRoPE, UCPE, and EpiPE replace this encoding with camera transformations and a shared spatial 2D RoPE. Of the 192 channels, 96 encode geometry and the remaining 96 encode spatial position, split equally between the two image axes. Camera transformations act on queries, keys, values, and outputs in both transformer self-attention and cross-attention. We use the reference camera as the world frame and form the token transformations from the relative pose and intrinsics at feature resolution.

Training protocol. All variants are trained in full from the same random initialization with the same seed, for 50,000 optimizer updates on two NVIDIA H200 GPUs. The global batch contains 20 image pairs, with 10 pairs per GPU and no gradient accumulation. We use AdamW with peak learning rate 4×10^{-4} and weight decay 10^{-4}, a cosine OneCycle schedule with a 5% warm-up phase, and gradient clipping at norm 1.0. We retain UniMatch's random color augmentation, random resizing, and 448×576 random crops, updating intrinsics consistently with the image transformations. The objective follows UniMatch and is the sum of inverse-depth L1 and inverse-depth gradient losses, each weighted by 20, with prediction weighting factor $\gamma = 0.9$. The data order and training settings are shared across variants.

Evaluation protocol. We evaluate the final 50,000-step checkpoints on all 576 DeMoN test pairs: 160 RGBD, 160 SUN3D, and 256 Scenes11 pairs. Evaluation uses the original 480×640 images and valid ground-truth pixels with depth in $(0.5, 10)$. AbsRel and RMSE are computed per pair and then averaged within each subset, without aligning or rescaling predicted depths to ground truth. Table 4 reports absolute errors for UniMatch and percentage changes relative to that baseline for the other encodings.