MADrive: Memory-Augmented Driving Scene Modeling

Polina Karpikova[*] [†]
Yandex

Daniil Selikhanovych[*]
Yandex Research, HSE University

Kirill Struminsky[*]
Yandex Research, HSE University

Ruslan Musaev
Yandex

Maria Golitsyna
Yandex

Dmitry Baranchuk
Yandex Research

Abstract

Recent advances in scene reconstruction have pushed toward highly realistic modeling of autonomous driving (AD) environments using 3D Gaussian splatting. However, the resulting reconstructions remain closely tied to the original observations and struggle to support photorealistic synthesis of significantly altered or novel driving scenarios. This work introduces MADRIVE*, a memory-augmented reconstruction framework designed to extend the capabilities of existing scene reconstruction methods by replacing observed vehicles with visually similar 3D assets retrieved from a large-scale external memory bank. Specifically, we release* MAD-CARS*, a curated dataset of* $\sim$*70K 360° car videos captured in the wild and present a retrieval module that finds the most similar car instances in the memory bank, reconstructs the corresponding 3D assets from video, and integrates them into the target scene through orientation alignment and relighting. The resulting replacements provide complete multi-view representations of vehicles in the scene, enabling photorealistic synthesis of substantially altered configurations, as demonstrated in our experiments.*

1. Introduction

Modern autonomous driving (AD) systems heavily rely on computer vision and machine learning models trained on large-scale and diverse datasets [8, 9, 20, 46, 61]. However, collecting and annotating such data in the real world is expensive, time-consuming, and often limited by safety and practicality. Driving simulators [55, 66, 74] offer an alternative by generating realistic novel views and rare, safety-critical scenarios that are difficult to record in the real world. Realistic simulation reduces the domain gap between synthetic and real data, improves model robustness, and enables systematic testing of perception and planning models by reproducing failure cases under different conditions. In this work, we reconstruct driving sequences captured by an autonomous vehicle to enable such controllable, photorealistic reenactments for safety testing and data augmentation.

Recent progress in multi-view reconstruction and novel view synthesis [31, 33, 69] has enabled photorealistic reconstruction of complex real-world scenes, forming a strong foundation for realistic driving simulators [74]. Unlike game engine–based environments, such methods preserve the visual properties of real data, reducing domain shift. Modern driving scene reconstruction methods [32, 63, 73] can accurately reproduce observed trajectories and support small viewpoint changes. However, their quality drops when extrapolating far from the observed data, as unseen regions lack geometric and photometric information. As a result, they cannot reliably simulate large trajectory changes such as U-turns or parking maneuvers, where missing observations cause visible artifacts and incomplete geometry.

To overcome these limitations, we replace dynamic objects with high-quality external assets. Traditional computer graphics assets are manually created by artists, making it costly to build a large and diverse library. Instead, we reconstruct assets directly from real-world data. We introduce MAD-CARS, **M**ulti-view **A**uto **D**ataset — a large-scale collection of 360° vehicle videos. The dataset contains about 70,000 car instances covering various makes, models, and colors, providing diverse and realistic inputs for asset reconstruction. Reconstructing assets from real-world data greatly expands their diversity compared to existing public car datasets.

To reconstruct realistic and reusable assets, we develop a car reconstruction pipeline tailored for our in-the-wild captures. A key challenge is adapting assets captured under unknown and scene-specific lighting to new environments. To solve this, we propose a relightable variant of Gaussian splatting that supports physically based rendering under novel illumination. Our optimization scheme uses a generative model [58] to simulate multi-lighting supervision during reconstruction, effectively separating material properties from lighting. This enables high-quality relighting and seamless integration of assets into new scenes.

Building on these components, we present MADRIVE,

[*]Equal contribution.
[†]Email: p.karpikova@yandex.ru

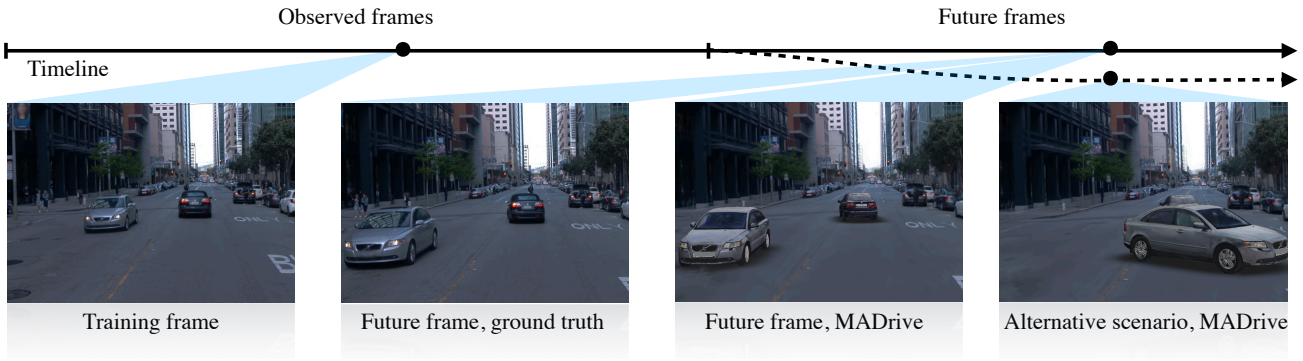

Figure 1. **MADRIVE** reconstructs a 3D driving scene from training frames (Left) and replaces partially observed vehicles in the scene with realistically reconstructed counterparts retrieved from **MAD-CARS**, our novel multi-view auto dataset. MADRIVE enables high-fidelity modeling of future scene views (Middle-left vs. Middle-right) and supports simulation of alternative scenarios, advancing novel-view synthesis in dynamic environments (Right).

a framework that replicates driving sequences by replacing observed vehicles with visually similar reconstructed 3D assets retrieved from a large external dataset. Since driving scenes are only partially observed, pixel-exact reconstruction of vehicle assets is ill-posed; our goal is therefore to produce visually plausible reenactments that preserve the roads, actors, and trajectories of the original scene. Accordingly, we evaluate on downstream perception tasks and distribution-level metrics rather than pixel-level accuracy. Using high-fidelity, relightable assets, MADRIVE preserves realistic appearance even under large trajectory changes and enables rendering of future or alternative frames beyond the original sequence. Quantitative results show that pretrained perception models perform similarly on our rendered frames and real hold-out data, confirming the realism and consistency of our reconstructions for downstream AD tasks.

In summary, our contributions are threefold: (1) a large-scale dataset of in-the-wild 360° vehicle captures for diverse asset reconstruction, (2) a relightable Gaussian splatting pipeline that separates object appearance from illumination for realistic rendering, and (3) an integrated framework that reconstructs, reenacts, and validates driving scenes with controllable trajectories, achieving photorealism and consistency with real-world perception results.

2. Related Work

Dynamic Urban Scene Reconstruction. Recent dynamic 3D scene reconstruction works adopt 3D Gaussian splatting [31] as an efficient and expressive representation [11, 57, 67, 68]. Several approaches, including StreetGS [63], AutoSplat [32], and HUGS [73], apply this representation to driving scenes by decomposing them into static backgrounds and dynamic vehicles placed with tracked 3D bounding boxes. HUGS incorporates optical flow and semantic segmentation to guide optimization and adds realistic shadow modeling, while AutoSplat improves vehicle reconstruction by exploiting bilateral symmetry and better initialization from image-to-3D priors [41]. Other works [12, 75] introduce dynamic Gaussian graphs to handle multiple moving objects of different nature. We refer the reader to Appendix A for a review of earlier NeRF-based scene representation methods. While these methods improve the fidelity of reconstructed urban scenes, they rely solely on observational data to model dynamic objects, making it difficult to capture complete vehicle geometry and appearance under sparse or occluded views.

3D Car Datasets. Several public datasets provide 3D car assets. Early collections such as SRN-Car [10] and Objaverse-Car [15] consist of CAD models that differ notably from real vehicles in texture realism and geometric detail. More recent efforts [16, 70] have focused on real-captured 3D car datasets. MVMC [70] includes 576 cars, each with an average of 10 views. 3DRealCar [16] provides $2,500$ car instances, each with $\sim$200 dense high-resolution RGB-D views. In contrast, MAD-CARS includes $\sim 70,000$ 360° car videos at a comparable resolution and average number of views as 3DRealCar, offering substantially greater generalization and diversity.

NVS with External 3D Car Assets. HUGSim [74] builds a closed-loop AD simulator by inserting 3D car models from 3DRealCar [16] into reconstructed scenes. We instead replace only the vehicles observed in the recording with retrieved matches, producing a close replica of the captured scene for reenactment.

Several approaches leverage CAD models for scene representation [2, 17, 22, 52, 55, 56], though such models often lack photorealistic textures and accurate geometry. To improve realism, some methods perform geometry refinement [17, 52, 55], while UrbanCAD [35] retrieves visually similar CAD models and refines their textures and illumina-

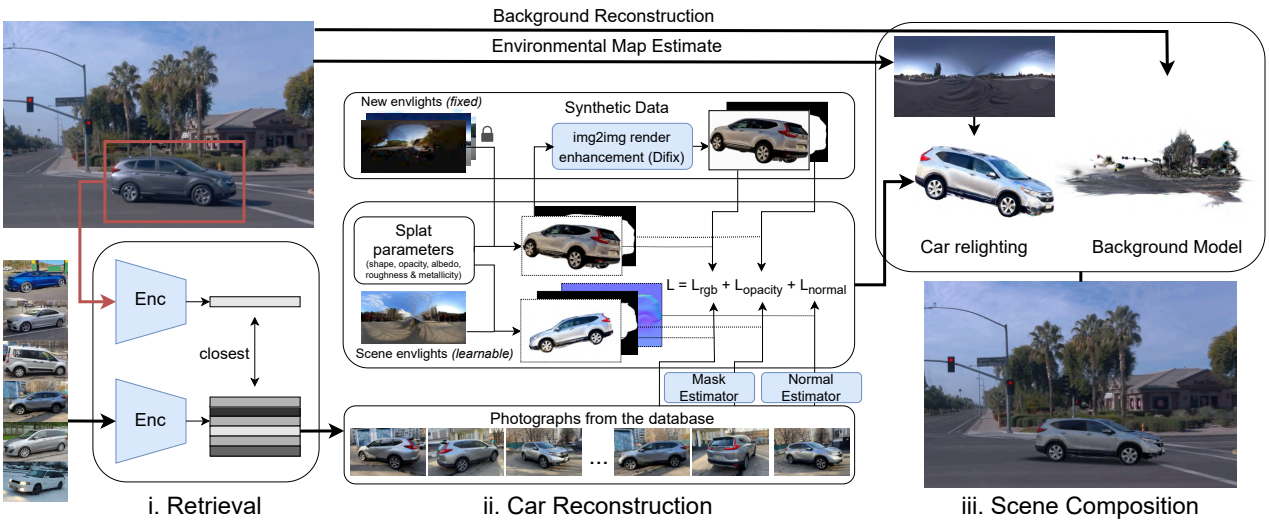

Figure 2. **MADRIVE Overview.** Given an input frame sequence, our retrieval scheme finds similar vehicles in an external database (Left). The 3D reconstruction pipeline then produces detailed vehicle models from the retrieved videos. The vehicles are represented using relightable 2D Gaussian splats. To enable relighting, we generate synthetic novel views under multiple illumination conditions. Opacity masks are applied to remove background splats, and the model geometry is regularized using external normal maps. (Middle). The reconstructed vehicles are adapted to the scene's lighting and composed with the background to produce the overall scene representation (Right).

tion to better match the scene. However, the obtained models still have a noticeable gap in realism and correspondence to actual cars. In contrast, MADRIVE retrieves real car instances from a large-scale dataset spanning diverse brands, materials, and lighting conditions, helping to narrow this realism gap.

Relighting. Exact relighting requires modeling full light transport via ray tracing [28]. NeRF-based relighting methods [26, 47, 53, 72] and recent ray-tracing extensions for Gaussian splats [7, 21, 38, 62] remain too costly for real-time use. Several rasterization-compatible alternatives instead approximate light transport with simplified shading models. LumiGauss [29] uses spherical harmonics [43] but requires real multi-illumination training data and handles only diffuse surfaces. GaussianShader [25] employs a split-sum approximation [30] for specular reflections, and R3DG [19] decomposes materials assuming known illumination. Our approach also uses a physically based shading model but generates synthetic multi-illumination data, removing the need for real multi-light captures or known illumination. For environmental map estimation, we apply DiffusionLight [42] to training frames, avoiding the costly gradient-based optimization of DiPIR [34].

3. Method

In this section, we describe MADRIVE that replaces the vehicles in the scene with visually similar, fully-observed 3D car assets, thereby enabling the prediction of future vehicle appearances following sharp turns or other complex maneuvers. Because driving scenes are only partially observed, recovering the exact geometry and appearance of each vehicle is ill-posed. Instead of attempting pixel-exact reconstruction, we retrieve real-world assets as data-driven priors to complete unobserved regions while preserving instance-level consistency in geometry and appearance. The overview of our method is presented in Figure 2.

3.1. Scene Decomposition and Reconstruction Overview

Following [63], we decompose each driving scene into static and dynamic components. The static part represents the background—ground, surroundings, and sky—while the dynamic part contains all moving vehicles. The static component can be reliably reconstructed from the ego-vehicle's motion and depth sensor data, which provide sufficient parallax to recover 3D structure.

We adapt the approach of Street Gaussians [32] to model the static scene. The surroundings are parameterized using 3D Gaussian splats [31], the ground is represented by horizontal 2D splats, and the sky is placed at an infinite distance and blended during rendering to avoid depth ambiguities.

Dynamic elements include all moving vehicles; however, we treat cars marked as stationary in the metadata as part of the static background for simplicity. Modeling dynamic objects poses two main challenges: handling compound motion and reconstructing objects from incomplete observations (e.g., when only one side of a vehicle is visible). Following prior work [32, 63], we approximate each observed vehicle as a set of static Gaussian splats within its moving 3D bounding box, effectively capturing its motion during training. Both static and dynamic components are initial-

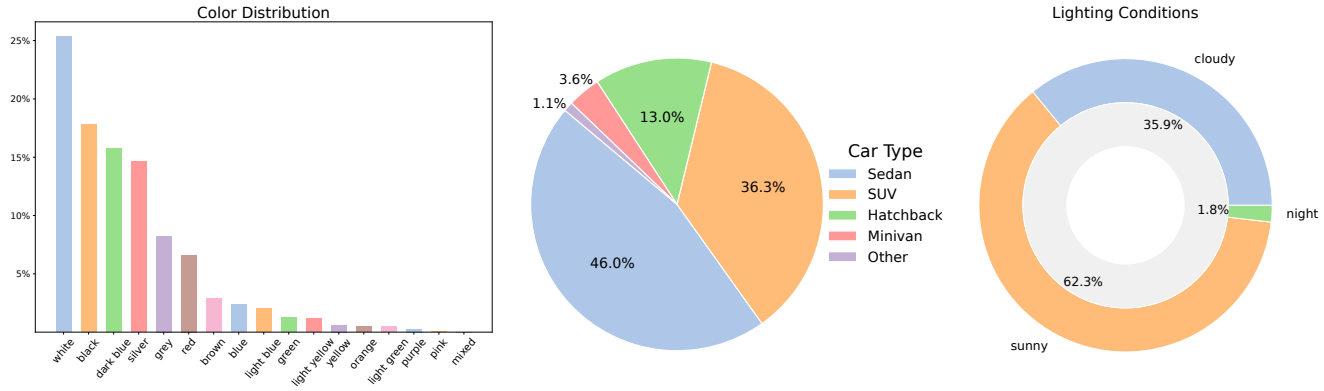

Figure 3. **MAD-CARS Analysis.** Memory-bank statistics on colors (Left), car types (Middle) and lighting conditions (Right).

ized from LiDAR points and jointly optimized using a photometric loss.

During inference, we reuse the static part of the scene. At the same time, we replace moving vehicles with 3D car models extracted from a bank of cars using the retrieval-based approach, described in the next section. Compared to direct reconstruction of dynamic objects, retrieval from a large asset bank offers stronger controllability and real-world diversity, while avoiding artifacts in unobserved regions. This substitution allows obtaining high-quality renders for configurations diverging from the ones observed during training. The computational bottlenecks of our method are static scene reconstruction (≈3 hours on a single GPU) and per-asset reconstruction (≈30 minutes). Retrieval, asset insertion, relighting, and rendering add only minor overhead and run efficiently in comparison.

3.2. Database Retrieval

Our goal is to replace observed vehicles in driving sequences with visually similar 3D assets retrieved from an external database. This subsection outlines our retrieval pipeline and the corresponding database construction.

Retrieval Query Information. For each driving sequence, we project the 3D bounding box of every detected vehicle onto the image plane to obtain a segmentation mask. After discarding small and overlapping masks, we extract cropped images centered on individual cars. Each crop is embedded using SigLIP2 [50], while the vehicle color is estimated with Qwen2.5-VL [64]. The color cue complements the embedding features, which primarily capture brand and type information, improving retrieval accuracy.

Database Collection and Statistics We introduce MAD-CARS, a large-scale collection of in-the-wild multi-view car videos sourced from online car-sale advertisements. The dataset contains $\sim$70,000 car instances, each with $\sim$85 frames at 1920×1080 resolution. It spans roughly 150 brands and covers diverse colors, car types, and various lighting conditions. Figure 3 summarizes the distributions of color, type, and illumination. Each instance includes metadata describing car attributes.

To ensure high-quality reconstruction, we curate the dataset by filtering out frames and instances that degrade multi-view consistency. Specifically, we remove low-quality or overly dark frames using CLIP-IQA [54], and employ Qwen2.5-VL [64] to detect persistent occlusions that obscure parts of the vehicle across most views—such as nearby grass, bushes, or fences—as well as to filter out interior and obstructed shots. Further data-collection details are provided in Appendix B.

To retrieve a matching asset, we first pre-select candidates with similar color and then identify the closest match in the embedding space. The selected instance is reconstructed into a 3D model using its associated multi-view image set. The reconstruction pipeline is detailed in the following section.

3.3. Car Reconstruction Details

Relightable Car Models. We begin by specifying the representation used to model vehicles. By default, Gaussian splatting models the entangled radiance observed in the training frames, implicitly coupling surface reflectance and illumination. In our setup, however, we need to explicitly separate lighting and material effects to enable model insertion into environments with different illumination. To this end, we adopt a relighting strategy based on physically-based shading [6].

We use a two-dimensional modification of Gaussian splats [24], which approximates the 3D model with a collection of flat Gaussian splats. Each splat is parameterized by its location $\mu \in \mathbb{R}^3$, orientation matrix $R \in SO(3)$, transparency $\alpha \in \mathbb{R}$, and *two* scale parameters $\sigma_x, \sigma_y \in \mathbb{R}$. Unlike 3D splats, 2D splats have well-defined surface normals $\mathbf{n} = \mathbf{n}(R)$, which are essential for surface relighting.

To disentangle scene lighting from surface materials, we adopt the lighting model from [39] for each splat. The model assumes distant illumination with incident radiance

Figure 4. **Qualitative comparison** of MADRIVE with non-retrieval-based driving scene reconstruction methods. Reconstruction of the training views (Top). Reconstruction of the hold-out (future) views (Bottom).

$L_i(\omega_i)$ and defines the outgoing radiance in direction ω_o according to the rendering equation [28]:

$$L(\omega_o) = \int_\Omega L_i(\omega_i) f(\omega_i, \omega_o)(\omega_i \cdot \mathbf{n}) \mathrm{d}\omega_i, \tag{1}$$

where $f(\omega_i, \omega_o)$ is the surface BSDF and the integration is taken over the hemisphere Ω around the surface point. The environment lighting L_i is parameterized as a high-resolution cubemap. Following [39], we parameterize each splat's BSDF using the Cook–Torrance shading model [14], with appearance defined by albedo $c \in \mathbb{R}^3$, roughness $r \in \mathbb{R}$, and metallicity $m \in \mathbb{R}$.

Finally, to avoid the cost of directly evaluating Eq. 1, we employ the differentiable split-sum approximation from [39], which allows us to jointly infer incident radiance and splat BSDF parameters during optimization.

Reconstruction Algorithm. Next, we specify the details of the reconstruction algorithm used for the representation above.

For a rendered frame I_i and the ground truth frame $\hat{I}_i$, our objective consists of image-based loss $\mathcal{L}_{\text{rgb}} = \mathcal{L}_1(I_i, \hat{I}_i) + \mathcal{L}_{SSIM}(I_i, \hat{I}_i)$ along with several regularizers. To exclude unnecessary background objects from the model, we generate masks $\hat{M}_i(x, y) = [\hat{I}_i(x, y) \textit{ is part of a car}]$ with Mask2Former [13] to indicate pixels that belong to the model. Our opacity loss promotes high transparency outside of car pixels $\mathcal{L}_{opacity} = \sum_{x,y}(1 - \hat{M}_i) \cdot T_i$, where T_i is the transparency map of the rendered frame. In our model, proper relighting requires accurate surface normals, so we additionally estimate normal maps $\hat{N}_i = n(\hat{I}_i)$ with a NormalCrafter model [4] and use the estimates to regularize Gaussian orientations. For the rendered normal maps N_i, the regularizer is $\mathcal{L}_{normal} = \sum_{x,y} \hat{M}_i \cdot (1 - N_i^T \hat{N}_i)$. The resulting objective is

$$\begin{aligned}\mathcal{L}_{gt}(I_i, \hat{I}_i) = \mathcal{L}_{rgb}(I_i, \hat{I}_i) + \lambda_{opacity}\mathcal{L}_{opacity}(I_i, \hat{I}_i) + \\ \lambda_{normal}\mathcal{L}_{normal}(I_i, \hat{I}_i).\end{aligned} \tag{2}$$

To address the limited lighting variability in our real captures, we additionally introduce a generative augmentation strategy that extends the Difix framework [58] beyond its original purpose. While Difix was originally proposed to enhance the photorealism of rendered novel views, we repurpose it to simulate appearance under diverse illumination conditions, effectively approximating multi-illumination supervision.

Concretely, we render the reconstructed model from random novel viewpoints and relight each render using randomly sampled environmental maps. These low-quality renders are then refined with Difix to produce synthetic enhanced frames $\tilde{I}_i$ that emulate novel realistic lighting. By pairing each $\tilde{I}_i$ with a physically based render I_i under the same illumination, we compute the objective in Eq. 2, using the α-mask of $\tilde{I}_i$ in $\mathcal{L}_{opacity}$ and omitting $\mathcal{L}_{normal}$ term.

This augmentation effectively disentangles illumination and material properties, allowing our model to generalize across lighting conditions that were never observed in the training data. We mix synthetic and real frames in equal proportion, introducing synthetic samples after the initial 10k iterations and refreshing them every 2.5k steps for 20k additional iterations.

After the Gaussian splatting reconstruction, we apply several postprocessing steps to obtain the final car asset. We remove stray splats that do not belong to the depicted car by computing instance segmentation [27] and discarding those that lie behind the training cameras or project outside the car mask in most views, using a soft threshold to handle segmentation errors. The cleaned point cloud is then oriented along its principal components, and the front direction is estimated using an orientation model [45].

3.4. Car Insertion and Relighting

After reconstructing the vehicle models with a standardized orientation, we integrate them into the captured scene. We first estimate each car's orientation by aligning the asset's bounding box with the detected vehicle bounding box in the scene. However, directly matching the bounding boxes

in scale and position often results in noticeable misalignment. To obtain accurate placement, we refine the transformation using the Iterative Closest Point (ICP) algorithm [3], aligning the reconstructed asset to the corresponding LiDAR point cloud and deriving the final scale and location from this alignment.

To ensure consistent lighting between the inserted assets and the reconstructed scene, we estimate the scene's environmental map to compute the outgoing radiance of each asset in Eq. 1. Since the Waymo dataset [46] lacks full 360° camera coverage and is captured in low dynamic range, we approximate the full high dynamic range environmental map with DiffusionLight [42] using the last training frame from the frontal camera. We further scale the estimated environmental map to minimize tone discrepancies between the last training frame and the rendered image. Finally, to enhance visual realism, we add a shadow beneath the car, modeled as a black semi-transparent plane composed of 2D splats positioned right below the wheels.

4. Experiments

In the following, we present the evaluation of MADRIVE. Section 4.1 describes the experimental setup, followed by the main results in Section 4.2. Finally, Section 4.3 analyzes the retrieval, car reconstruction components, and relighting in the scene.

4.1. Evaluation Setup

Scene Reconstruction Dataset. We reconstruct driving scenes from the Waymo Open Motion Dataset [18]. We select 12 challenging sequences featuring multiple vehicles, complex driving maneuvers, and diverse lighting conditions. Each scene is manually segmented into training and evaluation clips. In our experiments, we simultaneously use videos from frontal and two side cameras to capture a wide field of view and track cars moving across the scene. More evaluation setup details are provided in Appendix H.

Scene Extrapolation with Novel View Synthesis. For our evaluation, we selected driving scenes involving U-turns, intersection crossings, and parking departures — common accident scenarios that expose vehicles from diverse viewpoints and pose significant challenges for reconstruction. Each sequence was manually divided into training and testing subsets at the midpoint of the maneuver. We use the whole sequence to reconstruct the background and then remove the cars using the annotated bounding boxes in the Waymo dataset. Car reconstruction relies solely on the training portion, while the remaining frames are reserved for evaluating scene reconstruction quality.

Our goal is to generate realistic novel views by extrapolating beyond the observed data. Specifically, we insert the reconstructed car models into the background according to location and orientation specified by the bounding boxes on the holdout sequence. This setup intentionally tests the model under configurations that differ from the training views, while ensuring no test data leaks into the reconstruction process.

Baselines. We compare MADRIVE with the scene reconstruction Gaussian splatting-based methods that were previously considered for novel view synthesis: Street-Gaussians (SG) [63], AutoSplat [32] (our implementation), and HUGS [73]. Details on training and evaluation of baselines are given in Appendix J.

4.2. Main Experiments

Qualitative Evaluation. First, we provide visual scene reconstruction results for qualitative analysis. In Figure 4, we compare renderings on the training and hold-out frames. While SG, AutoSplat, and HUGS reproduce the training frames with high accuracy, their reconstructions tend to break down under novel viewpoints, causing vehicles to fall apart or distort. In contrast, our method, though slightly less precise on training frames, demonstrates substantially greater robustness to unseen configurations. Additional visual examples are shown in Figures 11, 12. We also provide the visualizations with modified trajectories in Figure 13.

Table 1. Comparison in terms of tracking and segmentation metrics.

Model	MOTA ↑	MOTP ↓	IDF1 ↑	Segmentation IoU ↑
Street-GS [63]	0.654	**0.105**	0.776	0.556
HUGS [73]	0.556	0.221	0.699	0.333
AutoSplat* [32]	0.589	0.154	0.716	0.489
MADRIVE (Ours)	**0.841**	0.138	**0.913**	**0.818**

*Denotes our reimplementation.

Quantitative Evaluation. In our main experiments, we evaluate tracking and segmentation performance on synthesized **test** frames. Specifically, we apply state-of-the-art tracking and segmentation models to both synthesized and ground truth frames and compare their outputs using established metrics for each task. For tracking, we use BotSort [1] with a YOLOv8n backbone and report Multiple Object Tracking Accuracy (MOTA↑), Multiple Object Tracking Precision (MOTP↓), and the identity F1 score (IDF1) [36]. For segmentation, we compute the average intersection-over-union (IoU) using instance segmentation masks predicted by Mask2Former [13].

Table 1 compares MADRIVE against baseline approaches. MADRIVE achieves substantially higher scores in two tracking metrics (MOTA and IDF1) and in the segmentation metric (IoU), confirming improved scene consistency and object reconstruction quality. This observation is also supported by the visual examples provided in Figure 4.

Figure 5. **Qualitative comparison of reconstructed vehicle assets** based on queries from KITTI-360 (reference and rotated viewpoints). MADRIVE, trained on the MAD-CARS dataset, produces 3D assets that better preserve shape consistency and visual realism across views compared to CAD-based and image-to-3D generative baselines.

The lower MOTP score of MADRIVE compared to Street-GS arises from Street-GS's better initial alignment in early test frames; however, later frames—where Street-GS tracking often fails—are excluded from the MOTP calculation. Per-scene results for all 12 sequences are provided in Appendix I, and the choice of reference masks used in the evaluation protocol is discussed in Appendix G.

4.3. Ablation Study

Retrieval. Here, we evaluate the retrieval module in isolation to assess how accurately the retrieved cars correspond to the original vehicles in the scene.

We compare retrieval performance on MAD-CARS against 3DRealCars [16], a high-quality publicly available dataset containing 2,500 car assets. To evaluate retrieval quality, we first compute the average L2 distance between each car image from the driving scene and its nearest neighbor in the memory bank, using SigLIP2 [50] as the image feature extractor. Then, we provide accuracy obtained with the Qwen2.5-VL-32B-Instruct model, which compares cars based on brand, model, color, and car type. For a fair comparison, we do not use the color filtering in this experiment.

Table 2 reports retrieval accuracy across different attributes, along with the average L2 distance to the closest instance. Cars retrieved from MAD-CARS more closely match the driving scene vehicles, which we attribute to the dataset's larger scale and diversity.

Notably, Table 2 highlights that retrieval based solely on feature embeddings often ignores car color, despite its importance for accurate vehicle replacement. A similar limitation has been observed with non-vision-language encoders such as DINOv2 [40]. Additional results in Appendix C show that applying a color-based pre-filtering improves color consistency between the retrieved and target vehicles.

Table 2. Retrieval performance w/o color filtering in terms of accuracy on the car brand, model, color and type and the distance to the closest instance for the MAD-CARS and 3DRealCar [16] datasets. MAD-CARS enables more accurate retrieval of cars across all attributes.

Dataset	Brand ↑	Model ↑	Color ↑	Car Type ↑	Distance ↓
3DRealCars	0.626	0.503	0.508	0.888	0.502
MAD-CARS	0.750	0.663	0.533	0.913	0.445

Car reconstruction. We provide a qualitative comparison with other car reconstruction approaches in Figure 5, where we visualized reconstruction alternatives. Given a query frame from the KITTI dataset, we compared the proposed approach with three alternatives: matching with a car model from a CAD dataset (UrbanCAD, [35]), and running cutting-edge image-to-3D models (SAM3D [48], Amodal3R [59]). Even though the latter closely matches the query frame, the second view indicates a subpar geometry recovery. Compared to other methods, we see that the diversity of our dataset allows MADRIVE to obtain models that closely match the query frame in terms of appearance (e.g., color, shape) and realism.

We also quantitatively compare the realism of reconstructed cars for the retrieval-based approach of MADRIVE and the recent image-to-3D generative models, including Amodal3R [59], TRELLIS [60], and SAM3D [48]. For this,

Figure 6. Qualitative ablation of reconstruction components, progressively adding each regularizer to the previous configuration. Starting without regularization (a), the reconstruction shows shape and texture artifacts with uneven edges. Adding opacity regularization (b) improves edge quality, while normal regularization (c) enhances surface smoothness. Incorporating synthetic data (d) further refines the results, and training with synthetic frames under varying lighting (e) helps disentangle illumination from object color, producing cleaner albedo and overall more consistent reconstructions.

we collected random crops using a hold-out set of real cars from the Waymo dataset and retrieved MAD-CARS images with MADRIVE. For image-to-3D models, we provide the input image from retrieved images of MAD-CARS. Then we rendered 3D models from 360 degrees. In Table 3, we provide FID [23] and KID [5] between the set of renderings for the reconstructed models with MADRIVE and image-to-3D models and the set of real images of hold-out cars from MAD-CARS. The results support our claim that the retrieval-augmented approach yields cars with better realism. In Appendix D, we detail the evaluation setup and provide visual results of MADRIVE and image-to-3D models.

Table 3. Quantitative assessment of car model quality on retrieved images.

Metrics	Amodal3R	TRELLIS	SAM3D	MADrive
FID ↓	43.33	42.93	53.13	**35.50**
KID$\times 10^3$ ↓	26.15	25.32	32.54	**18.08**

We further ablate the reconstruction components in Figure 6, progressively adding each regularizer. Opacity and normal regularization (b, c) improve edge quality and surface smoothness, while Difix-enhanced synthetic views (d) further refine geometry. Training with synthetic frames under varying lighting (e) disentangles illumination from object color, yielding more uniform albedo and cleaner reconstructions overall.

Relighting. We performed a qualitative comparison to evaluate the impact of the proposed relighting scheme. For several scenes, we reconstructed frames both with and without the relighting module. As shown in Figure 7, the relighting module effectively adjusts model colors to the surrounding illumination, reducing visual inconsistencies and making the inserted vehicles appear more naturally integrated into the scene. An alternative would be to apply generative relighting models; however, these add pipeline complexity and may lack temporal consistency. We compare with such approaches in Appendix E and find that our lightweight module offers a better trade-off between quality and controllability.

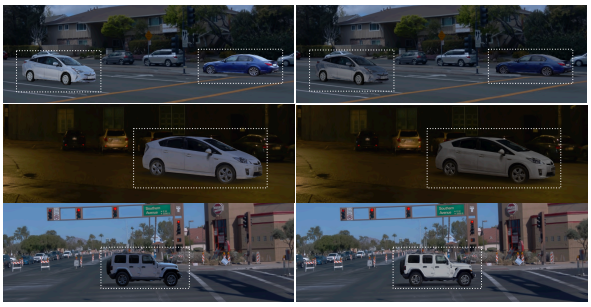

Figure 7. **Relighting ablation.** Rendered hold-out frames without (Left) and with (Right) relighting.

5. Conclusion

This work presents MADRIVE, a driving scene modeling framework that produces visually plausible and controllable reenactments of real driving scenarios, even under significantly altered vehicle trajectories. Powered by MAD-CARS, our large-scale multi-view car dataset, MADRIVE replaces dynamic vehicles with similar instances from the database, making a step towards modeling multiple potential outcomes for safety-critical analysis of autonomous driving systems.

Although our generated future frames look promising, they still differ from the ground truth, as discussed in Appendix K. At the moment, we only use the future frame to create the retrieval query, but it could also help adapt the inserted asset to better match the observed vehicle in future work. Another direction is to include advanced relighting methods that better capture reflections and shadows, improving realism under new lighting conditions.

References

[1] Nir Aharon, Roy Orfaig, and Ben-Zion Bobrovsky. Bot-sort: Robust associations multi-pedestrian tracking. *arXiv preprint arXiv:2206.14651*, 2022. 6

[2] Armen Avetisyan, Manuel Dahnert, Angela Dai, Manolis Savva, Angel X Chang, and Matthias Nießner. Scan2cad: Learning cad model alignment in rgb-d scans. In *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition*, pages 2614–2623, 2019. 2

[3] Paul Besl and H.D. McKay. A method for registration of 3-d shapes. ieee trans pattern anal mach intell. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 14:239–256, 1992. 6

[4] Yanrui Bin, Wenbo Hu, Haoyuan Wang, Xinya Chen, and Bing Wang. Normalcrafter: Learning temporally consistent normals from video diffusion priors. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 8330–8339, 2025. 5

[5] Mikołaj Bińkowski, Dougal J. Sutherland, Michael Arbel, and Arthur Gretton. Demystifying MMD GANs. In *International Conference on Learning Representations*, 2018. 8, 13

[6] Brent Burley and Walt Disney Animation Studios. Physically-based shading at disney. In *ACM SIGGRAPH*, pages 1–7. vol. 2012, 2012. 4

[7] Krzysztof Byrski, Marcin Mazur, Jacek Tabor, Tadeusz Dziarmaga, Marcin Kądziołka, Dawid Baran, and Przemysław Spurek. Raysplats: Ray tracing based gaussian splatting. *arXiv preprint arXiv:2501.19196*, 2025. 3

[8] Yohann Cabon, Naila Murray, and Martin Humenberger. Virtual kitti 2. *arXiv preprint arXiv:2001.10773*, 2020. 1

[9] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11621–11631, 2020. 1

[10] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015. 2

[11] Yurui Chen, Chun Gu, Junzhe Jiang, Xiatian Zhu, and Li Zhang. : Dynamic urban scene reconstruction and real-time rendering. *International Journal of Computer Vision*, 2026. 2

[12] Ziyu Chen, Jiawei Yang, Jiahui Huang, Riccardo de Lutio, Janick Martinez Esturo, Boris Ivanovic, Or Litany, Zan Gojcic, Sanja Fidler, Marco Pavone, Li Song, and Yue Wang. Omnire: Omni urban scene reconstruction. In *The Thirteenth International Conference on Learning Representations*, 2025. 2

[13] Bowen Cheng, Ishan Misra, Alexander G. Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1280–1289, 2022. 5, 6

[14] Robert L Cook and Kenneth E. Torrance. A reflectance model for computer graphics. *ACM Transactions on Graphics (ToG)*, 1(1):7–24, 1982. 5

[15] Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13142–13153, 2023. 2

[16] Xiaobiao Du, Yida Wang, Haiyang Sun, Zhuojie Wu, Hongwei Sheng, Shuyun Wang, Jiaying Ying, Ming Lu, Tianqing Zhu, Kun Zhan, and Xin Yu. 3drealcar: An in-the-wild rgb-d car dataset with 360-degree views. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 26488–26498, 2025. 2, 7

[17] Francis Engelmann, Jörg Stückler, and Bastian Leibe. Samp: shape and motion priors for 4d vehicle reconstruction. In *2017 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 400–408. IEEE, 2017. 2

[18] Scott Ettinger, Shuyang Cheng, Benjamin Caine, Chenxi Liu, Hang Zhao, Sabeek Pradhan, Yuning Chai, Ben Sapp, Charles R Qi, Yin Zhou, et al. Large scale interactive motion forecasting for autonomous driving: The waymo open motion dataset. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9710–9719, 2021. 6, 13

[19] Jian Gao, Chun Gu, Youtian Lin, Zhihao Li, Hao Zhu, Xun Cao, Li Zhang, and Yao Yao. Relightable 3d gaussians: Realistic point cloud relighting with brdf decomposition and ray tracing. In *European Conference on Computer Vision*, pages 73–89. Springer, 2024. 3

[20] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3354–3361, 2012. 1

[21] Shrisudhan Govindarajan, Daniel Rebain, Kwang Moo Yi, and Andrea Tagliasacchi. Radiant foam: Real-time differentiable ray tracing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4135–4145, 2025. 3

[22] Can Gümeli, Angela Dai, and Matthias Nießner. Roca: Robust cad model retrieval and alignment from a single image. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4012–4021, 2022. 2

[23] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2017. 8, 13

[24] Binbin Huang, Zehao Yu, Anpei Chen, Andreas Geiger, and Shenghua Gao. 2d gaussian splatting for geometrically accurate radiance fields. In *ACM SIGGRAPH 2024 conference papers*, pages 1–11, 2024. 4

[25] Yingwenqi Jiang, Jiadong Tu, Yuan Liu, Xifeng Gao, Xiaoxiao Long, Wenping Wang, and Yuexin Ma. Gaussianshader: 3d gaussian splatting with shading functions for reflective surfaces. In *Proceedings of the IEEE/CVF Confer-*

ence on Computer Vision and Pattern Recognition, pages 5322–5332, 2024. 3

[26] Haian Jin, Isabella Liu, Peijia Xu, Xiaoshuai Zhang, Songfang Han, Sai Bi, Xiaowei Zhou, Zexiang Xu, and Hao Su. Tensoir: Tensorial inverse rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 165–174, 2023. 3

[27] Glenn Jocher and Jing Qiu. Ultralytics yolo11, 2024. 5, 13

[28] James T Kajiya. The rendering equation. In *Proceedings of the 13th annual conference on Computer graphics and interactive techniques*, pages 143–150, 1986. 3, 5

[29] Joanna Kaleta, Kacper Kania, Tomasz Trzciński, and Marek Kowalski. Lumigauss: Relightable gaussian splatting in the wild. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 1–10, 2025. 3

[30] Brian Karis and Epic Games. Real shading in unreal engine 4. *Proc. Physically Based Shading Theory Practice*, 4(3):1, 2013. 3

[31] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139–1, 2023. 1, 2, 3

[32] Mustafa Khan, Hamidreza Fazlali, Dhruv Sharma, Tongtong Cao, Dongfeng Bai, Yuan Ren, and Bingbing Liu. Autosplat: Constrained gaussian splatting for autonomous driving scene reconstruction. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 8315–8321, 2025. 1, 2, 3, 6

[33] Shakiba Kheradmand, Daniel Rebain, Gopal Sharma, Weiwei Sun, Yang-Che Tseng, Hossam Isack, Abhishek Kar, Andrea Tagliasacchi, and Kwang Moo Yi. 3d gaussian splatting as markov chain monte carlo. *Advances in Neural Information Processing Systems*, 37:80965–80986, 2024. 1

[34] Ruofan Liang, Zan Gojcic, Merlin Nimier-David, David Acuna, Nandita Vijaykumar, Sanja Fidler, and Zian Wang. Photorealistic object insertion with diffusion-guided inverse rendering. In *European Conference on Computer Vision*, pages 446–465. Springer, 2024. 3

[35] Yichong Lu, Yichi Cai, Shangzhan Zhang, Hongyu Zhou, Haoji Hu, Huimin Yu, Andreas Geiger, and Yiyi Liao. Urbancad: Towards highly controllable and photorealistic 3d vehicles for urban scene simulation. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 27519–27530, 2025. 2, 7

[36] Anton Milan, Laura Leal-Taixé, Ian Reid, Stefan Roth, and Konrad Schindler. Mot16: A benchmark for multi-object tracking. *arXiv preprint arXiv:1603.00831*, 2016. 6

[37] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *Computer Vision – ECCV 2020*, pages 405–421, Cham, 2020. Springer International Publishing. 13

[38] Nicolas Moenne-Loccoz, Ashkan Mirzaei, Or Perel, Riccardo de Lutio, Janick Martinez Esturo, Gavriel State, Sanja Fidler, Nicholas Sharp, and Zan Gojcic. 3d gaussian ray tracing: Fast tracing of particle scenes. *ACM Transactions on Graphics (TOG)*, 43(6):1–19, 2024. 3

[39] Jacob Munkberg, Jon Hasselgren, Tianchang Shen, Jun Gao, Wenzheng Chen, Alex Evans, Thomas Müller, and Sanja Fidler. Extracting triangular 3d models, materials, and lighting from images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8280–8290, 2022. 4, 5

[40] Maxime Oquab, Timothée Darcet, Theo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Russell Howes, Po-Yao Huang, Hu Xu, Vasu Sharma, Shang-Wen Li, Wojciech Galuba, Mike Rabbat, Mido Assran, Nicolas Ballas, Gabriel Synnaeve, Ishan Misra, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. Dinov2: Learning robust visual features without supervision, 2023. 7

[41] Dario Pavllo, David Joseph Tan, Marie-Julie Rakotosaona, and Federico Tombari. Shape, pose, and appearance from a single image via bootstrapped radiance field inversion. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 2

[42] Pakkapon Phongthawee, Worameth Chinchuthakun, Nontaphat Sinsunthithet, Varun Jampani, Amit Raj, Pramook Khungurn, and Supasorn Suwajanakorn. Diffusionlight: Light probes for free by painting a chrome ball. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 98–108, 2024. 3, 6

[43] Ravi Ramamoorthi and Pat Hanrahan. An efficient representation for irradiance environment maps. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 497–500, 2001. 3

[44] Yossi Rubner, Carlo Tomasi, and Leonidas J Guibas. The earth mover's distance as a metric for image retrieval. *International journal of computer vision*, 40(2):99–121, 2000. 14

[45] Christopher Scarvelis, David Benhaim, and Paul Zhang. Orient anything. *arXiv preprint arXiv:2410.02101*, 2024. 5

[46] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurélien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, Vijay Vasudevan, Wei Han, Jiquan Ngiam, Hang Zhao, Aleksei Timofeev, Scott Ettinger, Maxim Krivokon, Amy Gao, Aditya Joshi, Yu Zhang, Jonathon Shlens, Zhifeng Chen, and Dragomir Anguelov. Scalability in perception for autonomous driving: Waymo open dataset. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2443–2451, 2020. 1, 6, 18

[47] Jiapeng Tang, Matthew Lavine, Dor Verbin, Stephan J Garbin, Matthias Nießner, Ricardo Martin Brualla, Pratul P Srinivasan, and Philipp Henzler. Rogr: Relightable 3d objects using generative relighting. *arXiv preprint arXiv:2510.03163*, 2025. 3

[48] SAM 3D Team, Xingyu Chen, Fu-Jen Chu, Pierre Gleize, Kevin J Liang, Alexander Sax, Hao Tang, Weiyao Wang, Michelle Guo, Thibaut Hardin, Xiang Li, Aohan Lin, Jiawei Liu, Ziqi Ma, Anushka Sagar, Bowen Song, Xiaodong Wang, Jianing Yang, Bowen Zhang, Piotr Dollár, Georgia Gkioxari, Matt Feiszli, and Jitendra Malik. Sam 3d: 3dfy anything in images. 2025. 7, 13

[49] Adam Tonderski, Carl Lindström, Georg Hess, William Ljungbergh, Lennart Svensson, and Christoffer Petersson. Neurad: Neural rendering for autonomous driving. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14895–14904, 2024. 13

[50] Michael Tschannen, Alexey Gritsenko, Xiao Wang, Muhammad Ferjad Naeem, Ibrahim Alabdulmohsin, Nikhil Parthasarathy, Talfan Evans, Lucas Beyer, Ye Xia, Basil Mustafa, Olivier Hénaff, Jeremiah Harmsen, Andreas Steiner, and Xiaohua Zhai. Siglip 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features. *arXiv preprint arXiv:2502.14786*, 2025. 4, 7

[51] Haithem Turki, Jason Y. Zhang, Francesco Ferroni, and Deva Ramanan. Suds: Scalable urban dynamic scenes. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12375–12385, 2023. 13

[52] Mikaela Angelina Uy, Jingwei Huang, Minhyuk Sung, Tolga Birdal, and Leonidas Guibas. Deformation-aware 3d model embedding and retrieval. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part VII 16*, pages 397–413. Springer, 2020. 2

[53] Dor Verbin, Pratul P Srinivasan, Peter Hedman, Ben Mildenhall, Benjamin Attal, Richard Szeliski, and Jonathan T Barron. Nerf-casting: Improved view-dependent appearance with consistent reflections. In *SIGGRAPH Asia 2024 Conference Papers*, pages 1–10, 2024. 3

[54] Jianyi Wang, Kelvin CK Chan, and Chen Change Loy. Exploring clip for assessing the look and feel of images. In *AAAI*, 2023. 4, 13

[55] Jingkang Wang, Sivabalan Manivasagam, Yun Chen, Ze Yang, Ioan Andrei Bârsan, Anqi Joyce Yang, Wei-Chiu Ma, and Raquel Urtasun. Cadsim: Robust and scalable in-the-wild 3d reconstruction for controllable sensor simulation. In *Proceedings of The 6th Conference on Robot Learning*, pages 630–642. PMLR, 2023. 1, 2

[56] Yuxi Wei, Zi Wang, Yifan Lu, Chenxin Xu, Changxing Liu, Hao Zhao, Siheng Chen, and Yanfeng Wang. Editable scene simulation for autonomous driving via collaborative llm-agents. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15077–15087, 2024. 2

[57] Guanjun Wu, Taoran Yi, Jiemin Fang, Lingxi Xie, Xiaopeng Zhang, Wei Wei, Wenyu Liu, Qi Tian, and Xinggang Wang. 4d gaussian splatting for real-time dynamic scene rendering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 20310–20320, 2024. 2

[58] Jay Zhangjie Wu, Yuxuan Zhang, Haithem Turki, Xuanchi Ren, Jun Gao, Mike Zheng Shou, Sanja Fidler, Zan Gojcic, and Huan Ling. Difix3d+: Improving 3d reconstructions with single-step diffusion models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 26024–26035, 2025. 1, 5

[59] Tianhao Wu, Chuanxia Zheng, Frank Guan, Andrea Vedaldi, and Tat-Jen Cham. Amodal3r: Amodal 3d reconstruction from occluded 2d images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9181–9193, 2025. 7, 13

[60] Jianfeng Xiang, Zelong Lv, Sicheng Xu, Yu Deng, Ruicheng Wang, Bowen Zhang, Dong Chen, Xin Tong, and Jiaolong Yang. Structured 3d latents for scalable and versatile 3d generation. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 21469–21480, 2025. 7, 13

[61] Pengchuan Xiao, Zhenlei Shao, Steven Hao, Zishuo Zhang, Xiaolin Chai, Judy Jiao, Zesong Li, Jian Wu, Kai Sun, Kun Jiang, et al. Pandaset: Advanced sensor suite dataset for autonomous driving. In *2021 IEEE international intelligent transportation systems conference (ITSC)*, pages 3095–3101. IEEE, 2021. 1

[62] Tao Xie, Xi Chen, Zhen Xu, Yiman Xie, Yudong Jin, Yujun Shen, Sida Peng, Hujun Bao, and Xiaowei Zhou. Envgs: Modeling view-dependent appearance with environment gaussian. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5742–5751, 2025. 3

[63] Yunzhi Yan, Haotong Lin, Chenxu Zhou, Weijie Wang, Haiyang Sun, Kun Zhan, Xianpeng Lang, Xiaowei Zhou, and Sida Peng. Street gaussians: Modeling dynamic urban scenes with gaussian splatting. In *European Conference on Computer Vision*, pages 156–173. Springer, 2024. 1, 2, 3, 6

[64] An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, et al. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*, 2024. 4, 13

[65] Jiawei Yang, Boris Ivanovic, Or Litany, Xinshuo Weng, Seung Wook Kim, Boyi Li, Tong Che, Danfei Xu, Sanja Fidler, Marco Pavone, and Yue Wang. EmerneRF: Emergent spatial-temporal scene decomposition via self-supervision. In *The Twelfth International Conference on Learning Representations*, 2024. 13

[66] Ze Yang, Yun Chen, Jingkang Wang, Sivabalan Manivasagam, Wei-Chiu Ma, Anqi Joyce Yang, and Raquel Urtasun. Unisim: A neural closed-loop sensor simulator. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1389–1399, 2023. 1

[67] Ziyi Yang, Xinyu Gao, Wen Zhou, Shaohui Jiao, Yuqing Zhang, and Xiaogang Jin. Deformable 3d gaussians for high-fidelity monocular dynamic scene reconstruction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 20331–20341, 2024. 2

[68] Zeyu Yang, Hongye Yang, Zijie Pan, and Li Zhang. Real-time photorealistic dynamic scene representation and rendering with 4d gaussian splatting. In *The Twelfth International Conference on Learning Representations*, 2024. 2

[69] Zehao Yu, Anpei Chen, Binbin Huang, Torsten Sattler, and Andreas Geiger. Mip-splatting: Alias-free 3d gaussian splatting. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 19447–19456, 2024. 1

[70] Jason Zhang, Gengshan Yang, Shubham Tulsiani, and Deva Ramanan. Ners: Neural reflectance surfaces for sparse-view 3d reconstruction in the wild. *Advances in Neural Information Processing Systems*, 34:29835–29847, 2021. 2

[71] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Scaling in-the-wild training for diffusion-based illumination harmonization and editing by imposing consistent light transport. In *The Thirteenth International Conference on Learning Representations*, 2025. 13

[72] Xiaoming Zhao, Pratul Srinivasan, Dor Verbin, Keunhong Park, Ricardo Martin Brualla, and Philipp Henzler. Illuminerf: 3d relighting without inverse rendering. *Advances in Neural Information Processing Systems*, 37:42593–42617, 2024. 3

[73] Hongyu Zhou, Jiahao Shao, Lu Xu, Dongfeng Bai, Weichao Qiu, Bingbing Liu, Yue Wang, Andreas Geiger, and Yiyi Liao. Hugs: Holistic urban 3d scene understanding via gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 21336–21345, 2024. 1, 2, 6

[74] Hongyu Zhou, Longzhong Lin, Jiabao Wang, Yichong Lu, Dongfeng Bai, Bingbing Liu, Yue Wang, Andreas Geiger, and Yiyi Liao. HUGSIM: A Real-Time, Photo-Realistic and Closed-Loop Simulator for Autonomous Driving . *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 48 (04):4673–4691, 2026. 1, 2

[75] Xiaoyu Zhou, Zhiwei Lin, Xiaojun Shan, Yongtao Wang, Deqing Sun, and Ming-Hsuan Yang. Drivinggaussian: Composite gaussian splatting for surrounding dynamic autonomous driving scenes. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 21634–21643, 2024. 2