RayRoPE: Projective Ray Positional Encoding for Multi-view Attention

Yu Wu[1], Minsik Jeon[1], Jen-Hao Rick Chang[2], Oncel Tuzel[2], and Shubham Tulsiani[1]

[1] Carnegie Mellon University
[2] Apple

Abstract. We study positional encodings for multi-view transformers that process tokens from a set of posed input images, and seek a mechanism that encodes patches uniquely, allows $SE(3)$-invariant attention with multi-frequency similarity, and can adapt to the geometry of the underlying 3D scene. We find that prior (absolute or relative) encoding schemes for multi-view attention do not meet these desiderata, and present RayRoPE to address this gap. RayRoPE represents patch positions based on associated rays and computes query-frame projective coordinates to ensure $SE(3)$ invariance. To adapt to scene geometry, RayRoPE predicts (without direct supervision) a per-token depth to obtain its position along the corresponding ray, while also modeling uncertainty and analytically computing the expected positional encoding. We validate our method on the tasks of novel-view synthesis, stereo depth estimation, and feed-forward 3DGS reconstruction. While remaining efficient, RayRoPE consistently improves over alternate position encoding schemes (*e.g.* 24% relative improvement on LPIPS in RE10K).

Keywords: positional encoding · multi-view attention · transformer

1 Introduction

Vision transformers have become ubiquitous [3,8,31,39] in image processing applications. At their core is a simple but general computation mechanism where image patches are encoded as 'tokens' and update their representations by 'attending' to one another [46]. This attention operation does not natively account for the position of the input tokens (*e.g.* whether a patch lies in the top left or the bottom right), so it is typical to additionally use a positional encoding to impart this information. When the tokens correspond to patches from a single image, pixel space serves as a natural coordinate system, and we can define positional encodings as a function of the 2D patch position. However, such an encoding is not always applicable: for tasks such as novel-view synthesis or multi-view stereo, tokens can originate from *different* images and cannot be associated with a single 2D coordinate system. In this work, we consider *multi-view* vision transformers and ask: *how should we define positional encodings for tokens corresponding to patches from a set of posed images?*

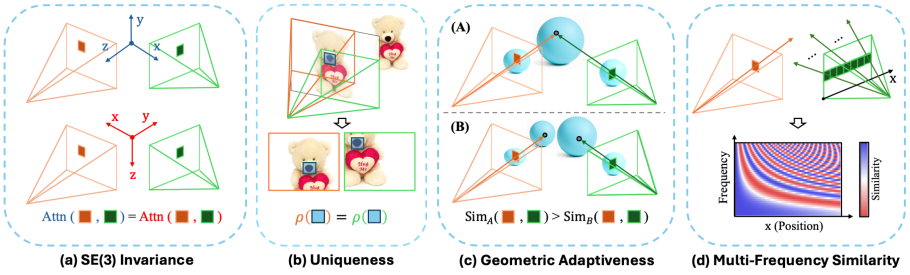

Fig. 1: Desiderata for a Multi-View Positional Encoding. We seek the following properties for positional encoding in multi-view attention: **(a)** The attention output should be invariant to the choice of global coordinate system (i.e., $SE(3)$ invariance). **(b)** The positional encoding of tokens that correspond to the same patch observed across different images should be the same. **(c)** The positional encoding can vary with the underlying scene geometry (*e.g.* yielding higher similarity when patches observe a common 3D point than when they do not). **(d)** Analogous to common 1D and 2D encodings, aspects of the positional encoding should vary at different frequencies, enabling multi-frequency similarity. RayRoPEsatisfies (a–d) by encoding ray segments projected into each query camera frame.

We seek to formulate a mechanism with the following properties: ***(a)*** $SE(3)$ ***invariance***: the attention operation should only depend on *relative* cameras and not a global coordinate system, ***(b) uniqueness***: if the same patch is viewed across different images (*e.g.* overlapping crops of a panorama), the position of the corresponding tokens should be the same, ***(c) geometry-adaptiveness***: position encodings can vary with underlying scene geometry *e.g.* if different patches 'see' the same corresponding 3D point, their positions should be more similar than if the same patches see different 3D points, and ***(d) multi-frequency similarity***: as typical for 1D and 2D position encodings [23], a multi-view position encoding should also compute similarity in attention at various 'frequencies'. While we are certainly not the first to consider the question of designing position encodings for multi-view transformers [1, 9, 26], current formulations [20, 22, 28] do not meet the above desiderata. In particular, these methods define position encodings via image coordinates and camera matrices, and while they ensure $SE(3)$ invariance, they do not represent patches uniquely, encode positions independently of (known or estimated) scene geometry, and do not leverage the camera information in a 'multi-frequency' manner.

We present a formulation, RayRoPE, that leverages the ray(s) corresponding to each patch to define positional embeddings for multi-view transformers. While this directly yields unique, multi-frequency encodings, these are not $SE(3)$ invariant and make attention depend on an arbitrary global coordinate system. RayRoPE overcomes this by instead using *relative* ray positions for attention, transforming them into the (projective) coordinates of the query token's camera. Additionally, instead of parameterizing rays with their origin and direction (equivalently, a point at infinity), RayRoPE replaces the latter by letting each

token *predict* (without direct supervision) a point along the ray to encode its position, allowing the embedding to vary with the underlying geometry. We also formulate a mechanism to efficiently incorporate uncertainty in predicted depths along a ray, yielding an analytical solution to compute the expected positional encoding across frequencies. Our formulation can be implemented efficiently, with runtime comparable to existing positional encoding methods.

We validate RayRoPE on a variety of 3D vision tasks, including novel-view synthesis [16, 20], stereo depth estimation [50], and feed-forward 3D Gaussian Splatting [17] reconstruction [35]. We integrate our positional encoding scheme into prior methods that leverage multi-view transformers for various tasks, and show that RayRoPE outperforms all prior (relative and absolute) positional encodings. We then perform extensive ablations and analysis on RayRoPE, verifying the importance of the above desiderata in our method and showing strong generalization beyond training views and datasets. We also show that RayRoPE's 'depth prediction' in the attention mechanism allows geometry to emerge without direct supervision.

Our core contributions can be summarized as follows:

- We propose RayRoPE a first multi-view positional encoding satisfying the desiderata above: uniqueness, invariance, geometric adaptiveness, and multi-frequency encodings.
- Our formulation enables tokens to determine their own positions via (emergent) depth and extends RoPE to model uncertainty in these positions.
- We provide an efficient and general implementation of RayRoPE which can be easily integrated into generic multi-view models, improving performance over prior positional encoding schemes.

2 Related Work

Positional Encodings in Transformers. The attention mechanism is inherently permutation-invariant, treating its input tokens as an unordered set [46]. To recover structural information, transformers require explicit positional encodings. Early approaches [3, 7, 8, 18, 31, 36] relied on absolute positional encodings (APE), which add or concatenate fixed or learned positional embeddings to token representations. More recently, the community shifted toward relative positional encodings (RPE), which model relative positions between token pairs by modifying the attention mechanism. This includes additive attention biases [30, 32, 38] and rotary positional encodings (RoPE) [41], which rotate query and key vectors in a position-dependent manner. RoPE confers translational invariance to transformers and has progressively become the standard positional encoding approach adopted in leading models across diverse domains [11, 12, 19, 25, 32, 37, 44]. We propose a RoPE-based positional encoding tailored for multi-view transformers, designed to encode spatial relationships in a manner invariant to arbitrary rigid transformations of the coordinate system.
Multi-view Transformers. Multi-view transformers have emerged as a powerful framework driving progress across a wide range of 3D vision tasks, such

as novel-view synthesis [1, 9, 13, 16, 42, 53, 54] and 3D scene understanding [10, 14, 15, 52]. These models take as input a set of posed images, where the key challenge is to encode spatial relationships among image patches across views. Existing approaches typically represent camera information as rays [9, 10, 16, 54] or camera matrices [26, 52] and concatenate them with input features, analogous to absolute positional encodings (APE). However, these encodings rely on an arbitrary global coordinate system; our method instead ensures $SE(3)$ invariance via relative encoding.

Relative Positional Encodings in 3D. Closest to our work are relative positional encodings based on camera poses [20, 22, 28, 48]. Designed for multi-view transformers, these methods transform attention features using camera pose matrices and ensure invariance to the choice of global coordinate system (see Sec. 3.2). Recent work [21, 27, 47, 48] also demonstrates that such pose-based relative encodings can be applied to video generation models for camera control. Despite their effectiveness, they cannot adapt to scene geometry or support multi-frequency similarity as in standard RoPE. While some methods [22, 28] alleviate this by incorporating standard RoPE on patch indices, such a design breaks uniqueness and is not geometrically grounded across views. In comparison, we propose a ray-based relative positional encoding that naturally supports frequency modulation, ensures uniqueness, and can adapt to the geometry of the underlying 3D scene being observed. Concurrent work [2] augments standard RoPE with depth, but its generalizability to settings with more than two views is not explored.

3 Preliminaries: Positional Encodings

We consider the attention mechanism operating on a set of token features $\{\boldsymbol{\tau}_i\}_{i=1}^{L}$. We use $\mathbf{q}_i, \mathbf{k}_i, \mathbf{v}_i \in \mathbb{R}^D$ to denote the corresponding query, key, and value feature of token $\boldsymbol{\tau}_i$. Consider a query token $\boldsymbol{\tau}_i$ that attends to a set of tokens $\{\boldsymbol{\tau}_j\}$, the output of attention on token $\boldsymbol{\tau}_i$ can be written as:

$$\mathbf{o}_i = \mathrm{Attn}(\mathbf{q}_i, \{\mathbf{k}_j, \mathbf{v}_j\}) = \frac{\sum_j \exp(\mathbf{q}_i^\intercal \mathbf{k}_j)\mathbf{v}_j}{\sum_j \exp(\mathbf{q}_i^\intercal \mathbf{k}_j)} \tag{1}$$

The pairwise nature of attention allows the design of relative positional encodings that provide certain invariance properties. In the following sections, we review the standard Rotary Position Embeddings (RoPE) (Sec. 3.1) for translational invariance and recent camera-based relative positional encodings (Sec. 3.2) for $SE(3)$ invariance. We then analyze the limitations of these approaches (Sec. 3.3).

3.1 Rotary Positional Encoding

RoPE was initially proposed to encode the 1D positions x_i in language models by transforming features with a series of SO(2) rotations at various frequencies. We use $\oplus$ to denote matrix concatenation along the diagonal. RoPE encodes a position x as a $D \times D$ matrix:

$$\rho_D(x) \equiv \oplus_{f=1}^{D/2} \rho_2(\omega_f x) \equiv \oplus_{f=1}^{D/2} e^{i\omega_f x} \tag{2}$$

where we denote 2×2 rotational matrices with $e^{i\omega x}$ for notational simplicity. $\{\omega_f\}_{f=1}^{D/2}$ is a set of predefined rotational frequencies. RoPE encoding is used to transform query and key features, yielding $\rho(x_i)\mathbf{q}_i, \rho(x_j)\mathbf{k}_j$. The resulting attention score between any two tokens only depends on the relative position $x_j - x_i$, making the attention invariant to translations of positions. By rotating different channels of features at various frequencies, the model can learn to reason about the positional information at different scales (see Fig. 1 (d)). For a general position $\mathbf{x} \in \mathbb{R}^C$ that is C-dimensional, it is straightforward to extend RoPE:

$$\rho_D(\mathbf{x}) \equiv \oplus_{f=1}^{D/2C} \oplus_{c=1}^{C} e^{i\omega_f x_c} \tag{3}$$

This is commonly used for the pixel indices (u, v) in vision transformers [3,12,33, 39] and the three-dimensional positions (u, v, t) in video transformers [19,51]. For multi-view transformers, while it is possible to directly apply RoPE to positions in 3D (such as ray maps), the relative positions will depend on the rotation of the global coordinate frame, leading to suboptimal performance (see Sec. 5.1).

3.2 Camera-Based Relative Positional Encoding

Instead of using SO(2) encodings, **CaPE** [20] encodes cameras using their extrinsics $T \in \mathbb{R}^{4\times4}$. The corresponding $D \times D$ encoding is computed by repeating T along the diagonal, denoted as $E_i^{\text{cape}} = \oplus_{n=1}^{D/4} T_i$. The resulting attention score only conditions on the relative poses $T_i T_j^{-1}$, which is independent of the choice of global coordinate frame, making the transformer model $SE(3)$ invariant. However, the camera-based encoding used in CaPE cannot support multi-frequency similarities, limiting the model's ability to reason about high-frequency details. **GTA** [28] compensates for this by incorporating standard RoPE on image patch indices. A patch position is represented by $\mathbf{x} = (T, u, v)$, and the corresponding encoding is a concatenation of CaPE and standard 2D RoPE:

$$E_i^{\text{gta}} = E^{\text{gta}}(\mathbf{x}_i) \equiv (\oplus_{n=1}^{D/8} T_i) \oplus \rho_{\frac{D}{2}}(u_i, v_i) \tag{4}$$

In addition to applying encodings to query and key features, GTA also applies encodings to value and output features. **PRoPE** [22] further improves GTA by replacing the camera extrinsics T with the full projection matrix $P = KT \in \mathbb{R}^{4\times4}$ (where the intrinsics K are lifted to 4×4). This enables the model to also reason about the camera intrinsics.

3.3 Limitations of Existing Methods

In Fig. 1, we illustrate four desirable properties for positional encodings in multi-view transformers: **(1)** $SE(3)$ invariance, **(2)** uniqueness, **(3)** geometry-adaptiveness, and **(4)** multi-frequency similarity. While standard RoPE supports multi-frequency encoding, it is not $SE(3)$ invariant. The camera-based

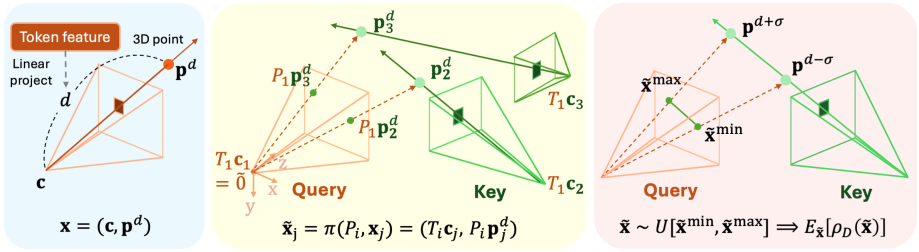

Fig. 2: Overview of RayRoPE. (a) We encode image patch position as a ray segment $\mathbf{x} = (\mathbf{c}, \mathbf{p}^d)$, where $\mathbf{c}$ is the camera center and $\mathbf{p}^d$ is the point at depth d along the ray r. We use a linear layer to allow each token to predict the depth d along the ray, thus enabling RayRoPE to adapt to the scene geometry. **(b)** To ensure $SE(3)$ invariance, we compute the positional encodings using ray positions projected to the query camera frame with $P_i = K_i T_i$, yielding $\tilde{\mathbf{x}}_j = \pi(P_i, \mathbf{x}_j)$. **(c)** To model the uncertainty in depth prediction, we also predict an uncertainty σ, yielding an estimated range between $\mathbf{p}^{d-\sigma}$ and $\mathbf{p}^{d+\sigma}$, and use an analytically computed expected position encoding for the corresponding token.

relative encodings [20,22,28] satisfy $SE(3)$ invariance, but cannot perform multi-frequency encodings. GTA and PRoPE compensate for this by including standard 2D RoPE based on patch indices (u, v). This hybrid design, however, still does not incorporate frequencies for the camera and also breaks the uniqueness property. As illustrated in Fig. 1 (d), if the same image patch is viewed across different overlapping images, its patch indices (u, v) will change, resulting in a different positional encoding. Furthermore, these prior methods lack explicit geometry adaptiveness, and the computed encodings cannot vary with the underlying 3D scene structure (Fig. 1 (c)).

4 RayRoPE

We propose RayRoPE, a relative positional encoding that satisfies all four desiderata introduced previously for multi-view attention. We begin by defining a ray-based position representation (Sec. 4.1) and formulate relative encoding via projection onto the query frame (Sec. 4.2). We then introduce expected RoPE encoding to address depth ambiguity (Sec. 4.3) and finally show that RayRoPE can be implemented efficiently, with runtime comparable to baselines (Sec. 4.4).

4.1 Image Patch as Ray Segments

A common way to represent a patch position in 3D is to use the ray starting from the camera center and passing through the patch center, parameterized by the camera center $\mathbf{c}$ and ray direction $\mathbf{r}$. This representation inherently satisfies the uniqueness property. However, it cannot adapt to the geometry of the observed scene, as it is unaware of the depth to the 3D point being intersected by the ray.

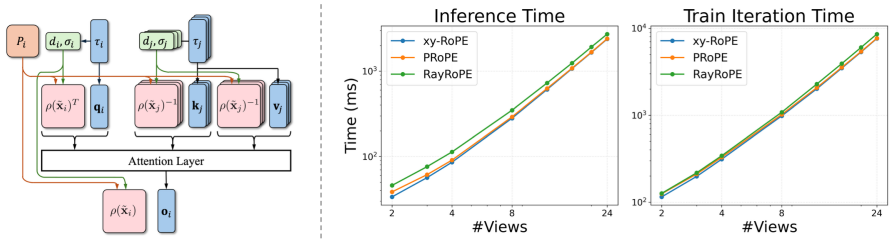

Fig. 3: Left: Applying RayRoPE to attention. We predict per-token depths and uncertainties via a linear projection from features τ, which are used to compute ray segments. We project the rays with query camera P_i, and the computed encodings are applied to $\mathbf{q}, \mathbf{k}, \mathbf{v}$ and $\mathbf{o}$ features. **Right: Efficiency Analysis.** We compare the runtime of different methods. While RayRoPE computes additional encodings and depths, this only results in a marginal overhead (13% at inference and 4% at training relative to PRoPE for N=3, with a similar trend for larger N).

To address this limitation, we define our position representation as a ray segment in homogeneous coordinates $\mathbf{x} = (\mathbf{c}, \mathbf{p}^d)$, where $\mathbf{p}^d \in \mathbb{P}^3$ is the point where the ray intersects the 3D scene at depth d. The ray direction $\mathbf{r}$ is equivalent to $\mathbf{p}^\infty$ (via setting homogeneous term to 0). When a depth map is provided as input, the ray-segment representation can naturally incorporate this information. In the more common cases where scene geometry is unknown, we make the model estimate the depth d of the 3D point intersected by the ray (see Fig. 2 (a)). Specifically, we add a single linear layer to each attention layer to predict the depth d. The depth is estimated per-layer; each attention layer now projects input features $\{\tau_i\}$ to compute a depth for each token, which is used to compute the ray encodings. These layers are jointly learned with the model *without any additional supervision*.

4.2 Relative Encodings in Query Frame

The ray representation defined above is in the global coordinate frame. To ensure $SE(3)$ invariance, we compute RayRoPE encodings in each query token's local frame, as illustrated in Fig. 2(b). We first define a projective operator that projects global rays into the query camera frame, and then apply standard RoPE to the projected rays in the query frame to support both invariance and multi-frequency similarity.

Projection onto Query Camera. Given the query camera matrix $P_i = K_i T_i \in \mathbb{R}^{4\times4}$ (assuming K_i, T_i are lifted to 4×4) and an arbitrary ray $(\mathbf{c}, \mathbf{p}^d)$, we define the projected ray as:

$$\tilde{\mathbf{x}} = \pi(P_i, \mathbf{x}) = (T_i\mathbf{c}, P_i\mathbf{p}^d) \tag{5}$$

$T_i\mathbf{c} = [c_x, c_y, c_z, 1]^\intercal$ represents the 3D camera center transformed into the query frame. $P_i\mathbf{p}^d = [u, v, 1, 1/d']^\intercal$ is the projection of the 3D point $\mathbf{p}^d$ onto query camera, where (u, v) is the projected pixel coordinate and $1/d'$ is the disparity. Given the transformed (and normalized) homogeneous coordinates, we compactly rep-

resent each projected ray $\tilde{\mathbf{x}}$ as a 6D vector: $[c_x, c_y, c_z, u, v, 1/d']$, which will be used to compute the RoPE encodings.

Encodings. Given the projected rays, we apply RoPE (Eq. 3) with multi-frequency similarity. For a fixed query camera P_i, we define the RayRoPE encoding as $\rho_D(\tilde{\mathbf{x}}) = \rho_D(\pi(P_i, \mathbf{x}))$. Similar to GTA, we apply our encodings to query, key, value, and output features (see Fig. 3):

$$\begin{aligned} &\text{Attn}^{\text{ours}}(\mathbf{q}_i, \{\mathbf{k}_j, \mathbf{v}_j\}) = \\ &\rho_D(\tilde{\mathbf{x}}_i)\text{Attn}(\rho_D(\tilde{\mathbf{x}}_i)^\intercal \mathbf{q}_i, \{\rho_D(\tilde{\mathbf{x}}_j)^{-1}\mathbf{k}_j, \rho_D(\tilde{\mathbf{x}}_j)^{-1}\mathbf{v}_j\}) \end{aligned} \tag{6}$$

We can show that the above attention expands to the form:

$$\frac{\sum_j \exp(q_i^\intercal \rho_D(\tilde{\mathbf{x}}_i - \tilde{\mathbf{x}}_j)k_j)\ \rho_D(\tilde{\mathbf{x}}_i - \tilde{\mathbf{x}}_j)v_j}{\sum_j \exp(q_i^\intercal \rho_D(\tilde{\mathbf{x}}_i - \tilde{\mathbf{x}}_j)k_j)} \tag{7}$$

which depends only on the relative positions $\tilde{\mathbf{x}}_i - \tilde{\mathbf{x}}_j$ in the query frame, ensuring $SE(3)$ invariance.

4.3 Modeling Uncertainty via Expected RoPE

RayRoPE position representation relies on the predicted depth along the rays (Sec. 4.1). This prediction is only an approximation, and can lead to noisy RoPE encoding, especially for the components where the frequency ω is high. To alleviate this issue, we predict an uncertainty value σ along with each depth d. As shown in Fig. 2 (c), this yields a ray segment bounded by two points: $\mathbf{x}^{\min} = (\mathbf{c}, \mathbf{p}^{d-\sigma})$, $\mathbf{x}^{\max} = (\mathbf{c}, \mathbf{p}^{d+\sigma})$. Instead of using RoPE on a single position, we propose an expected RoPE encoding $\mathbb{E}_{\tilde{\mathbf{x}}}[\rho_D(\tilde{\mathbf{x}})]$, over the distribution of $\tilde{\mathbf{x}}$.

$$\mathbb{E}_{\tilde{\mathbf{x}}}[\rho_D(\tilde{\mathbf{x}})] = \oplus_{f=1}^{D/2C} \oplus_{c=1}^{C} \mathbb{E}_{x_c}[e^{i\omega_f x_c}] \tag{8}$$

We assume the projected position $\tilde{\mathbf{x}}$ follows a uniform distribution ranging from $\tilde{\mathbf{x}}^{\min}$ to $\tilde{\mathbf{x}}^{\max}$, i.e., for each component of the ray, $x_c \sim U(x^{\min}, x^{\max})$. We can analytically compute its expected RoPE encoding using the equation below (see Appendix for details):

$$\mathbb{E}_{x_c}[e^{i\omega x_c}] = \frac{\int_{x^{\min}}^{x^{\max}} e^{i\omega x_c} dx_c}{x^{\max} - x^{\min}} = \frac{e^{i\omega x^{\max}} - e^{i\omega x^{\min}}}{i\omega(x^{\max} - x^{\min})} \tag{9}$$

For deterministic components (e.g., camera centers) where $x^{\min} = x^{\max}$, the expectation reduces to regular RoPE. For positions with larger uncertainty, the expected rotation is 'smoothed out' as $x^{\max} - x^{\min}$ increases. The expected RoPE naturally maintains the relative positions when multiplied in attention (assuming positions are independent):

$$\mathbb{E}_{\tilde{\mathbf{x}}_i}[\rho_D(\tilde{\mathbf{x}}_i)]\mathbb{E}_{\tilde{\mathbf{x}}_j}[\rho_D(\tilde{\mathbf{x}}_j)]^{-1} = \mathbb{E}_{\tilde{\mathbf{x}}_i, \tilde{\mathbf{x}}_j}[\rho_D(\tilde{\mathbf{x}}_i - \tilde{\mathbf{x}}_j)] \tag{10}$$

By modeling uncertainty with expected RoPE, we can produce stable yet geometrically-aware encodings even when predicted depths are approximate.

Method	CO3D [34]			Objaverse [6]			RE10K [55]		
	PSNR↑	LPIPS↓	SSIM↑	PSNR↑	LPIPS↓	SSIM↑	PSNR↑	LPIPS↓	SSIM↑
Plucker raymap [16]	16.54	0.673	0.538	19.98	0.273	0.844	23.45	0.141	0.747
RoPE on rays	16.85	0.589	0.549	22.17	0.117	0.902	25.29	0.095	0.809
GTA [28]	16.50	0.602	0.544	21.87	0.134	0.891	24.31	0.124	0.774
PRoPE [22]	17.49	0.539	0.563	22.16	0.123	0.896	24.48	0.112	0.784
RayRoPE	**18.40**	**0.461**	**0.592**	**22.42**	**0.110**	**0.905**	**26.07**	**0.085**	**0.831**
PRoPE (Large)	19.77	0.329	0.631	24.66	0.067	0.925	27.77	0.059	0.866
RayRoPE(Large)	**20.23**	**0.308**	**0.662**	**24.83**	**0.064**	**0.929**	**28.31**	**0.055**	**0.876**

Table 1: Comparison on novel-view synthesis. When incorporated in LVSM [16], RayRoPE outperforms all baseline positional encoding methods across three datasets. The bottom two rows show results from models trained at increased scale.

4.4 Applying RayRoPE Efficiently

While the proposed encodings depends on the query camera, we can efficiently implement it via grouping the attention computation by query view. For each query view, we first compute the corresponding encodings, then perform attention between the query tokens in the current view and all key/value tokens. Because in modern transformers, attention is almost always bottlenecked by the total number of tokens—which remains unchanged—this grouping adds negligible overhead (see Appendix for details).

We benchmark the runtime of RayRoPE against regular 2D RoPE [12] and PRoPE [22]. We use the same LVSM setting as in Sec. 5.1, and summarize results in Fig. 3. Despite the query-dependent encodings and depth prediction, RayRoPE's efficiency remains *highly comparable* to the baseline methods, across various number of views N. For example, with $N = 3$ (which is the default setting in Sec. 5.1), RayRoPE results in a marginal 4% overhead at training and 13% at inference when compared to PRoPE. Notably, RayRoPE is also compatible with acceleration methods such as KV-caching [29] and FlashAttention [5].

5 Experiments

We first verify the effectiveness of RayRoPE on novel-view synthesis (NVS), benchmarking it against state-of-the-art positional encodings within the LVSM [16] framework (Sec. 5.1). We then ablate the design choices of RayRoPE and analyze its internal behavior (Sec. 5.2). Finally, we show that RayRoPE extends well beyond this single setting, improving a diverse range of 3D vision tasks and model architectures (Sec. 5.3): stereo depth estimation, feed-forward 3DGS reconstruction, multi-view diffusion.

5.1 Novel-View Synthesis

Setup. We verify RayRoPE on novel-view synthesis (NVS) by applying it to LVSM [16], a state-of-the-art view synthesis model. Each model takes two posed reference views and the target camera pose, and performs self-attention across all

Method	CO3D [34]			Objaverse [6]			RE10K [55]		
	PSNR↑	LPIPS↓	SSIM↑	PSNR↑	LPIPS↓	SSIM↑	PSNR↑	LPIPS↓	SSIM↑
RayRoPE	18.40	0.461	0.592	22.42	0.110	0.905	26.07	0.085	0.831
① w/o uncertainties	17.28	0.594	0.560	20.37	0.175	0.885	25.84	0.088	0.826
② w/o geo-adaptiveness	17.06	0.553	0.557	22.42	0.111	0.904	25.36	0.093	0.812
③ w/o multi-frequency	17.49	0.550	0.563	21.37	0.223	0.862	24.27	0.115	0.779
④ w/o $\mathbf{v}, \mathbf{o}$ encoding	18.00	0.510	0.577	21.72	0.127	0.898	25.56	0.091	0.819

Table 2: Ablations on RayRoPE. The performance drops of **Variant** ①, ②, ③ highlights the importance of uncertainty modeling, geometric adaptiveness, and multi-frequency similarities, respectively.

views. Following PRoPE [22], we train downsized LVSM variants (∼47M parameters) from scratch with different positional encoding methods. To demonstrate scalability, we also train larger (∼150M parameters) variants with an 8× batch size. We train models separately on three datasets: CO3D [34], Objaverse [6], and RealEstate10K (RE10K) [55], and measure reconstruction quality against ground-truth views. For Objaverse, we render a high-quality 80K subset [43] with diverse intrinsics. Except for the large-scale experiments, we train all models with three random seeds and report mean results.

Baselines. We compare RayRoPE against several baseline positional encoding methods. Plucker raymap is the original implementation in LVSM, which concatenates 6D Plucker raymaps to the input tokens. We include the best existing camera-based relative positional encodings: GTA [28] and PRoPE [22], both of which are $SE(3)$ invariant. We also design a RoPE-on-rays baseline, which naively applies RoPE to the raymap $(\mathbf{c}, \mathbf{r})$ in global coordinates (as discussed in Sec. 3.1). Following [22], we concatenate the "CamRay" intrinsics raymap to the input for both PRoPE and RayRoPE.

Results. As shown in Table 1, RayRoPE consistently outperforms all baselines across the three datasets at both small and large scales. Overall, relative encodings (GTA, PRoPE, RayRoPE) tend to outperform the absolute Plucker raymap. While it works well on RE10K and Objaverse, the naive RoPE-on-rays baseline struggles on the more challenging CO3D. This highlights the importance of $SE(3)$ invariance for reasoning about complex geometric relationships.

In Fig. 4, we visualize several example scenes from each evaluation dataset. For target views that overlap significantly with the reference views, RayRoPE generates superior high-frequency details compared to baselines. For more challenging target views with little overlap with the reference views, while all methods exhibit a reduction in sharpness, RayRoPE is nonetheless able to generate more coherent novel views.

5.2 Ablations and Analysis

In this section, we first ablate the key design choices of RayRoPE, verifying the importance of the four desiderata in Fig. 1. We then analyze RayRoPE's behavior by focusing on the emergent depth and uncertainty.

Ablations. In Table 2, we ablate the design choices of RayRoPEbased on the LVSM experiments in Sec. 5.1. Keeping all other configurations the same, we

Fig. 4: Qualitative examples on novel-view synthesis. RayRoPE synthesizes more 3D-consistent views with sharper details.

compare the full RayRoPE to several variants. **Variant** ①: Remove the uncertainty prediction σ (thus disabling expected RoPE). **Variant** ②: Remove the depth prediction d (thus disabling geometric adaptiveness), using ray direction $\mathbf{p}^{\infty}$ instead of $\mathbf{p}^{d}$. **Variant** ③: Remove the multi-frequency encoding. **Variant** ④: Remove the encoding on the value and output features $(\mathbf{v}, \mathbf{o})$, applying it only to query and key features.

All four variants degrade performance to varying extents. Variant ① shows that uncertainty modeling via expected RoPE is crucial for robust performance, especially when pose variations are larger (CO3D and Objaverse). Variant ② highlights the importance of geometric adaptiveness via depth prediction, while Variant ③ highlights the benefits of multi-frequency encodings. Variant ④ shows that applying the encodings to value and output features is helpful, consistent with prior work [28].

Analysis of Emergent Depth. We analyze the predicted depth d (Sec. 4.1) and uncertainty σ (Sec. 4.3) and show that they encode meaningful geometric information. We measure the error of predicted depths on CO3D scenes and plot it against the predicted uncertainties (left panel of Fig. 5). Each point corresponds to one image at a specific layer. The model exhibits high uncertainty

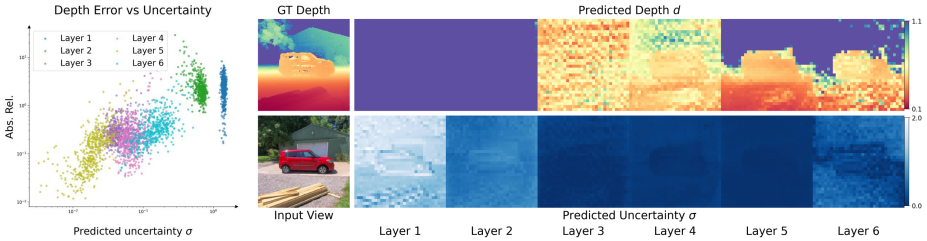

Fig. 5: *Left*: Error on predicted depths vs predicted uncertainties. In deeper layers (Layers 5–6), the depth errors and uncertainties are strongly positively correlated, showing that the model predicts higher uncertainty when confidence is low. ***Right*: Predicted depths and uncertainties across layers**. The predicted depth at deeper layers aligns well with the ground-truth depth. The predicted σ gradually decreases across layers, showing that confidence in depth prediction gradually improves.

in the early layers, consistent with ambiguity at the beginning of processing. As features propagate through the network (Layers 5–6), the uncertainty decreases and shows a strong positive correlation with depth error, indicating that higher uncertainty is assigned to less reliable predictions. RayRoPE subsequently incorporates such uncertainty by computing the expected RoPE. The right panel of Fig. 5 visualizes the evolution of the predicted depth and uncertainty maps across all six layers on a representative scene. By Layer 5, geometrically plausible depth maps emerge despite the absence of depth supervision. This result supports the conclusion that RayRoPE leverages depth predictions in a geometrically meaningful manner. See Appendix for more visual examples.

5.3 Broader Applications

We have evaluated and analyzed RayRoPE on the LVSM framework for novel-view synthesis. In this section, we show that RayRoPE can serve as a plug-and-play mechanism that is generally applicable to any multi-view attention module with camera conditioning. Across stereo depth estimation [50], feed-forward 3DGS reconstruction [35], and multi-view diffusion [20], RayRoPE consistently improves over the respective baselines.

Stereo Depth Estimation We adopt UniMatch [50], a multi-view transformer trained for multiple tasks, including stereo depth estimation. The model takes stereo image pairs with known poses. We apply RayRoPE to the cross-image attention layers of UniMatch. Following the original work, we train and evaluate the model on RGBD [40], SUN3D [49], and Scenes11 [45]. To evaluate out-of-distribution performance, we also benchmark on an unseen dataset, ScanNet [4]. As shown in Tab. 3, compared to both the original UniMatch and the variant with PRoPE, the model with RayRoPE achieves more accurate depth predictions. Qualitative results in Fig. 6 further illustrate improved geometric consistency, showing that RayRoPE yields more accurate depth structures.

Feed-forward 3D Gaussian Splatting Reconstruction. We also experiment with TokenGS [35], a recent feed-forward 3DGS [17] reconstruction model.

Method	RGBD [40]		SUN3D [49]		Scenes11 [45]		ScanNet [4](Unseen)	
	Abs Rel↓	RMSE↓	Abs Rel↓	RMSE↓	Abs Rel↓	RMSE↓	Abs Rel↓	RMSE↓
UniMatch [50]	0.119	0.628	0.135	0.406	0.086	0.742	0.127	0.346
+PRoPE [22]	**0.106**	0.589	0.113	0.338	0.051	0.494	0.101	0.285
+RayRoPE	**0.106**	**0.574**	**0.109**	**0.328**	**0.047**	**0.473**	**0.095**	**0.276**

Table 3: Comparisons on stereo depth estimation. We report absolute relative error (Abs. Rel.) and root mean square error (RMSE) across four datasets, including the out-of-distribution ScanNet [4]. The UniMatch [50] model with RayRoPE encoding predicts more accurate depths than the baselines.

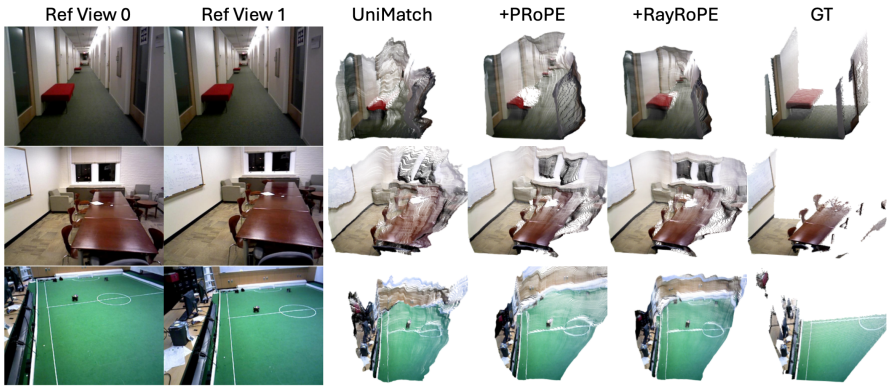

Fig. 6: Example 3D points reprojected from stereo depth estimation. RayRoPE produces more accurate depth predictions and thus better 3D reconstruction.

TokenGS first encodes posed multi-view inputs into multi-view features and then decodes a set of scene-level 3DGS tokens into 3DGS parameters by cross-attending to the multi-view features. We apply RayRoPE to the encoder self-attention and the image-to-3DGS cross-attention. In the cross-attention, we apply RayRoPE only to the image features and skip ray projection, since the 3DGS query tokens do not have explicit positions. Following the original work's settings, we train on DL3DV-10K [24] with four input views and evaluate the rendering quality of the predicted 3DGS. As shown in Table 4, we find that RayRoPE outperforms both the original method and the PRoPE variant. These results suggest that our method generalizes across architectures (e.g., cross-attention between image features and implicit latent tokens).

Multi-view diffusion model. We adopt EscherNet [20], a large camera-controlled multi-view diffusion model. We retrain EscherNet variants on DL3DV-10K [24] with CaPE (the original design), PRoPE, and RayRoPE. As shown in Table 5, RayRoPE remains advantageous, with the performance gap widening as the number of reference views increases. These results also indicate that RayRoPE's depth predictions remain robust even on the noisy images in diffusion models.

Method	2 views			4 views			6 views		
	PSNR↑	LPIPS↓	SSIM↑	PSNR↑	LPIPS↓	SSIM↑	PSNR↑	LPIPS↓	SSIM↑
TokenGS [35]	18.84	0.467	0.584	22.85	0.363	0.732	23.83	0.343	0.760
+PRoPE [22]	19.02	0.461	**0.598**	23.04	0.361	0.741	24.09	0.342	0.770
+RayRoPE	**19.15**	**0.457**	**0.598**	**23.40**	**0.339**	**0.758**	**24.63**	**0.312**	**0.793**

Table 4: Comparison on feed-forward 3DGS reconstruction. We adapt TokenGS [35], a feed-forward model that predict 3DGS from posed input views. We benchmark the rendering quality of the output 3D Gaussian splats given 2, 4, and 6 input views. RayRoPE helps the model to predict better Gaussian splats.

Method	3 views			6 views			9 views		
	PSNR↑	LPIPS↓	SSIM↑	PSNR↑	LPIPS↓	SSIM↑	PSNR↑	LPIPS↓	SSIM↑
EscherNet [20]	15.81	0.279	0.373	16.31	0.246	0.391	16.42	0.236	0.393
+PRoPE [22]	16.87	0.254	0.434	17.85	0.211	0.480	18.36	0.189	0.504
+RayRoPE	**17.14**	**0.239**	**0.464**	**18.16**	**0.209**	**0.504**	**19.03**	**0.188**	**0.543**

Table 5: Comparison on Multi-view diffusion. We apply RayRoPE to EscherNet [20], a large multi-view diffusion model for view synthesis. On DL3DV-10K [24] with 3, 6, or 9 reference views, RayRoPE consistently outperforms both baselines.

6 Discussion

In this work, we identified four desirable properties for positional encodings in multi-view attention (Fig. 1). Guided by these criteria, we introduced RayRoPE, a ray-based positional encoding that processes a set of posed input images. RayRoPE consistently improves performance on various tasks and models in 3D vision, showing that satisfying these properties yields more effective multi-view attention. While RayRoPE could incorporate uncertainty when predicting depth along a ray, it would also be interesting to model uncertainty in the camera matrices. More broadly, while this work focused on positional embeddings for *posed* input images, designing such encodings for multi-view transformers that process unposed (or mixed) images remains an open challenge.

Acknowledgment

We would like to thank Zihan Wang and Qitao Zhao for insightful discussions about the project. This project was supported by Apple and NSF Award IIS2345610. This work used computation resources at NCSA Delta, NCSA Delta-AI, and Bridges-2 through allocation CIS240289, CIS251333, and CIS240132 from the Advanced Cyberinfrastructure Coordination Ecosystem: Services & Support (ACCESS) program, which is supported by U.S. National Science Foundation grants #2138259, #2138286, #2138307, #2137603, and #2138296.

References

1. Bahmani, S., Skorokhodov, I., Qian, G., Siarohin, A., Menapace, W., Tagliasacchi, A., Lindell, D.B., Tulyakov, S.: Ac3d: Analyzing and improving 3d camera control in video diffusion transformers. In: CVPR (2025)
2. Bai, Y., Li, H., Huang, Q.: Positional encoding field. arXiv preprint arXiv:2510.20385 (2025)
3. Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., Joulin, A.: Emerging properties in self-supervised vision transformers. In: ICCV (2021)
4. Dai, A., Chang, A.X., Savva, M., Halber, M., Funkhouser, T., Nießner, M.: Scannet: Richly-annotated 3d reconstructions of indoor scenes. In: CVPR (2017)
5. Dao, T., Fu, D., Ermon, S., Rudra, A., Ré, C.: Flashattention: Fast and memory-efficient exact attention with io-awareness. Neurips (2022)
6. Deitke, M., Schwenk, D., Salvador, J., Weihs, L., Michel, O., VanderBilt, E., Schmidt, L., Ehsani, K., Kembhavi, A., Farhadi, A.: Objaverse: A universe of annotated 3d objects. In: CVPR (2023)
7. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. In: NAACL-HLT (2019)
8. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An image is worth 16x16 words: Transformers for image recognition at scale. In: ICLR (2021)
9. Gao, R., Hołyński, A., Henzler, P., Brussee, A., Martin-Brualla, R., Srinivasan, P., Barron, J.T., Poole, B.: Cat3d: create anything in 3d with multi-view diffusion models. In: NeurIPS (2024)
10. Guizilini, V., Vasiljevic, I., Chen, D., Ambruș, R., Gaidon, A.: Towards zero-shot scale-aware monocular depth estimation. In: ICCV (2023)
11. Guo, D., Yang, D., Zhang, H., Song, J., Zhang, R., Xu, R., Zhu, Q., Ma, S., Wang, P., Bi, X., et al.: Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. arXiv preprint arXiv:2501.12948 (2025)
12. Heo, B., Park, S., Han, D., Yun, S.: Rotary position embedding for vision transformer. In: ECCV (2024)
13. Hong, Y., Zhang, K., Gu, J., Bi, S., Zhou, Y., Liu, D., Liu, F., Sunkavalli, K., Bui, T., Tan, H.: Lrm: Large reconstruction model for single image to 3d. In: ICLR (2024)
14. Hong, Y., Lin, C., Du, Y., Chen, Z., Tenenbaum, J.B., Gan, C.: 3d concept learning and reasoning from multi-view images. In: CVPR (2023)
15. Jain, A., Katara, P., Gkanatsios, N., Harley, A.W., Sarch, G., Aggarwal, K., Chaudhary, V., Fragkiadaki, K.: Odin: a single model for 2d and 3d segmentation. In: CVPR (2024)
16. Jin, H., Jiang, H., Tan, H., Zhang, K., Bi, S., Zhang, T., Luan, F., Snavely, N., Xu, Z.: Lvsm: A large view synthesis model with minimal 3d inductive bias. In: ICLR (2025)
17. Kerbl, B., Kopanas, G., Leimkühler, T., Drettakis, G.: 3d gaussian splatting for real-time radiance field rendering. ACM Transactions on Graphics **42**(4), 1–14 (2023)
18. Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.Y., et al.: Segment anything. In: ICCV (2023)
19. Kong, W., Tian, Q., Zhang, Z., Min, R., Dai, Z., Zhou, J., Xiong, J., Li, X., Wu, B., Zhang, J., et al.: Hunyuanvideo: A systematic framework for large video generative models. arXiv preprint arXiv:2412.03603 (2024)

20. Kong, X., Liu, S., Lyu, X., Taher, M., Qi, X., Davison, A.J.: Eschernet: A generative model for scalable view synthesis. In: CVPR (2024)
21. Li, C., Yang, Y., Shao, J., Zhou, H., Schwarz, K., Liao, Y.: Rerope: Repurposing rope for relative camera control. arXiv preprint (2026)
22. Li, R., Yi, B., Liu, J., Gao, H., Ma, Y., Kanazawa, A.: Cameras as relative positional encoding. In: NeurIPS (2025)
23. Li, Y., Si, S., Li, G., Hsieh, C.J., Bengio, S.: Learnable fourier features for multi-dimensional spatial positional encoding. In: NeurIPS (2021)
24. Ling, L., Sheng, Y., Tu, Z., Zhao, W., Xin, C., Wan, K., Yu, L., Guo, Q., Yu, Z., Lu, Y., et al.: Dl3dv-10k: A large-scale scene dataset for deep learning-based 3d vision. In: CVPR (2024)
25. Liu, H., Li, C., Wu, Q., Lee, Y.J.: Visual instruction tuning. In: NeurIPS (2023)
26. Liu, R., Wu, R., Van Hoorick, B., Tokmakov, P., Zakharov, S., Vondrick, C.: Zero-1-to-3: Zero-shot one image to 3d object. In: ICCV (2023)
27. Liu, S., Ng, K.W., Jang, W., Guo, J., Han, J., Liu, H., Douratsos, Y., Pérez, J.C., Zhou, Z., Phung, C., et al.: Scaling sequence-to-sequence generative neural rendering. arXiv preprint arXiv:2510.04236 (2025)
28. Miyato, T., Jaeger, B., Welling, M., Geiger, A.: Gta: A geometry-aware attention mechanism for multi-view transformers. In: ICLR (2024)
29. Pope, R., Douglas, S., Chowdhery, A., Devlin, J., Bradbury, J., Heek, J., Xiao, K., Agrawal, S., Dean, J.: Efficiently scaling transformer inference. Proceedings of machine learning and systems (2023)
30. Press, O., Smith, N.A., Lewis, M.: Train short, test long: Attention with linear biases enables input length extrapolation. In: ICLR (2022)
31. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al.: Learning transferable visual models from natural language supervision. In: ICML (2021)
32. Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., Liu, P.J.: Exploring the limits of transfer learning with a unified text-to-text transformer. In: JMLR (2020)
33. Ravi, N., Gabeur, V., Hu, Y.T., Hu, R., Ryali, C., Ma, T., Khedr, H., Rädle, R., Rolland, C., Gustafson, L., et al.: Sam 2: Segment anything in images and videos. In: ICLR (2025)
34. Reizenstein, J., Shapovalov, R., Henzler, P., Sbordone, L., Labatut, P., Novotny, D.: Common objects in 3d: Large-scale learning and evaluation of real-life 3d category reconstruction. In: ICCV (2021)
35. Ren, J., Tyszkiewicz, M.J., Huang, J., Gojcic, Z.: Tokengs: Decoupling 3d gaussian prediction from pixels with learnable tokens. In: CVPR (2026)
36. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: CVPR (2022)
37. Schenck, C., Reid, I., Jacob, M.G., Bewley, A., Ainslie, J., Rendleman, D., Jain, D., Sharma, M., Dubey, K.A., Wahid, A., et al.: Learning the ropes: Better 2d and 3d position encodings with string. In: ICML (2025)
38. Shaw, P., Uszkoreit, J., Vaswani, A.: Self-attention with relative position representations. In: NAACL (2018)
39. Siméoni, O., Vo, H.V., Seitzer, M., Baldassarre, F., Oquab, M., Jose, C., Khalidov, V., Szafraniec, M., Yi, S., Ramamonjisoa, M., et al.: Dinov3. arXiv preprint arXiv:2508.10104 (2025)
40. Sturm, J., Engelhard, N., Endres, F., Burgard, W., Cremers, D.: A benchmark for the evaluation of rgb-d slam systems. In: IROS (2012)

41. Su, J., Ahmed, M., Lu, Y., Pan, S., Bo, W., Liu, Y.: Roformer: Enhanced transformer with rotary position embedding. Neurocomputing **568** (2024)
42. Szymanowicz, S., Zhang, J.Y., Srinivasan, P., Gao, R., Brussee, A., Holynski, A., Martin-Brualla, R., Barron, J.T., Henzler, P.: Bolt3d: Generating 3d scenes in seconds. In: ICCV (2025)
43. Tang, J., Chen, Z., Chen, X., Wang, T., Zeng, G., Liu, Z.: Lgm: Large multi-view gaussian model for high-resolution 3d content creation. In: ECCV (2024)
44. Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., et al.: Llama: Open and efficient foundation language models. arXiv preprint arXiv:2302.13971 (2023)
45. Ummenhofer, B., Zhou, H., Uhrig, J., Mayer, N., Ilg, E., Dosovitskiy, A., Brox, T.: Demon: Depth and motion network for learning monocular stereo. In: CVPR (2017)
46. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: NeurIPS (2017)
47. Wang, Y., Zhang, Q., Cai, S., Wu, T., Ackermann, J., Kuang, Z., Zheng, Y., Rajič, F., Tang, S., Wetzstein, G.: Bullettime: Decoupled control of time and camera pose for video generation. In: CVPR (2026)
48. Xiang, C., Liu, J., Zhang, J., Yang, X., Fang, Z., Wang, S., Wang, Z., Zou, Y., Su, H., Zhu, J.: Geometry-aware rotary position embedding for consistent video world model. arXiv preprint (2026)
49. Xiao, J., Owens, A., Torralba, A.: Sun3d: A database of big spaces reconstructed using sfm and object labels. In: CVPR (2013)
50. Xu, H., Zhang, J., Cai, J., Rezatofighi, H., Yu, F., Tao, D., Geiger, A.: Unifying flow, stereo and depth estimation. In: TPAMI (2023)
51. Yang, Z., Teng, J., Zheng, W., Ding, M., Huang, S., Xu, J., Yang, Y., Hong, W., Zhang, X., Feng, G., et al.: Cogvideox: Text-to-video diffusion models with an expert transformer. In: ICLR (2025)
52. Yifan, W., Doersch, C., Arandjelović, R., Carreira, J., Zisserman, A.: Input-level inductive biases for 3d reconstruction. In: CVPR (2022)
53. Zhang, K., Bi, S., Tan, H., Xiangli, Y., Zhao, N., Sunkavalli, K., Xu, Z.: Gs-lrm: Large reconstruction model for 3d gaussian splatting. In: ECCV (2024)
54. Zhou, J., Gao, H., Voleti, V., Vasishta, A., Yao, C.H., Boss, M., Torr, P., Rupprecht, C., Jampani, V.: Stable virtual camera: Generative view synthesis with diffusion models. In: ICCV (2025)
55. Zhou, T., Tucker, R., Flynn, J., Fyffe, G., Snavely, N.: Stereo magnification: Learning view synthesis using multiplane images. In: ACM SIGGRAPH (2018)