StreetForward: Perceiving Dynamic Street with Feedforward Causal Attention

Zhongrui Yu, Zhao Wang, Yida Wang, Xueyang Zhang, Yifei Zhan, and Kun Zhan

Li Auto Inc.
{yuzhongrui,wangzhao6,wangyida1,zhangxueyang,zhanyifei,zhankun}@lixiang.com

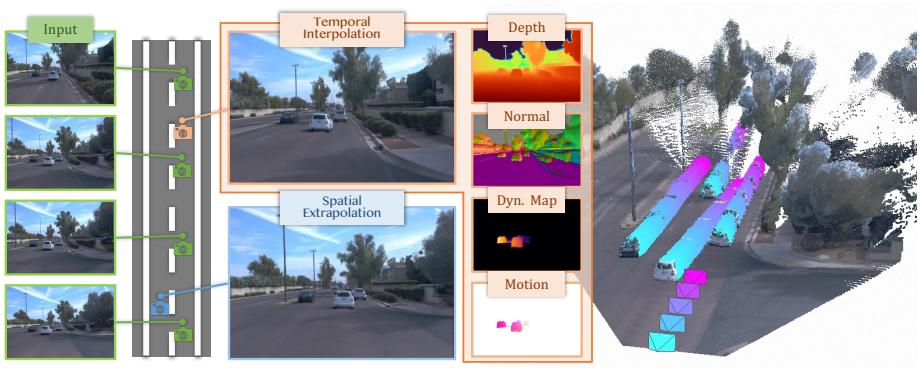

Fig. 1: StreetForward: High-fidelity spatio-temporal extrapolated view synthesis via feedforward 3DGS dynamic street reconstruction. The right pane illustrates our 3DGS representation of a dynamic street with precise per-instance velocities, enabling motion-aware rendering at novel poses and times independent of segmentation or tracking.

Abstract. Feedforward reconstruction is crucial for autonomous driving applications, where rapid scene reconstruction enables efficient utilization of large-scale driving datasets in closed-loop simulation and other downstream tasks, eliminating the need for time-consuming per-scene optimization. We present *StreetForward*, a pose-free and tracker-free feedforward framework for dynamic street reconstruction. Building upon the alternating attention mechanism from Visual Geometry Grounded Transformer (VGGT), we propose a simple yet effective temporal mask attention module that captures dynamic motion information from image sequences and produces motion-aware latent representations. Static content and dynamic instances are represented uniformly with 3D Gaussian Splatting, and are optimized jointly by cross-frame rendering with spatio-temporal consistency, allowing the model to infer per-pixel velocities and produce high-fidelity novel views at new poses and times. We train and evaluate our model on the Waymo Open Dataset, demonstrating superior performance on novel view synthesis and depth estimation compared to existing methods. Furthermore, zero-shot inference on CARLA validates the generalization capability of our approach.

1 Introduction

Dynamic scene reconstruction is essential for closed-loop autonomous driving simulation, which requires accurate modeling both static structure and moving agents. Despite strong progress in 3D reconstruction and neural rendering, most established solutions, ranging from classical SfM (e.g., COLMAP [33]) to Neural Radiance Fields (NeRFs) [6,27,37] and 3D Gaussian Splatting (3DGS) [8,17,48], depend on iterative, scene-specific optimization. This process is computationally expensive and impractical for autonomous-driving simulation where large-scale, diverse, and frequently updated environments must be reconstructed rapidly.

To address efficiency, recent works [3–5, 13, 46, 50, 57] exploit learned priors from large-scale datasets to reduce or eliminate optimization, enabling fast feed-forward inference. DUSt3R [40] predicts relative pose and depth from a pair of image in seconds, and VGGT [35] extends this paradigm to unordered multi-view inputs, producing camera poses and reconstructed point cloud in a single forward pass. Follow up works [9, 28, 39, 41, 58] further extend these capacities, but focus on static scenes and do not solve 4D reconstructions.

Feedforward reconstruction in 4D scenes is bottlenecked by two main challenges. The first one is **geometry degregation under motion**. Recent works (Page4D [55], VGGT4D [14]) show that the Alternating Attention (AA) features of VGGT contain implicit motion cues and they improve geometry by deriving dynamic masks from intermediate features and amplifying motion cues. The second challenge is **motion modeling**. Recent attempts [16,19] attach a lightweight head to AA features to predict scene flow or object motion, but this typically requires explicit 4D supervision, which is limited in scale. It is also constrained by the structure of AA layer in VGGT: as AA is designed for unordered images, it treats tokens from all frames equally in global attention, which weakens its ability to represent directed motion (from source frame to target frame). Other works such as DGGT [4] leverage an external tracker to model object motion, which introduces extra dependencies and potential failure modes.

We introduce **StreetForward**, a feedforward reconstruction framework to address both geometry and motion estimation limitations of VGGT. Building on AA, we add causal masked attention that explicitly models source → target attention to strengthen motion-aware features. We additionally predict Gaussian attributes and optimize geometry and motion jointly via cross-frame rendering. As shown in Fig. 1, our model reconstructs sharp geometry under spatial extrapolation, produces a dynamic map for static/dynamic separation without an external segmentation model, and enables rendering at novel timestamps through predicted motion. This yields high-fidelity view synthesis at both novel poses and novel times, supporting closed-loop autonomous-driving simulation.

In Summary, our contributions are:

- We present a pose-/tracker-/segmentation-free feedforward 4D reconstruction framework with support for novel-view and novel-time rendering. A detailed comparison to existing approaches is provided in Table 1.
- We introduce a simple yet effective causal masked attention mechanism that enhances motion modeling in VGGT without explicit 4D supervision.

	SfM		*Feedforward*						
Capability	StreetGS [48]	OmniRe [8]	VGGT [35]	DVGT [59]	DGGT [4]	STORM [50]	EVolSplat^{4D} [26]	Flux4D [36]	**Ours**
Novel View Synthesis (3DGS)	✓	✓	✗	✗	✓	✓	✓	✓	✓
No camera poses needed	✗	✗	✓	✓	✓	✗	✓	✓	✓
No per-scene optimization	✗	✗	✓	✓	✓	✓	✓	✓	✓
No LiDAR needed	✓	✓	✓	✓	✓	✓	✓	✗	✓
Models objects motions	✓	✓	✗	✗	✓	✓	✓	✓	✓
▷ No tracker needed	✗	✗	—	—	✗	✓	✗	✓	✓
▷ Rigid-object	✓	✓	—	—	✓	✓	✓	✓	✓
▷ Deformable-object	✗	✓	—	—	✓	✓	✗	✗	✓

Table 1: Dynamic 3D street reconstruction approaches. Approaches are grouped into Structure from Motion (SfM) and Feedforward. ✓ indicates the capability is supported while ✗ is not; "No tracker needed" means no instance-level tracker is required to synthesize views at novel timestamps. Our method is highlighted in the last column, which satisfies all listed capabilities.

– Extensive experiments demonstrate the competitive performance of our approach on dynamic street scene reconstruction and strong out-of-domain generalization..

2 Related Works

Dynamic Urban Reconstruction. Dynamic reconstruction methods built on NeRFs [27] and 3DGS [17] usually model motion through canonical-space deformations [1,11,30,42,43,51] or anchor-based transformations [6,18,21,22,24,38], often requiring point tracking or optical flow during training. For urban scenes, dynamic reconstruction is commonly achieved with 3D box annotations [44,48] or scene graphs [45,56]. Recent methods [7,12,49,52,54] further incorporate generative priors to complete missing regions under sparse-view inputs. However, these approaches remain limited by costly per-scene optimization and their reliance on brittle tracking/flow supervision or expensive labels. This motivates data-driven feedforward reconstruction, which learns geometry and motion priors from large-scale data and enables fast inference on unseen scenes.

Feedforward Reconstruction. Recent feedforward methods reconstruct geometry directly from images or videos. DUSt3R [40] predicts pixel-aligned point maps from image pairs, while MASt3R [10], CUT3R [39], and VGGT [35] extend this paradigm to multi-view or video inputs, with follow-ups improving long-sequence fusion [9] and driving scenes [59]. Page-4D [55] and VGGT-4D [14] show that VGGT intermediates carry motion cues and use them to enhance geometry in dynamic scenes. Orthogonally, SCube [32] and subsequent work [23] lift 2D features into 3D feature volumes and decode voxel-centered Gaussians. On the Gaussian Splatting side, AnySplat [15] performs feedforward 3DGS from unconstrained views; 4DGT [47] learns a 4D Gaussian Transformer from monocular videos; and EVolSplat^{4D} [26] proposes an efficient volumetric 4D Gaussian pipeline for street synthesis. Closest to our setting, Flux4D [36], STORM [50], and DGGT [4] address 4D street scenes, but typically depend on LiDAR, simplified motion models, external tracking, or short primitive lifespans, which limits deformable-object modeling and long-term fusion.

3 Methodology

Reconstructing dynamic urban scenes with large 3D feedforward models such as VGGT [35] is challenging because their Alternating-Attention backbones typically prioritize understanding static structure [55]. Our approach models both static and dynamic scene components with 3DGS [17] primitives, representing dynamics through per-pixel velocities, per-frame poses, and decoupled static/dynamic geometry. To maintain consistent static scene geometry, we discard the per-Gaussian lifespan attribute commonly used in prior methods [4,47] Instead, static GS primitives persist for the full duration of the sequence, and we regularize against opacity collapse to ensure optimization focusing on refining geometry rather than suppressing contributions of these GS primitives in novel-view synthesis. For potential dynamic regions predicted by the motion head, the GS primitives are aggregated across frames using the predicted velocities. The motion head is adapted from the AA layers of VGGT [35], with additional causal attention mask to strengthen temporal modeling. Since precise per-pixel velocity annotations are unavailable in most datasets, the motion head in trained without explicit supervision.

We begin with video tokenization in Sec. 3.1 then estimate camera parameters and the static scene contents in Sec. 3.2. Then we introduce the proposed Causal Dynamics module in Sec. 3.3 and the Spatio-temporal Consistency regularization in Sec. 3.4. Our overall pipeline is illustrated in Fig. 2.

3.1 Input Tokenization

Each input video frame is encoded into high-level patchified tokens using a DINO encoder [29], denoted $\mathcal{E}_{\text{DINO}}$. These tokens are then processed through alternating-attention layers $\mathcal{E}_{\text{attns}}$ of depth L, which interleaves cross-frame and intra-frame attention to yield scene-aware features.

Following VGGT notation, let $z_{f,p}^{I,(L)} \in \mathbb{R}^D$ denote the image token of frame $f \in \{1, \ldots, F\}$ at patch index $p \in \{1, \ldots, P\}$ after $\mathcal{E}_{\text{DINO}}$, and collect them into

$$\mathbf{X} \in \mathbb{R}^{B \times F \times P \times D}, \qquad \mathbf{X}[b, f, p, :] = z_{f,p}^{I,(L)}.$$

Here B is the batch size, F is the number of frames per clip, P is the number of patches per frame, and D is the token dimension. The size of image patch in $S \times S$, for an input of size $H \times W$, the number of patches is $P = (H/S) \cdot (W/S)$. When processing cross-frame attention, we flatten the image tokens along the frame–patch axes:

$$\mathbf{Z} = \text{flatten}_{(f,p)}\big(\mathbf{X}\big) \ \in \ \mathbb{R}^{B \times (FP) \times D}.$$

3.2 Pose Estimation and Static Scene Reconstruction

From the latent tokens produced by $\mathcal{E}_{\text{attns}}$, we estimate the camera parameters for each frame and initialize a static 3D representation. A camera head $\mathcal{D}_{\text{cam}}$

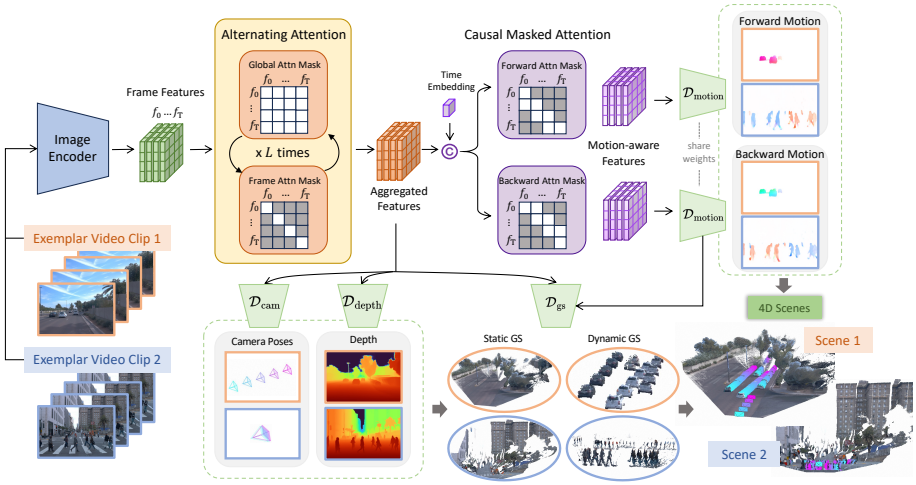

Fig. 2: The proposed **StreetForward** pipeline. We illustrate two common types of dynamic scenes with rigid objects(vehicles) and deformable objects(pedestrians) in and . The input video is first encoded into per-frame patchified features and then processed by L times alternating global- and frame-attention to aggregate information across frames. These aggregated features are directly decoded by a camera head, a depth head and a Gaussian Head to obtain poses, depth and Gaussian attributes. Then causal masked attention is introduce to form motion-aware features, which are used to estimate both forward and backward motion as well as dynamic mask for separating static and dynamic Gaussians. The final 4D scene is obtained by combining static Gaussians with dynamic Gaussians propagated across time using the predicted motion.

predicts intrinsics $\mathbf{K}_f$ and extrinsics $(\mathbf{R}_f, \mathbf{t}_f)$, and a depth head $\mathcal{D}_{\text{dep}}$ produces pixel-aligned depth D_f. For each pixel u, we predict a 3DGS primitive with Gaussian head $\mathcal{D}_{\text{gs}}$

$$g_f(u) = \big(\, \mu_f(u),\ \mathbf{\Sigma}_f(u),\ \alpha_f(u),\ \mathbf{c}_f(u) \,\big),$$

with center

$$\mu_f(u) = \mathbf{R}_f^\top \Big(\mathbf{K}_f^{-1}\, \tilde{\mathbf{u}}\, D_f(u) - \mathbf{t}_f \Big), \quad \tilde{\mathbf{u}} = (u_x, u_y, 1)^\top,$$

and covariance initialized compactly (e.g., $\mathbf{\Sigma}_f(u) = \sigma^2 \mathbf{I}$, scaled by local image/geometry cues [25]), while opacity $\alpha_f(u)$ and color $\mathbf{c}_f(u)$ are initialize from appearance priors.

To focus on the static we suppress likely dynamic regions predicted by our motion head $\mathcal{D}_{\text{motion}}$, which is described in detail in the next section. Specifically, let $s_{f,u}$ denote the dynamic probability at pixel u in frame f, we threshold the dynamic probability and define a static indicator

$$\chi_{f,u} = \mathbb{I}[s_{f,u} \le \tau_{\text{dyn}}]\,.$$

The set of static 3DGS primitives at frame f and fused global static collection across frames are:

$$\mathcal{G}_f^{\text{static}} = \big\{\, g_f(u) \;:\; \chi_{f,u} = 1 \,\big\}, \quad \mathcal{G}^{\text{static}} = \bigcup_{f=1}^{F} \mathcal{G}_f^{\text{static}}.$$

3.3 Causal Dynamics Modeling

To facilitate motion-aware downstream decoding, we firstly augment the tokens output by $\mathcal{E}_{\text{attns}}$ with explicit temporal information. Specifically, we concatenate a sinusoidal embedding $\tau_f \in \mathbb{R}^{d_t}$ to every token in frame f:

$$\tilde{\mathbf{X}}[b, f, p, :] = \mathbf{X}[b, f, p, :] \;\oplus\; \tau_f \;\in\; \mathbb{R}^{D'}, \quad D' = D + d_t,$$

where $\oplus$ denotes concatenation along the feature dimension. The augmented tokens are flattened along the frame-patch axes to obtain $\tilde{\mathbf{Z}} \in \mathbb{R}^{B\times(FP)\times D'}$

Operating on the flattened tokens $\tilde{\mathbf{Z}}$, we apply multi-head attention with a **frame-structured causal mask** that restricts query–key pairs to a designated source–target frame relation. Let frame(i) map a flattened token index $i \in \{1, \ldots, F\cdot P\}$ to its frame. For a given source frame f_{s} and target frame f_{t}, the binary mask $\mathbf{M} \in \{0,1\}^{B\times H\times(F\cdot P)\times(F\cdot P)}$ is

$$\mathbf{M}[b, h, i, j] = \begin{cases} 1, & \text{if frame}(i) = f_{\text{s}} \text{ and frame}(j) = f_{\text{t}}, \\ 0, & \text{otherwise,} \end{cases}$$

where $f_{\text{t}} = f_{\text{s}} + 1$ for forward prediction and $f_{\text{t}} = f_{\text{s}} - 1$ for backward prediction. With per-head projections $\mathbf{W}_Q^{(h)}, \mathbf{W}_K^{(h)}, \mathbf{W}_V^{(h)} \in \mathbb{R}^{D'\times d_h}$ for $h \in \{1, \ldots, H\}$, we set

$$\mathbf{Q}^{(h)} = \tilde{\mathbf{Z}}\,\mathbf{W}_Q^{(h)}, \quad \mathbf{K}^{(h)} = \tilde{\mathbf{Z}}\,\mathbf{W}_K^{(h)}, \quad \mathbf{V}^{(h)} = \tilde{\mathbf{Z}}\,\mathbf{W}_V^{(h)},$$

and compute

$$\mathbf{A}^{(h)} = \text{softmax}\left(\frac{\mathbf{Q}^{(h)}{\mathbf{K}^{(h)}}^{\top}}{\sqrt{d_h}} + \log \mathbf{M}\right) \mathbf{V}^{(h)}, \tag{1}$$

where $\log \mathbf{M}$ assigns $-\infty$ to disallowed entries. Concatenating heads and applying an output projection $\mathbf{W}_O \in \mathbb{R}^{(Hd_h)\times D'}$ yields dynamics-aware features

$$\hat{\mathbf{Z}} = \text{Concat}_h\big(\mathbf{A}^{(h)}\big)\,\mathbf{W}_O \in \mathbb{R}^{B\times(F\cdot P)\times D'},$$

which we reshape to $\mathbf{Y} \in \mathbb{R}^{B\times F\times P\times D'}$.

Velocity Decoding To estimate numerical motion, we regress a dense velocity field from the motion-aware tokens $\mathbf{Y}$ using a DPT-style decoder $\mathcal{D}_{\text{motion}}$ [31]. The decoder converts the tokens into dense feature maps, fuses multi-scale features and ouptus per-pixel forward and backward velocities $\mathbf{v}_{f,u} \equiv [\mathbf{v}_{f,u}^{+}, \mathbf{v}_{f,u}^{-}] \in \mathbb{R}^6$, together with a pixel-aligned dynamic probability $\sigma_{f,u} > 0$ to weight training losses and to distinguish dynamic from static region. As shown in Fig. 3, our proposed Causal Dynamics yields precise velocities while avoiding false motion on static content (e.g, parked vehicles).

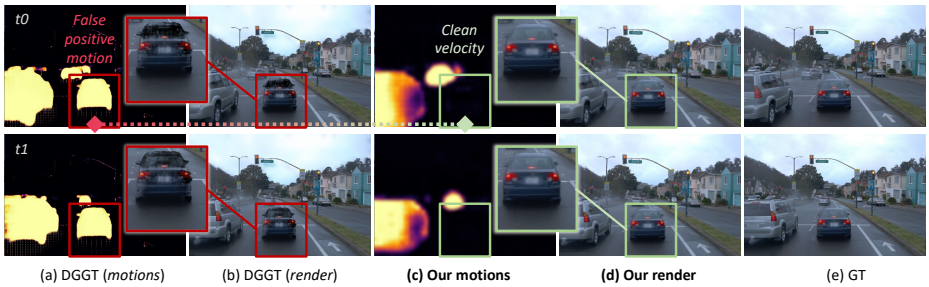

Fig. 3: Robustness to false-positive motion-mask prompts. Our method correctly assigns near-zero dynamic probability to parked or slowly moving vehicles, even if they are labeled as dynamic in annotations. This accurate understanding of motion (c) eliminates the motion ambiguity seen in DGGT (a), resulting in stable and clear rendering of the target vehicle over time (d).

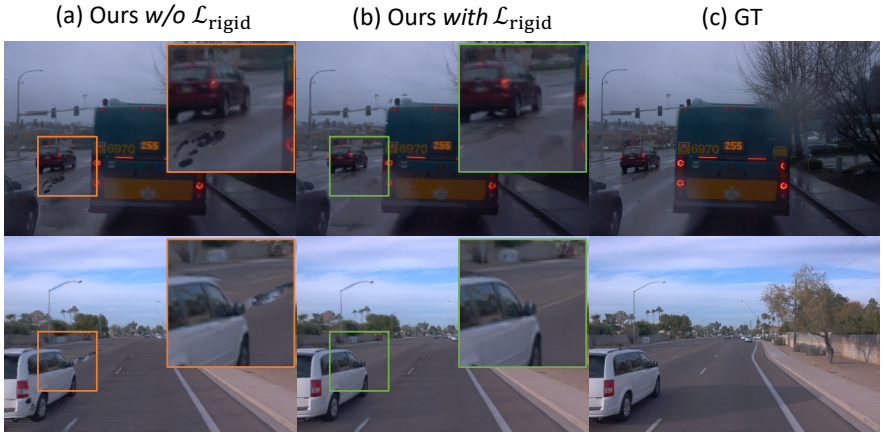

Fig. 4: Enforcing rigid regularization ($\mathcal{L}_{\text{rigid}}$) removes the structural floaters seen around rigid objects in (a).

3.4 Motion Consistency

Local Rigidity Motion. Let $\mu_{f,u} \in \mathbb{R}^3$ be the 3D Gaussian location associated with pixel (f, u) and let $\mathbf{v}_{f,u} \in \mathbb{R}^3$ be the velocity predicted by the motion decoder. With time step Δt, we propagate the Gaussian center as

$$\mu_{f+1,u} = \mu_{f,u} + \Delta t\, \mathbf{v}_{f,u}$$

In practice, learning motion purely from rendering and reconstruction losses is ill-posed, and explicit per-pixel motion supervision is typically unavailable. In addition, motion estimates from off-the-shelf trackers [4, 19, 47] lack robustness. To stabilize motion learning, we impose a local rigidity prior [24] that encourages piecewise-rigid motion within local neighborhoods in both image space and 3D space. For 2D rigidity, we encourage neighboring pixels to share similar motion:

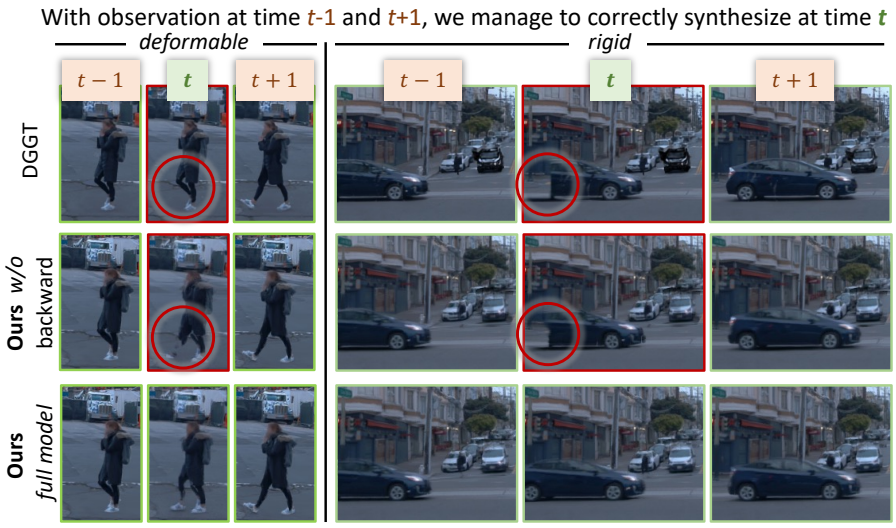

Fig. 5: Temporal interpolation. With observation at time t-1 and t+1, we manage to correctly synthesize at time t. The figure compares the results of our full model against an ablation without backward fusion by predicting forward velocity only, and DGGT [4]. The left three columns show pedestrian is with correct poses, and the right three columns display vehicle geometry reconstruction, where our model delivers complete geometries.

$$\underbrace{\mathcal{L}_{\text{rigid-2D}}}_{\{\mathcal{D}_{\text{motion}},\mathcal{D}_{\text{dep}}\}} = \sum_{f}\sum_{u}\sum_{u'\in\mathcal{N}(u)} \omega(\sigma_{f,u},\sigma_{f,u'})\,\|v_{f,u}-v_{f,u'}\|_2^2 \tag{2}$$

where $\mathcal{N}(u)$ is a fixed 2D local window, and $\omega(\cdot,\cdot)$ converts the dynamic confidence to weight. For 3D rigidity, we compute the K nearest neighbors of each 3D point and enforce consistent velocities:

$$\underbrace{\mathcal{L}_{\text{rigid-3D}}}_{\{\mathcal{D}_{\text{motion}},\mathcal{D}_{\text{dep}}\}} = \sum_{f}\sum_{u}\sum_{u'\in\mathcal{N}_K(u)} \omega(\sigma_{f,u},\sigma_{f,u'})\,\|v_{f,u}-v_{f,u'}\|_2^2 \tag{3}$$

The overall rigidity motion regularization is:

$$\mathcal{L}_{\text{rigid}} = \mathcal{L}_{\text{rigid-2D}} + \mathcal{L}_{\text{rigid-3D}} \tag{4}$$

Temporal consistency. We translate dynamic Gaussians across time according to predicted velocities, enabling temporal fusion that mitigates occlusions. For a dynamic Gaussian centered at $\mu_{f,u}$, we predict its forward and backward velocities $v_u^+ = v_u^{f\to f+1}$ and $v_u^- = v_u^{f\to f-1}$. We obtain proposals in the adjacent frames via propagation:

$$\mu_u^{f-1|f} = \mu_u^f + \Delta t\,\mathbf{v}_u^-, \quad \mu_u^{f+1|f} = \mu_u^f + \Delta t\,\mathbf{v}_u^+.$$

Likewise, dynamic Gaussians from adjacent frames are warped into the current frame f. At render time we deliberately exclude the dynamic Gaussians parameterized at f itself. This forces dynamic object in frame f must explained from adjacent-frame dynamics, preventing degenerate placement of other frames' instance out of the current field of view during training. Let $\mathcal{T}$ denote a temporal window that excludes the current timestamp ($t \neq f$). The set of Gaussians used to render frame f is

$$\mathcal{G}^f \;=\; \mathcal{G}^{\text{static}} \;\cup\; \bigcup_{t\in\mathcal{T}} \mathcal{G}^{\text{dynamic}}_{f\leftarrow t},$$

where $\mathcal{G}^{\text{dynamic}}_{f\leftarrow t}$ are dynamic Gaussians from frame t warped into frame f (e.g., $t \in \{f-1, f+1\}$ by default). To regularize the motion field, we adopt a constant-velocity assumption over the short interval Δt, and penalize deviations from forward–backward symmetry:

$$\mathcal{L}_{\text{fb}} = \sum_u ||\mathbf{v}_u^{f\to f+1} + \mathbf{v}_u^{f\to f-1}||_1. \tag{5}$$

As illustrated in Fig. 5, this joint forward–backward fusion enables novel temporal synthesis for both deformable and rigid objects, producing realistic actions and complete geometries.

4 Training

We outline the reconstruction objective (Sec. 4.1), geometric regularization (Sec. 4.2), and motion regularization (Sec. 4.3).

4.1 Rendering Losses

For the parametric model defined by our architecture, a per-frame image reconstruction objective serves as an effective baseline. We jointly optimize $\mathcal{E}_{\text{attns}}$, $\mathcal{D}_{\text{cam}}$, $\mathcal{D}_{\text{gs}}$, $\mathcal{D}_{\text{motion}}$, and $\mathcal{D}_{\text{dep}}$, using an RGB reconstruction loss:

$$\underbrace{\mathcal{L}_{\text{rgb}}}_{\{\mathcal{E}_{\text{attns}},\,\mathcal{D}_{\text{cam}},\,\mathcal{D}_{\text{vel}},\,\mathcal{D}_{\text{dep}}\}} = \sum_f \sum_u \left\| \hat{\mathcal{I}}_f(u) - \mathcal{I}_{\text{gt},f}(u) \right\|_1, \tag{6}$$

where $\hat{\mathcal{I}}$ is rendering with $\mathcal{G}^f$ without using dynamic instance from current frame. To avoid degenerate solutions where 3DGS opacities collapse toward zero instead of relocating splats to correct 3D positions, we bias early training toward geometry updates by attenuating opacity gradients from $\mathcal{L}_{\text{rgb}}$, while using the dedicated opacity regularizer described below.

4.2 Geometric Regularization

Rendering-only optimization can improve original-view or short-baseline interpolation but risks overfitting training views and degrading 3D geometric precision. This harms view extrapolation, especially in autonomous driving when the egocentric viewpoint shifts laterally. We therefore add geometry-focused regularizers.

Opacity stabilization. We encourage splats to retain non-trivial opacity (closer to 1 than 0) so they contribute to multi-view rendering rather than masking geometric errors:

$$\underbrace{\mathcal{L}_{\alpha}}_{\{\mathcal{D}_{\text{dep}}\}} = \lambda_{\alpha} \sum_{f} \sum_{k \in \mathcal{V}_f} w_{f,k} \, ||1 - \alpha_k^f||_2^2, \tag{7}$$

where $\lambda_{\alpha} > 0$ is a scalar weight, $\mathcal{V}_f$ indexes visible splats at frame f, and $w_{f,k} \geq 0$ is an optional per-splat weight (e.g., based on visibility or static/dynamic masks). This term prevents opacity collapse, while (6) primarily drives geometry correction.

Depth consistency. We align the predicted per-frame depth with the depth produced by 3DGS rasterization, with optional supervision by ground-truth depth when available. Under front-to-back alpha compositing, let $\{(z_i(u), \alpha_i(u))\}_{i=1}^{N(u)}$ be splats along the ray at pixel u, sorted by increasing depth; define $T_1(u) = 1$ and $T_i(u) = \prod_{j<i}(1 - \alpha_j(u))$ [17]. The depth consistency is enforced as

$$\underbrace{\mathcal{L}_{\text{depth}}}_{\{\mathcal{D}_{\text{dep}}, \mathcal{E}_{\text{attns}}\}} = \sum_{f} \sum_{u} \left\| D_f(u) - \sum_{i=1}^{N(u)} T_i(u) \, \alpha_i(u) \, z_i(u) \right\|_1 . \tag{8}$$

4.3 Motion Regularization

The local rigidity motion regularization in (4) (Sec. 3.4) encourages dynamic objects to follow a coherent transform over time, while the forward–backward agreement in (5) promotes smooth, temporally consistent per-instance trajectories by considering both motion directions. These motion regularizers complement $\mathcal{L}_{\text{rgb}}$, $\mathcal{L}_{\alpha}$, and $\mathcal{L}_{\text{depth}}$ to balance the fidelity of appearance with stable 3D geometry and dynamics.

5 Implementation

Our image encoder, $\mathcal{E}_{\text{DINO}}$, is taken from DINOv2 [29] and kept frozen throughout training. The alternating-attention backbone consists 36 layers, with 18 frame-attention layers interleaved with 18 global-attention layers. The AA layers, camera head and depth head are initialized from Pi3 [41] pretrained weights, frozen at initialization, and progressively unfrozen as training proceeds.

For temporal modeling, we employ a 4-layer causal masked-attention module with the same architecture as the global-attention block, with 16 heads each layer. We randomly downsample sequences temporally by $1\times$ - $4\times$, making the causal horizon effectively ± 1 to ± 4 frames and encouraging robustness to varying temporal sampling rates.

The Gaussian head and motion head share the depth-head architecture, with different output channel sizes. We initialize the motion head conservatively by setting the weights of its final linear layer to zero. We further adopt a staged

training schedule: Initially, the motion head is excluded from training, and we disable dynamic Gaussian aggregation across time. We render each frame using static Gaussians from all frames plus the dynamic Gaussians from the current frame only. In the second stage, once the Gaussian and depth heads produce high-quality single-frame reconstructions, we activate the motion head and perform temporal aggregation of dynamic Gaussians. This strategy stabilizes optimization and allows for accurate per-frame geometry before temporal fusion.

6 Experiments

We evaluate **StreetForward** on the Waymo Open Dataset [34] and the Carla dataset. Baselines include recent feedforward methods STORM [50], DGGT [4], AnySplat [15], and DepthAnythingv3 [20], as well as the optimization-based PVG [6]. For depth estimation, we additionally compare with VGGT [35] and Pi3 [41]. Unless otherwise noted, best, second-best, and third-best results in the tables are marked with **best**, second, and third, respectively.

6.1 Waymo

We follow the evaluation protocol of STORM [50]. The Model is trained on the training split (798 scenes, approximately 190–200 frames each) and evaluated on the test split (202 scenes), covering diverse scenarios such as dense traffic, nighttime scenes, and rainy weather. Each evaluation video clip consists of 20 images, and the 1st, 5th, 10th, and 15th frames are provided as context images, and the remaining frames are treated as target views for evaluation. We report standard metrics including Peak Signal-to-Noise Ratio (PSNR), Structure Similarity Index Measure (SSIM) for rendering quality and Depth Root Mean Square Error (RMSE) for geometry accuracy.

View synthesis. We report PSNR/SSIM on (i) dynamic-only regions and (ii) the full image in Table 2. Our proposed method StreetForward achieves the best performance on dynamic regions, indicating more faithful synthesis of moving content without relying on tracking or segmentation at inference time. On the full-frame metrics, our approach remains highly competitive and is close to the leading baseline. We highlight the best and second-best values in each column. Baseline results for PVG [6], STORM [50] and DGGT [4] are sourced from their published reports. Fig. 7 provides qualitative results on challenging cases. For each example, the two subfigures correspond to temporal interpolation and extrapolation respectively. AnySplat [15] explicitly model object motion, which leads to ghosting artifacts on dynamic objects and opacity collapse in some cases that creates visible holes. STORM [50] produces blurred dynamic objects due to inaccurate velocity estimation. DGGT [4] relies on tracker [53] to predict dynamic motion and can yield implausible trajectories under extrapolation, such as vehicle drifting in the sky, collisions, or partially missing objects. Our method produces more coherent motion and geometry under both interpolation and extrapolation.

Fig. 6: Qualitative comparision on Waymo Open Datasets.

Method	Dynamic only		Full image	
	PSNR↑	SSIM↑	PSNR↑	SSIM↑
PVG [6]	15.51	0.128	22.38	0.661
AnySplat [15]	15.99	0.418	23.32	0.687
DA3 w/ GS Head [20]	15.96	0.429	23.27	0.686
STORM [50]	22.10	0.624	26.38	0.794
DGGT [4]	20.99	0.821	**27.41**	**0.846**
Ours	**24.30**	**0.827**	27.01	0.818

Table 2: Waymo original-view (intrapolation) synthesis in PSNR and SSIM. Best and second-best per column are indicated by color boxes. Values for STORM and DGGT are transcribed from the cited papers.

Method	Static Only	Dynamic Only	Full Image
VGGT [35]	3.84	6.36	4.07
Pi3 [41]	5.46	7.93	5.61
DA3 [20]	11.25	12.20	11.66
PVG [6]	-	15.91	13.01
STORM [50]	-	7.50	5.48
AnySplat [15]	9.83	10.45	10.31
DGGT [4]	3.47	6.37	4.08
Ours	**3.09**	**3.45**	**3.14**

Table 3: Waymo sparse-depth RMSE ↓ (LiDAR). Values for STORM and DGGT are transcribed from the cited papers.

Depth estimation. Geometry accuracy is evaluated using per-pixel depth RMSE against sparse LiDAR depth projected into each camera view, and the results are reported in Table 3. We report errors on static-only regions, dynamic-only regions and full image (all pixels with valid LiDAR depth). StreetForward consistently yields the lowest error, with especially strong improvements on moving vehicles, highlighting the benefit of causal motion encoding and velocity-aware fusion for maintaining dynamic geometry. Fig. 7 shows consistent qualitative gains, including cleaner reconstruction of thin structures (poles, traffic lights, cables) and sharper geometry and renderings for dynamic vehicles.

6.2 Carla

For domain-transfer validation, we render a Carla test set of 100 clips with matched camera intrinsics and fixed extrinsics, ensuring controlled geometry and viewpoint distribution. We additionally render shifted viewpoints (left/right lane offsets) from the original camera views, enabling a controlled evaluation of novel-view synthesis. Our model is trained on Waymo training set and evaluate on Carla in a zero-shot manner without any fine-tuning. Following the same protocol as above, we report novel-view synthesis quality (PSNR/SSIM) on both dynamic-only regions and full images, as well as dense depth accuracy (RMSE) in Table 4 and Table 5. Across all metrics and evaluation regimes, StreetForward outperforms the strongest baseline, with particularly clear advantages on dynamic regions. These results indicate improved robustness to domain shift in both appearance reconstruction and motion-consistent geometry.

Method	Dynamic Only		Full Image	
	PSNR↑	SSIM↑	PSNR↑	SSIM↑
AnySplat [15]	13.18	0.339	19.57	0.552
DA3 w/ GS Head [20]	14.11	0.399	20.41	0.591
STORM [50]	12.80	0.313	18.23	0.542
DGGT [4]	21.32	0.691	23.91	0.747
Ours	**22.37**	**0.741**	**24.62**	**0.759**

Table 4: Carla original-view RGB synthesis under zero-shot transfer from Waymo. We report PSNR↑ and SSIM↑ for dynamic-only regions (vehicle masks from SAMv3 [2] used for evaluation only) and for the full image.

Method	Static Only	Dynamic Only	Full Image
VGGT [35]	6.11	8.55	6.24
Pi3 [41]	9.85	12.76	9.22
AnySplat [15]	8.29	13.12	9.14
DGGT [4]	6.53	7.93	6.56
Ours	**6.01**	**7.16**	**6.01**

Table 5: Carla dense-depth RMSE↓ under zero-shot transfer from Waymo. The static-only score excludes dynamic pixels; dynamic-only is computed on vehicle regions; full-image uses all valid pixels.

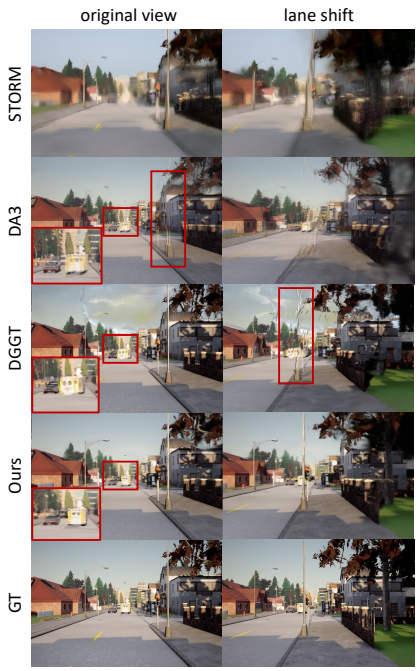

Fig. 7: Qualitative comparison on Carla Datasets.

6.3 Ablation Study

We ablate key components on the Waymo Open Dataset [34] and report PSNR and SSIM on dynamic-only and full-image partitions (Table 6). The full model

consistently outperforms all variants. In brief, causal masked attention is crucial for temporally directed motion encoding; local rigidity and forward–backward consistency stabilize trajectories (see qualitative evidence in Fig. 4 and Fig. 5; and opacity stabilization prevents a common failure under multi-frame fusion where primitives reduce opacity instead of aligning geometrically, which otherwise produces low-opacity artifacts in novel views in Fig. 3.

Table 6: Ablation on causal modeling and regularization (Waymo). Variants: w/o causal attention; fixed horizon (k=1); w/o $\mathcal{L}_{\text{rigid}}$; w/o backward velocity (v^-); w/o $\mathcal{L}_{\text{fb}}$; w/o $\mathcal{L}_\alpha$.

Method / Variant	Dynamic only		Full image	
	PSNR↑	SSIM↑	PSNR↑	SSIM↑
Ours	**24.30**	**0.827**	**27.01**	**0.818**
w/o causal attention	22.36	0.714	24.07	0.781
fixed causal horizon k=1	22.96	0.722	24.66	0.760
w/o $\mathcal{L}_{\text{rigid}}$	22.64	0.743	26.21	0.786
w/o v^- (forward-only)	23.26	0.786	26.88	0.809
w/o $\mathcal{L}_{\text{fb}}$	23.03	0.721	26.43	0.794
w/o $\mathcal{L}_\alpha$	21.02	0.659	25.85	0.683

Causal dynamic modeling. Removing causal masked attention or fixing the horizon (k=1) weakens temporal discrimination in motion tokens, leading to inconsistent velocities under varying sampling rates and noticeable drops on dynamic-only PSNR/SSIM. These failures manifest as ghosting and temporal blur in interpolation, as seen in Fig. 5, and reduced robustness to false-positive motion prompts in Fig. 3.

Motion regularization. Local rigidity ($\mathcal{L}_{\text{rigid}}$) reduces velocity drift and preserves multi-frame aggregation; without it, structural floaters and collapsed proposals appear (Fig. 4), consistent with the quantitative drop in Table 6. Forward-only motion (w/o v^-) remains competitive numerically but loses occlusion-recovery details on dynamic regions; dropping $\mathcal{L}_{\text{fb}}$ further reduces temporal coherence, as reflected in Fig. 5.

Geometric regularization. Opacity stabilization ($\mathcal{L}_\alpha$) is essential under cross-frame fusion. Without it, primitives tend to lower opacity instead of aligning in 3D, yielding low-opacity artifacts and degraded PSNR/SSIM; Fig. 3 illustrates how the regularizer prevents such "black holes" and keeps splats participating in rendering so the RGB loss refines geometry rather than masking errors.

7 Conclusion

Our proposed **StreetForward** enables feedforward dynamic street reconstruction with causal cross-frame attention and velocity decoding, without requiring tracking or segmentation at inference. On Waymo, it achieves state-of-the-art depth estimation and strong dynamic-instance view synthesis while remaining competitive on full-image rendering. It also shows strong generalization on out-of-domain data.

References

1. Cao, A., Johnson, J.: Hexplane: A fast representation for dynamic scenes. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 130–141 (2023)
2. Carion, N., Gustafson, L., Hu, Y.T., Debnath, S., Hu, R., Suris, D., Ryali, C., Alwala, K.V., Khedr, H., Huang, A., Lei, J., Ma, T., Guo, B., Kalla, A., Marks, M., Greer, J., Wang, M., Sun, P., Rädle, R., Afouras, T., Mavroudi, E., Xu, K., Wu, T.H., Zhou, Y., Momeni, L., Hazra, R., Ding, S., Vaze, S., Porcher, F., Li, F., Li, S., Kamath, A., Cheng, H.K., Dollár, P., Ravi, N., Saenko, K., Zhang, P., Feichtenhofer, C.: Sam 3: Segment anything with concepts (2025), https://arxiv.org/abs/2511.16719
3. Charatan, D., Li, S.L., Tagliasacchi, A., Sitzmann, V.: pixelsplat: 3d gaussian splats from image pairs for scalable generalizable 3d reconstruction. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 19457–19467 (2024)
4. Chen, X., Xiong, Z., Chen, Y., Li, G., Wang, N., Luo, H., Chen, L., Sun, H., Wang, B., Chen, G., et al.: Dggt: Feedforward 4d reconstruction of dynamic driving scenes using unposed images. arXiv preprint arXiv:2512.03004 (2025)
5. Chen, Y., Xu, H., Zheng, C., Zhuang, B., Pollefeys, M., Geiger, A., Cham, T.J., Cai, J.: Mvsplat: Efficient 3d gaussian splatting from sparse multi-view images. In: European Conference on Computer Vision. pp. 370–386. Springer (2024)
6. Chen, Y., Gu, C., Jiang, J., Zhu, X., Zhang, L.: Periodic vibration gaussian: Dynamic urban scene reconstruction and real-time rendering. International Journal of Computer Vision (2026)
7. Chen, Y., Wang, Y., Zhang, X., Zhan, K., Jia, P., Zhan, Y., Lang, X.: Styledstreets: Multi-style street simulator with spatial and temporal consistency. arXiv preprint arXiv:2503.21104 (2025)
8. Chen, Z., Yang, J., Huang, J., de Lutio, R., Esturo, J.M., Ivanovic, B., Litany, O., Gojcic, Z., Fidler, S., Pavone, M., et al.: Omnire: Omni urban scene reconstruction. In: The Thirteenth International Conference on Learning Representations
9. Deng, K., Ti, Z., Xu, J., Yang, J., Xie, J.: Vggt-long: Chunk it, loop it, align it–pushing vggt's limits on kilometer-scale long rgb sequences. arXiv preprint arXiv:2507.16443 (2025)
10. Duisterhof, B.P., Zust, L., Weinzaepfel, P., Leroy, V., Cabon, Y., Revaud, J.: MASt3r-sfm: a fully-integrated solution for unconstrained structure-from-motion. In: International Conference on 3D Vision 2025 (2025), https://openreview.net/forum?id=5uw1GRBFoT
11. Fridovich-Keil, S., Meanti, G., Warburg, F.R., Recht, B., Kanazawa, A.: K-planes: Explicit radiance fields in space, time, and appearance. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12479–12488 (2023)
12. Gao, R., Chen, K., Li, Z., Hong, L., Li, Z., Xu, Q.: Magicdrive3d: Controllable 3d generation for any-view rendering in street scenes. arXiv preprint arXiv:2405.14475 (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. arXiv preprint arXiv:2311.04400 (2023)
14. Hu, Y., Cheng, C., Yu, S., Guo, X., Wang, H.: Vggt4d: Mining motion cues in visual geometry transformers for 4d scene reconstruction. arXiv preprint arXiv:2511.19971 (2025)

15. Jiang, L., Mao, Y., Xu, L., Lu, T., Ren, K., Jin, Y., Xu, X., Yu, M., Pang, J., Zhao, F., et al.: Anysplat: Feed-forward 3d gaussian splatting from unconstrained views. ACM Transactions on Graphics (TOG) **44**(6), 1–16 (2025)
16. Karhade, J., Keetha, N., Zhang, Y., Gupta, T., Sharma, A., Scherer, S., Ramanan, D.: Any4d: Unified feed-forward metric 4d reconstruction. arXiv preprint arXiv:2512.10935 (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) (July 2023), `https://repo-sam.inria.fr/fungraph/3d-gaussian-splatting/`
18. Li, Z., Wang, Q., Cole, F., Tucker, R., Snavely, N.: Dynibar: Neural dynamic image-based rendering. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4273–4284 (2023)
19. Lin, C., Lin, Y., Pan, P., Yu, Y., Yan, H., Fragkiadaki, K., Mu, Y.: Movies: Motion-aware 4d dynamic view synthesis in one second. arXiv preprint arXiv:2507.10065 (2025)
20. Lin, H., Chen, S., Liew, J., Chen, D.Y., Li, Z., Shi, G., Feng, J., Kang, B.: Depth anything 3: Recovering the visual space from any views. arXiv preprint arXiv:2511.10647 (2025)
21. Liu, Y.L., Gao, C., Meuleman, A., Tseng, H.Y., Saraf, A., Kim, C., Chuang, Y.Y., Kopf, J., Huang, J.B.: Robust dynamic radiance fields. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13–23 (2023)
22. Lu, F., Xu, Y., Chen, G., Li, H., Lin, K.Y., Jiang, C.: Urban radiance field representation with deformable neural mesh primitives. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 465–476 (2023)
23. Lu, Y., Ren, X., Yang, J., Shen, T., Wu, Z., Gao, J., Wang, Y., Chen, S., Chen, M., Fidler, S., et al.: Infinicube: Unbounded and controllable dynamic 3d driving scene generation with world-guided video models. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 27272–27283 (2025)
24. Luiten, J., Kopanas, G., Leibe, B., Ramanan, D.: Dynamic 3d gaussians: Tracking by persistent dynamic view synthesis. In: 3DV (2024)
25. Meuleman, A., Shah, I., Lanvin, A., Kerbl, B., Drettakis, G.: On-the-fly reconstruction for large-scale novel view synthesis from unposed images. ACM Transactions on Graphics **44**(4) (2025)
26. Miao, S., Li, S., Wang, P., Bai, D., Liu, B., Wang, Y., Geiger, A., Liao, Y.: Evolsplat4d: Efficient volume-based gaussian splatting for 4d urban scene synthesis. arXiv preprint arXiv:2601.15951 (2026)
27. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: Nerf: Representing scenes as neural radiance fields for view synthesis. Communications of the ACM **65**(1), 99–106 (2021)
28. Murai, R., Dexheimer, E., Davison, A.J.: Mast3r-slam: Real-time dense slam with 3d reconstruction priors. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 16695–16705 (2025)
29. Oquab, M., Darcet, T., Moutakanni, T., Vo, H., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., et al.: Dinov2: Learning robust visual features without supervision. arXiv preprint arXiv:2304.07193 (2023)
30. Pumarola, A., Corona, E., Pons-Moll, G., Moreno-Noguer, F.: D-nerf: Neural radiance fields for dynamic scenes. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10318–10327 (2021)

31. Ranftl, R., Bochkovskiy, A., Koltun, V.: Vision transformers for dense prediction. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 12179–12188 (2021)
32. Ren, X., Lu, Y., Liang, H., Wu, Z., Ling, H., Chen, M., Fidler, S., Williams, F., Huang, J.: Scube: Instant large-scale scene reconstruction using voxsplats. Advances in Neural Information Processing Systems **37**, 97670–97698 (2024)
33. Schönberger, J.L., Zheng, E., Pollefeys, M., Frahm, J.M.: Pixelwise view selection for unstructured multi-view stereo. In: European Conference on Computer Vision (ECCV) (2016)
34. Sun, P., Kretzschmar, H., Dotiwalla, X., Chouard, A., Patnaik, V., Tsui, P., Guo, J., Zhou, Y., Chai, Y., Caine, B., et al.: Waymo open dataset: Autonomous driving dataset. CVPR (2020)
35. Wang, J., Chen, M., Karaev, N., Vedaldi, A., Rupprecht, C., Novotny, D.: Vggt: Visual geometry grounded transformer. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 5294–5306 (2025)
36. Wang, J., Che, H., Chen, Y., Yang, Z., Goli, L., Manivasagam, S., Urtasun, R.: Flux4d: Flow-based unsupervised 4d reconstruction. In: The Thirty-ninth Annual Conference on Neural Information Processing Systems
37. Wang, P., Liu, L., Liu, Y., Theobalt, C., Komura, T., Wang, W.: Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. Advances in Neural Information Processing Systems **34**, 27171–27183 (2021)
38. Wang, Q., Ye, V., Gao, H., Austin, J., Li, Z., Kanazawa, A.: Shape of motion: 4d reconstruction from a single video (2024)
39. Wang, Q., Zhang, Y., Holynski, A., Efros, A.A., Kanazawa, A.: Continuous 3d perception model with persistent state. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 10510–10522 (2025)
40. Wang, S., Leroy, V., Cabon, Y., Chidlovskii, B., Revaud, J.: Dust3r: Geometric 3d vision made easy. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 20697–20709 (2024)
41. Wang, Y., Zhou, J., Zhu, H., Chang, W., Zhou, Y., Li, Z., Chen, J., Pang, J., Shen, C., He, T.: π^3: Permutation-equivariant visual geometry learning. arXiv preprint arXiv:2507.13347 (2025)
42. Wu, G., Yi, T., Fang, J., Xie, L., Zhang, X., Wei, W., Liu, W., Tian, Q., Wang, X.: 4d gaussian splatting for real-time dynamic scene rendering. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 20310–20320 (2024)
43. Wu, T., Zhong, F., Tagliasacchi, A., Cole, F., Oztireli, C.: D^ 2nerf: Self-supervised decoupling of dynamic and static objects from a monocular video. Advances in Neural Information Processing Systems **35**, 32653–32666 (2022)
44. Wu, Z., Liu, T., Luo, L., Zhong, Z., Chen, J., Xiao, H., Hou, C., Lou, H., Chen, Y., Yang, R., et al.: Mars: An instance-aware, modular and realistic simulator for autonomous driving. In: CAAI International Conference on Artificial Intelligence. pp. 3–15. Springer (2023)
45. Xiong, Y., Zhou, X., Wang, Y., Sun, D., Yang, M.H.: Drivinggaussian++: Towards realistic reconstruction and editable simulation for surrounding dynamic driving scenes. IEEE Transactions on Pattern Analysis and Machine Intelligence (2026)
46. Xu, H., Peng, S., Wang, F., Blum, H., Barath, D., Geiger, A., Pollefeys, M.: Depthsplat: Connecting gaussian splatting and depth. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 16453–16463 (2025)
47. Xu, Z., Li, Z., Dong, Z., Zhou, X., Newcombe, R., Lv, Z.: 4dgt: Learning a 4d gaussian transformer using real-world monocular videos (2025)

48. Yan, Y., Lin, H., Zhou, C., Wang, W., Sun, H., Zhan, K., Lang, X., Zhou, X., Peng, S.: Street gaussians: Modeling dynamic urban scenes with gaussian splatting. In: European Conference on Computer Vision. pp. 156–173. Springer (2024)
49. Yan, Y., Xu, Z., Lin, H., Jin, H., Guo, H., Wang, Y., Zhan, K., Lang, X., Bao, H., Zhou, X., Peng, S.: Streetcrafter: Street view synthesis with controllable video diffusion models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2025)
50. Yang, J., Huang, J., Chen, Y., Wang, Y., Li, B., You, Y., Sharma, A., Igl, M., Karkus, P., Xu, D., et al.: Storm: Spatio-temporal reconstruction model for large-scale outdoor scenes. arXiv preprint arXiv:2501.00602 (2024)
51. Yang, Z., Gao, X., Zhou, W., Jiao, S., Zhang, Y., Jin, X.: Deformable 3d gaussians for high-fidelity monocular dynamic scene reconstruction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 20331–20341 (2024)
52. Yu, Z., Wang, H., Yang, J., Wang, H., Cao, J., Ji, Z., Sun, M.: Sgd: Street view synthesis with gaussian splatting and diffusion prior. In: 2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV). pp. 3812–3822. IEEE Computer Society (2025)
53. Zhang, B., Ke, L., Harley, A.W., Fragkiadaki, K.: Tapip3d: Tracking any point in persistent 3d geometry. arXiv preprint arXiv:2504.14717 (2025)
54. Zhao, G., Ni, C., Wang, X., Zhu, Z., Zhang, X., Wang, Y., Huang, G., Chen, X., Wang, B., Zhang, Y., et al.: Drivedreamer4d: World models are effective data machines for 4d driving scene representation. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 12015–12026 (2025)
55. Zhou, K., Wang, Y., Chen, G., Chang, X., Beaudouin, G., Zhan, F., Liang, P.P., Wang, M.: Page-4d: Disentangled pose and geometry estimation for 4d perception. arXiv preprint arXiv:2510.17568 (2025)
56. Zhou, X., Lin, Z., Shan, X., Wang, Y., Sun, D., Yang, M.H.: Drivinggaussian: Composite gaussian splatting for surrounding dynamic autonomous driving scenes. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 21634–21643 (2024)
57. Zhu, Z., Wu, Z., Zhu, Z., Zhou, L., Sun, H., Wan, B., Ma, K., Chen, G., Ye, H., Xie, J., et al.: Worldsplat: Gaussian-centric feed-forward 4d scene generation for autonomous driving. arXiv preprint arXiv:2509.23402 (2025)
58. Zhuo, D., Zheng, W., Guo, J., Wu, Y., Zhou, J., Lu, J.: Streaming 4d visual geometry transformer. arXiv preprint arXiv:2507.11539 (2025)
59. Zuo, S., Xie, Z., Zheng, W., Xu, S., Li, F., Jiang, S., Chen, L., Yang, Z.X., Lu, J.: Dvgt: Driving visual geometry transformer. arXiv preprint arXiv:2512.16919 (2025)