

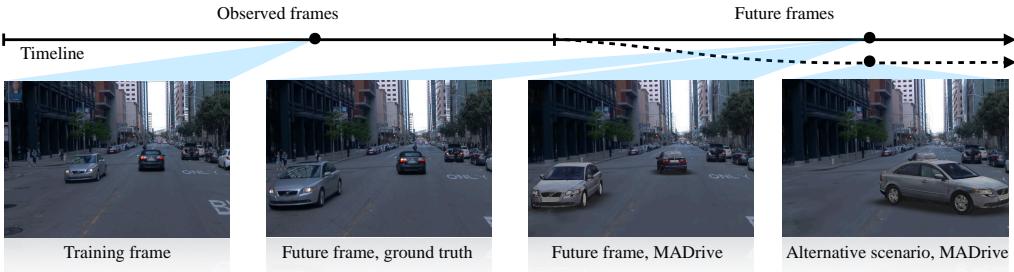
000 MADRIVE: MEMORY-AUGMENTED DRIVING SCENE 001 MODELING 002

003 **Anonymous authors**

004 Paper under double-blind review

005 ABSTRACT

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



024
025
026
027
028
029
030
031
032
033 Figure 1: **MADRIVE** reconstructs a 3D driving scene from training frames ([Left](#)) and replaces
034 partially observed vehicles in the scene with realistically reconstructed counterparts retrieved from
035 **MAD-CARS**, our novel multi-view auto dataset. **MADRIVE** enables high-fidelity modeling of
036 future scene views ([Middle-left](#) vs. [Middle-right](#)) and supports simulation of alternative scenarios,
037 advancing novel-view synthesis in dynamic environments ([Right](#)).

038 1 INTRODUCTION

039 Autonomous driving (AD) is one of the key areas in computer vision, requiring extensive and costly
040 data collection (Xiao et al., 2021; Sun et al., 2020; Geiger et al., 2012; Cabon et al., 2020; Caesar
041 et al., 2020) to train accurate and robust perception and planning models (Bansal et al., 2018; Huang
042 et al., 2024b; Gao et al., 2020). Driving simulators (Wang et al., 2023b; Zhou et al., 2024a; Yang
043 et al., 2023a) aim at offering a powerful alternative by enabling the generation of highly realistic
044 novel views and rare scenarios, especially safety-critical ones that are too dangerous or impractical
045 to capture in the wild. Accurate modeling of such scenarios is essential to mitigate domain shift
046 during training, which could otherwise lead to failures in deployments. When failures occur on real
047 roads, it is important to “debug” the driving policy and “replay” the scenario to address the root
048 cause.

049 Recent advances in multi-view reconstruction and novel view synthesis (Kerbl et al., 2023a; Yu
050 et al., 2024; Kheradmand et al., 2024) provide a foundation for developing highly realistic driving
051 simulators (Zhou et al., 2024a), designed to faithfully replicate real-world scenes. Such solutions
052 can be used for real-time, controllable simulations that preserve the visual domain of real-world
053 data, unlike game engine-based simulations, which often introduce significant domain shifts.

054 Modern driving scene reconstruction methods (Zhou et al., 2024b; Yan et al., 2024; Khan et al.,
 055 2024) have achieved impressive photo-realism in rendering the observed views, while also enabling
 056 the simulation of slight vehicle trajectory deviations, e.g., lane changes (Khan et al., 2024; Zhou
 057 et al., 2024b). However, since novel view synthesis is limited to the geometry observed in the data,
 058 existing reconstruction methods cannot reliably model vehicles beyond the observed views, which
 059 limits their usefulness for simulating alternative outcomes or reenacting failures in driving scenarios.
 060

061 **Motivation.** In real-world driving, AD system failures often require human intervention. To diag-
 062 nose such such failures, it is essential to reproduce the situation - modeling how events would have
 063 unfolded without intervention. Controllable reenactments not only help identify root causes, but also
 064 generate diverse training examples for robust AD. Therefore, we focus on controllable simulation of
 065 raw camera inputs to replicate frame sequences with similar surroundings and vehicles.
 066

067 **Contributions.** To overcome the limitations of existing methods, we introduce MADRIVE, a
 068 memory-augmented reconstruction framework that integrates external 3D car models into captured
 069 driving scenes. To faithfully adapt these models to the surrounding scene, we further propose
 070 physically-based relighting and insertion techniques, resulting in visually consistent novel driving
 071 scene views.
 072

073 Our method is motivated by the assumption that the variety of car models, types, and colors is
 074 relatively limited (Wikipedia contributors, 2024), making it feasible to build a dataset that covers
 075 the majority of cars typically seen on the road. To this end, we present a dataset containing 360°
 076 view sequences of ~70,000 cars, curated from online sale advertisements.
 077

078 Furthermore, leveraging integrated high-fidelity car models enables a more challenging evaluation
 079 setting. Previous solutions for driving scene reconstruction mainly focus on replicating or editing
 080 unseen *intermediate* frames (Yan et al., 2024; Khan et al., 2024; Zhou et al., 2024b;a). Our work
 081 instead considers *driving scene extrapolation* — predicting the future appearance of vehicles based
 082 on a sequence of past views. We show that MADRIVE can faithfully render a diverse range of
 083 plausible vehicle trajectories, offering a foundation for simulation applications that model alternative
 084 outcomes of real driving scenarios. In summary, the contributions are as follows.
 085

- 086 • We present MAD-CARS, **M**ulti-**V**iew **A**uto **D**ataset — a curated, large-scale collection of 360°
 087 car videos. It comprises ~70,000 car instances with diverse brands, models, colors, and lighting
 088 conditions, significantly expanding the scope of existing public multi-view car datasets.
 089
- 090 • We propose MADRIVE— a **M**emory-**A**ugmented **D**riving scene reconstruction framework aimed
 091 at realistic synthesis of diverse and complex driving scenarios. Given sparse car views in the
 092 scene, MADRIVE retrieves similar vehicles from a car video database, reconstructs them into
 093 high-quality 3D assets, and naturally integrates them into the scene, replacing the original cars.
 094
- 095 • We describe a novel evaluation setting for assessing driving scene reconstruction methods in
 096 significantly altered views, and show that MADRIVE produces more realistic renderings, as evi-
 097 denced by reduced performance degradation in downstream perception tasks.
 098

099 2 RELATED WORK

100 **Dynamic Urban Scene Reconstruction.** NeRFs (Mildenhall et al., 2020) can be used to model
 101 dynamic urban scenes. SUDS (Turki et al., 2023) uses a single network for dynamic actors, which
 102 limits the possibility of altering the behavior of the actors. EmerNeRF (Yang et al., 2024b) fol-
 103 lows a similar idea to SUDS by decomposing the scene purely into static and dynamic components.
 104 NeuRAD (Tonderski et al., 2024) takes advantage of monocular or LiDAR-based 3D bounding box
 105 predictions and proposes a joint optimization of object poses during the reconstruction process. Al-
 106 though these methods produce reasonable results, they are still 1) limited to the high training cost
 107 and low rendering speed; or 2) do not address extrapolation of future vehicle appearance far be-
 108 yond the original camera views. Recent dynamic 3D scene reconstruction methods increasingly
 109 adopt 3D Gaussian Splatting (Kerbl et al., 2023b) as an efficient and expressive scene representa-
 110 tion (Yang et al., 2023b; Wu et al., 2024; Yang et al., 2024c; Chen et al., 2023). Several approaches
 111 [StreetGS (Yan et al., 2024), AutoSplat (Khan et al., 2024), HUGS (Zhou et al., 2024b)] adopt these
 112 methods to driving scene modeling and decompose the scenes into a static background and fore-
 113 ground vehicles, placed in the scene using 3D bounding boxes derived from tracking data. These
 114 methods also propose various modifications to improve driving scene reconstruction and novel view
 115

synthesis. Both HUGS and AutoSplat represent the ground as a plane of 2D splats. HUGS further leverages additional information (optical flow and semantic segmentation) to guide splat optimization and introduces a method for realistic shadow placement. AutoSplat (Khan et al., 2024) improves car reconstruction from limited viewpoints by exploiting the bilateral symmetry of vehicles to augment side views and by employing more accurate splat initialization via an image-to-3D model (Pavllo et al., 2023). DrivingGaussian (Zhou et al., 2024c) uses composite dynamic Gaussian graph to handle multiple moving objects, individually reconstructing each object and restoring their accurate positions and occlusion relationships within the scene. OmniRe (Chen et al., 2025) leverages dynamic neural scene graphs based on Gaussian representations to unify the reconstruction of static backgrounds, driving vehicles, and non-rigidly moving dynamic actors, which enables human-centered simulations. Despite these advances, accurately reconstructing the full appearance of a vehicle in the scene, particularly from sparse or occluded views, remains a substantial challenge.

3D Car Datasets. Several public datasets provide 3D car assets. Early collections such as SRN-Car (Chang et al., 2015) and Objaverse-Car (Deitke et al., 2023) contain CAD models that deviate significantly from real-world cars in terms of texture realism and geometric details.

More recent efforts (Zhang et al., 2021; Du et al., 2024) have focused on real captured 3D car datasets. MVMC (Zhang et al., 2021) includes 576 cars, each with an average of 10 views. 3DReal-Car (Du et al., 2024) 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, thereby offering substantially greater generalization and diversity.

Novel View Synthesis with External 3D Car Assets. HUGSim (Zhou et al., 2024a) builds a closed-loop AD simulator by inserting 3D car models from 3DRealCars (Du et al., 2024). In contrast, we replace observed vehicles in real scenarios with retrieved counterparts, enabling extrapolation and rollout of actual driving situations.

Several approaches leverage CAD models for scene representation (Engelmann et al., 2017; Wang et al., 2023b; Uy et al., 2020; Avetisyan et al., 2019; Gümeli et al., 2022), but these assets often differ notably from real vehicles. To improve realism, some methods apply geometry tuning (Uy et al., 2020; Wang et al., 2023b; Engelmann et al., 2017), whereas UrbanCAD (Lu et al., 2024) retrieves similar CAD models and refines their textures and lighting to better match the scene while preserving CAD-level controllability. However, the obtained models still have a noticeable gap in realism and correspondence to actual cars.

Meanwhile, MADRIVE retrieves real cars instances from a large-scale database spanning diverse brands, models, materials, colors, and lighting conditions — aiming at closing the realism gap while preserving accurate scene alignment.

Relighting. Given a set of input views, scene reconstruction approaches based on radiance fields recover the outgoing radiance along with scene geometry. The radiance field depends on the scene’s lighting and varies when an object is placed in a different context. In general, the outgoing radiance is governed by the rendering equation (Kajiya, 1986). An exact solution to scene relighting would involve modeling light propagation via ray tracing. Although some recent works introduce solutions for efficient ray tracing (Xie et al., 2024; Govindarajan et al., 2025; Moenne-Loccoz et al., 2024; Byrski et al., 2025), relighting remains beyond the scope of their work.

As an alternative, several recent works model light propagation using approximations to the rendering equation from real-time graphics. LumiGauss (Kaleta et al., 2024) introduces a splat-based relighting method using spherical harmonics (Ramamoorthi & Hanrahan, 2001), but it requires multi-illumination data and is restricted to diffuse surfaces. GaussianShader (Jiang et al., 2024) employs the split-sum approximation (Karis & Games, 2013) to enhance specular reflections during reconstruction. In contrast, our approach leverages a similar PBR-based shading model while enabling relighting for scenes captured under fixed illumination.

Our relighting procedure requires an environmental map. In (Liang et al., 2024), DilPIR employs a generative model to infer it using a gradient-based procedure. In turn, we estimate the environmental map using training frames with (Phongthawee et al., 2024) without additional costly optimization procedure.

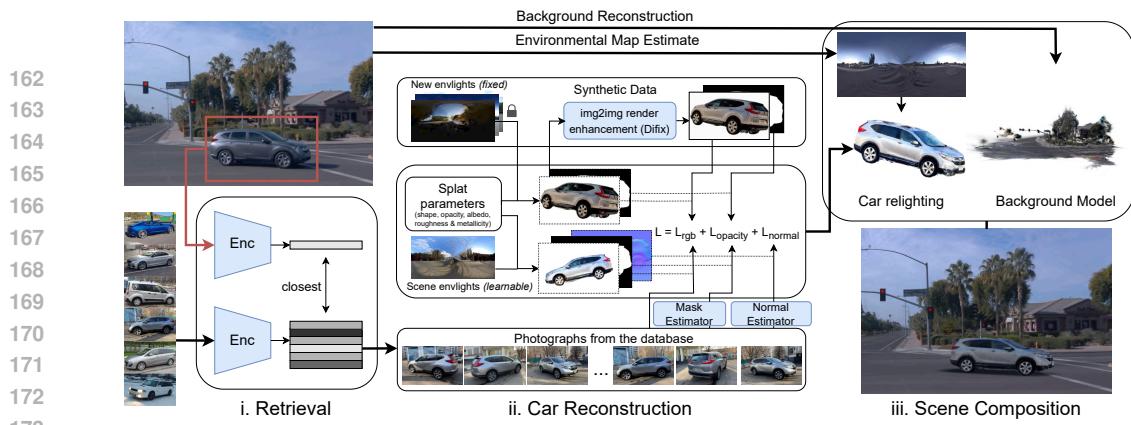


Figure 2: **MADRIVE Overview**. Given an input frame sequence, our retrieval scheme finds similar vehicles in an external database (Left). The 3D reconstruction pipeline then produces detailed vehicle models from the retrieved videos. The vehicles are represented with relightable 2D Gaussian splats. Opacity masks are used to remove background splats. The model geometry is regularized with external normals maps. (Middle). The reconstructed vehicles are adapted to the scene’s lighting conditions and composed with the background to produce the overall scene representation (Right).

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. The overview of our method is presented in Figure 2. In the following, we describe the proposed method in detail.

3.1 DRIVING SCENE RECONSTRUCTION

Following (Yan et al., 2024), we decompose the scene into static and dynamic components. The static component can be reconstructed based on the video from the moving vehicle. The movement parallax and the availability of depth sensor data allow to recover the scene structure.

We adapt the approach from Street Gaussians (Khan et al., 2024) to represent the static component of the scene consisting of three parts: ground, surroundings, and sky. We parameterize the surroundings with 3D Gaussian Splats (Kerbl et al., 2023a). We represent the ground part of the scene with horizontal 2D Gaussian splats. We avoid distance estimation ambiguities by putting the sky at an infinite distance and blending it into the scene at the last step.

The dynamic component includes all moving vehicles in the scene. For simplicity, we treat cars labeled as stationary in the dataset metadata as part of the static component. In general, there are two challenges in estimating and modeling the dynamic part of the scene. First, it requires accounting for compound motion. Second, observations often capture only a limited portion of a dynamic object. For example, predicting a vehicle’s side turn is difficult if its appearance from certain angles was never observed.

In line with Gaussian splatting-based urban driving scene modeling works (Yan et al., 2024; Khan et al., 2024), we represent observed vehicles as static Gaussian splats within the corresponding moving bounding box to model the compound motion in the scene. To obtain the static part of the scene, we initialize both static and dynamic parts with LIDAR data and train the splats with 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. This substitution allows obtaining high-quality renders for configurations significantly diverging from the ones observed during training.

3.2 RETRIEVAL AND CAR RECONSTRUCTION

We propose reconstructing the dynamic part of the scene with a retrieval-based approach. Specifically, we first extract crops of the moving cars observed in the scene and find the similar car instances in the database of multi-view car captures. Then, given the retrieved images, we construct photorealistic 3D car models and replace the original cars in the scene with the obtained 3D assets. Despite

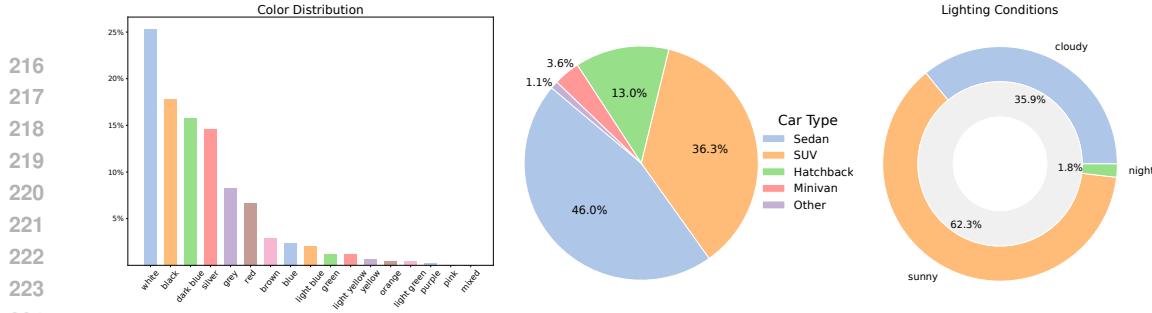


Figure 3: **MAD-CARS Analysis.** Memory-bank statistics on colors (Left), car types (Middle) and lighting conditions (Right).

the limited car visibility in the scene, retrieval-augmented reconstruction enables faithful 3D car reconstruction even from a single frame.

Retrieval Details. To produce a retrieval query, we compute a mask by projecting the 3D bounding box of a car onto an image plane. After filtering out small masks and overlapping masks, we use the remaining ones to extract image crops containing individual cars. For each crop, we compute an image embedding using SigLIP2 (Tschannen et al., 2025) and extract the car color using Qwen2.5-VL (Yang et al., 2024a). This color cue complements the image features, which tend to focus more on brand and car type, as observed in our experiments. To retrieve the car instance in the database, we first collect database entries with similar color and then select the one with the closest image embedding. We used YOLOv11 (Jocher & Qiu, 2024) to obtain instance segmentation masks for filtering out nearby cars in the scene. Once matches are found, we reconstruct the corresponding 3D car models using the associated multi-view image sets. The following section details our reconstruction procedure.

Relightable Car Models. We begin by specifying the representation used to model vehicles. By default, Gaussian splatting approximates the radiance field observed in the training frames as a whole. 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 (Burley & Studios, 2012).

We use a two-dimensional modification of Gaussian splats (Huang et al., 2024a), 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 effects.

To disentangle scene lighting from surface materials, we adopt the lighting model from (Munkberg et al., 2022) for each splat. The model assumes distant illumination with incident radiance $L_i(\omega_i)$ and defines the outgoing radiance in direction ω_o according to the rendering equation (Kajiya, 1986):

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

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 (Munkberg et al., 2022), we parameterize each splat's BSDF using the Cook–Torrance shading model (Cook & Torrance, 1982), 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 (Munkberg et al., 2022), which allows us to jointly infer incident radiance and splat BSDF parameters during optimization.

Car Reconstruction Details. 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 $\mathcal{L}_{\text{rgb}} = \mathcal{L}_1(I_i, \hat{I}_i) + \mathcal{L}_{\text{SSIM}}(I_i, \hat{I}_i)$ along with several regularizers. To exclude unnecessary background objects from the model, we generate masks $\hat{M}_i(x, y) = [\hat{I}_i(x, y) \text{ is part of a car}]$ with



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**).

Mask2Former (Cheng et al., 2022) to indicate pixels that belong to the model. Our opacity loss promotes high transparency outside of car pixels $\mathcal{L}_{\text{opacity}} = \sum_{x,y} (1 - \hat{M}_i) \cdot T_i$, 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 (Bin et al., 2025) and use the estimates to regularize Gaussian orientations. For the rendered normal maps N_i , the regularizer is $\mathcal{L}_{\text{normal}} = \sum_{x,y} \hat{M}_i \cdot (1 - N_i^T \hat{N}_i)$. The resulting objective is

$$\mathcal{L}_{\text{gt}}(I_i, \hat{I}_i) = \mathcal{L}_{\text{rgb}}(I_i, \hat{I}_i) + \lambda_{\text{opacity}} \mathcal{L}_{\text{opacity}}(I_i, \hat{I}_i) + \lambda_{\text{normal}} \mathcal{L}_{\text{normal}}(I_i, \hat{I}_i). \quad (2)$$

Furthermore, to promote realistic appearance under novel lighting conditions, we incorporate additional synthetic data. Disentangling illumination and scene materials is particularly challenging for in-the-wild captures, where lighting remains fixed throughout training. To address this limitation, we approximate a multi-illumination setup using synthetic data. Specifically, we render random model views under varying pre-defined environmental lighting and enhance these renders using the Difix image-to-image model (Wu et al., 2025a). The enhanced images \tilde{I}_i are then used as training samples. To compute the objective in Eq. 2, we render frames I_i with the same environmental lighting as in \tilde{I}_i , use the α -channel of \tilde{I}_i as the mask, and omit the normal regularization. We add the synthetic frames in the same proportion as the real frames. Synthetic frames are introduced after the initial 10k gradient steps and regenerated every 2.5k steps throughout the following 20k steps.

3.3 CAR INSERTION AND RELIGHTING

The final stage of our pipeline integrates reconstructed cars into the learned scene. First, we prepare each car for the insertion. We remove occasional splats that are either positioned behind the training cameras or project onto pixels outside the car mask in the training images. The car is then oriented based on the principal components of its point cloud, further refined using an orientation model (Scarvelis et al., 2024) to ensure proper alignment within the driving scene.

Next, the aligned point cloud is placed inside the bounding box of the original car to be replaced. To achieve precise alignment in both scale and position, we use the Iterative Closest Point (ICP) algorithm (Besl & McKay, 1992) and apply the resulting transformation to the inserted car. To enhance visual realism, we add a shadow beneath the car, modeled as a black plane composed of 2D splats placed under the wheels. While more sophisticated shadow placement based on sun position, as explored in (Zhou et al., 2024a), could be considered, we find it non-essential for our method.

Since the retrieved car asset is captured under different lighting conditions, we estimate the target scene’s environment map and adjust the car’s appearance via Eq. 1 to ensure visual consistency. As the Waymo dataset (Sun et al., 2020) lacks full 360° camera coverage, we approximate lighting conditions using DiffusionLight (Phongthawee et al., 2024), which reconstructs missing environment map regions via diffusion-based inpainting. Given a training frame, we estimate the environment map and align it with the corresponding camera orientation. In addition, we adjust the scale of the environmental map to minimize the tone discrepancy between the last training frame and our render.

3.4 DATABASE COLLECTION AND STATISTICS

This work introduces MAD-CARS, a large-scale database of multi-view car videos in the wild, sourced from online car sale advertisements. The database contains $\sim 70,000$ diverse video instances, each averaging ~ 85 frames, with most car instances available at a resolution of 1920×1080 .

324 It includes cars from ~ 150 brands, covering a broad range of colors, car types, and three lighting
 325 conditions. Distributions of color, car type, and lighting are illustrated in Figure 3. The metadata for
 326 each car instance is presented in the dataset.

327 The data is carefully curated by filtering out frames and entire car instances that could negatively
 328 impact 3D reconstruction. In more detail, we remove low quality and overly dark frames with CLIP-
 329 IQA (Wang et al., 2023a) and use Qwen2.5-VL (Yang et al., 2024a) to detect finger blocked shots,
 330 car interior views, the frames where the car view is occluded, e.g., by fences, trees, other vehicles,
 331 etc. More data collection details are provided in Appendix A.

333 4 EXPERIMENTS

334 In the following, we report the MADRIVE performance. Section 4.1 describes our evaluation
 335 setup. Then, we proceed to the main results in Section 4.2. Finally, we explore the retrieval, car
 336 reconstruction and relighting procedures in Section Section 4.3.

337 4.1 EVALUATION SETUP

339 **Scene Reconstruction Dataset.** We reconstruct the driving scenes from the Waymo Open Motion
 340 dataset (Ettinger et al., 2021). We picked 12 particularly challenging scenes containing multiple
 341 cars, driving maneuvers and diverse lighting conditions. Then, we manually select scene sequences
 342 and divide them into training and evaluation clips. In our experiments, we simultaneously use videos
 343 from frontal and two side cameras to capture a wide field of view and track cars moving across the
 344 scene. More evaluation setup details are provided in Appendix B.

345 **Scene Extrapolation with Novel View Synthesis.** For our evaluation, we selected driving scenes
 346 involving U-turns, intersection crossings, and parking departures — common accident scenarios
 347 that also reveal vehicles from diverse viewpoints, posing challenges for reconstruction. Each frame
 348 sequence was manually split into training and testing subsets at the midpoint of the maneuver. We
 349 use the whole sequence to reconstruct the background and then remove the cars using the annotated
 350 bounding boxes in the Waymo dataset. Car reconstruction is performed using only the first part of
 351 the sequence, while the second part is reserved for evaluating scene reconstruction quality. We aim
 352 to generate realistic views of the scene by extrapolating the observed data. In particular, we insert the
 353 reconstructed car models into the background according to location and orientation specified by the
 354 bounding boxes on the holdout sequence. By design, our setup evaluates scenes under configurations
 355 that differ significantly from the frames seen during training. At the same time, the data split ensures
 356 that test frames do not leak into the car reconstruction process.

356 **Baselines.** We compare MADRIVE with the scene reconstruction Gaussian splatting-based meth-
 357 ods that were previously considered for novel view synthesis: Street-Gaussians (SG) (Yan et al.,
 358 2024), AutoSplat (Khan et al., 2024) (our implementation), and HUGS (Zhou et al., 2024b). Details
 359 on training and evaluation of baselines are given in Appendix D.

360 4.2 MAIN EXPERIMENTS

361 **Qualitative Evaluation.** First, we provide visual scene reconstruction results for qualitative anal-
 362 ysis. In Figure 4, we compare rendering results on the training and hold-out frames. Although SG,
 363 AutoSplat, and HUGS produce accurate approximations of training frames, on the test frames cars
 364 tend to fell apart for novel view angles. Compared to baselines, our method cannot reproduce the
 365 training frames with the same precision, but is significantly more robust to deviations from training
 366 configurations. More visual examples are presented in Figures 9, 10. We also provide the visualiza-
 367 tions with modified trajectories in Figure 11.

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

| 369 Model | 370 MOTA \uparrow | 371 MOTP \downarrow | 372 IDF1 \uparrow | 373 Segmentation IoU \uparrow |
|------------------------------------|---------------------|-----------------------|---------------------|---------------------------------|
| 374 Street-GS (Yan et al., 2024) | 0.654 | 0.105 | 0.776 | 0.556 |
| 375 HUGS (Zhou et al., 2024b) | 0.556 | 0.221 | 0.699 | 0.333 |
| 376 AutoSplat* (Khan et al., 2024) | 0.589 | 0.154 | 0.716 | 0.489 |
| 377 MADRIVE (Ours) | 0.841 | 0.138 | 0.913 | 0.818 |

378 *Denotes our reimplementations.

379 **Quantitative Evaluation.** In our main experiments, we assess tracking and segmentation perfor-
 380 mance on synthesized **test** frames. Specifically, we apply state-of-the-art tracking and segmentation



Figure 5: **Qualitative comparison of reconstructed vehicles** on KITTI-360 from the reference and rotated viewpoints. MADRIVE on top of the MAD-CARS dataset produces more similar and realistic 3D assets.

models to both synthesized and ground truth frames and compare their outputs using established metrics for each task. For tracking, we use BotSort (Aharon et al., 2022) with a YOLOv8n backbone, reporting multiple object tracking accuracy (MOTA↑), precision (MOTP↓), and identity F1 score (IDF1) (Milan et al., 2016). For segmentation, we compute the average intersection-over-union (IoU) using instance segmentation masks obtained with Mask2Former (Cheng et al., 2022). Table 1 presents the comparison of MADRIVE against the baselines. MADRIVE shows substantially superior performance compared to the baselines in 2 tracking metrics (MOTA and IDF) and segmentation metric IoU. This observation is also supported by the visual examples provided in Figure 4. We explain the MOTP gap between our method and Street-GS by the better car alignment of Street-GS in the first test frames, while later frames, where the tracker fails to detect Street-GS cars, are not counted in the MOTP calculation. We provide per scene results for all 12 scenes in Appendix C. We also discuss the choice for the reference masks in the evaluation protocol in Appendix E.

4.3 FURTHER EVALUATION

Retrieval. Here, we evaluate the performance of the proposed retrieval module isolated from other components to address how accurately the retrieved cars correspond to the original cars in the scene.

We compare the retrieval performance on the proposed dataset against 3DRealCars (Du et al., 2024), highly accurate publicly available dataset of 2,500 car assets. To evaluate the retrieval quality, we first calculate the average L2 distance between the car images from the driving scene and the nearest cars from the memory bank. We use SigLIP2 So (Tschanen et al., 2025) as an image feature extractor. Then, we provide the accuracy obtained with the Qwen2.5-VL-32B-Instruct model, asked to compare the cars in terms of their brand, model, color and car type. For a fair comparison, we do not use the color filtering in this experiment.

Table 2 shows the retrieval accuracy across different attributes, and the average L2 distance to the closest instance. We observe that candidates retrieved using MAD-CARS more accurately match the cars in the driving scenes, which we attribute to the significantly larger scale of MAD-CARS.

Importantly, Table 2 highlights that retrieval based solely on feature extractors often disregards car color, despite its importance for realistic car replacement. The similar problem has been observed for non-visual-language encoders such as DINOv2 (Oquab et al., 2023). Our additional results

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 (Du et al., 2024) 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 |

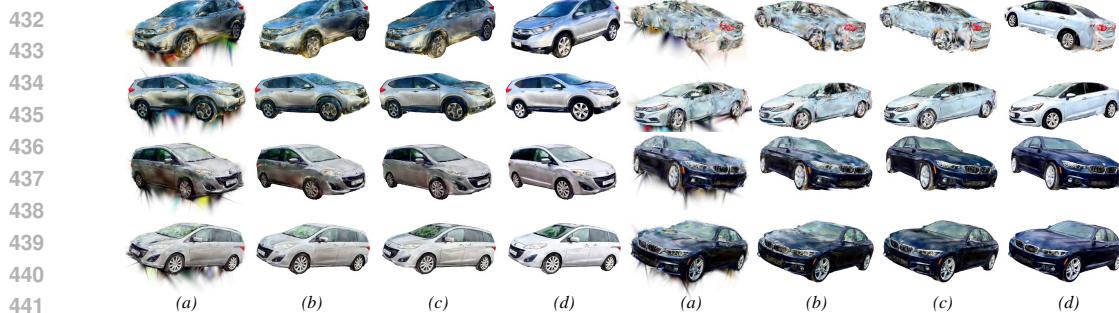


Figure 6: Ablation of reconstruction regularizers, each setting adds one component to the previous and shows albedo. (a) Without regularization, shape and texture artifacts cause uneven edges. (b) Adding opacity regularization improves edge quality. (c) Adding normal regularization enhances surface smoothness. (d) Training with synthetic frames under varying lighting disentangles illumination from object color, yielding cleaner albedo and reconstructions.

in Appendix F show that applying a color-based pre-filtering improves color consistency between the retrieved and target vehicles.

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: car reconstruction on a different car dataset (Du et al., 2024), matching with a car model from a CAD dataset (Lu et al., 2024), and running a cutting edge image-to-3D models (Lin et al., 2025; Wu et al., 2025b). 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 further ablate the components of the proposed reconstruction algorithm. Figure 6 shows the recovered geometry and albedo for several cars. Reconstruction without regularization (a) produces noticeable artifacts in both shape and texture, leading to uneven edges after background removal. Introducing opacity regularization to suppress the background during reconstruction (b) improves edge quality. Adding normal regularization (c) further enhances surface smoothness and consistency. Finally, incorporating synthetic frames under varying lighting conditions (d) helps disentangle scene illumination from object color, yielding more accurate albedo and overall cleaner reconstructions.

Relighting. We conclude with a qualitative comparison of the proposed relighting scheme. For a number of scenes, we reconstructed scene frames with and without the relighting module. Figure 7 shows that the relighting module adapts the model colors to the environment, helping to reduce the inconsistencies that break immersion and make the inserted models appear naturally lit within the scene.

5 CONCLUSION

This work presents MADRIVE, a novel driving scene reconstruction approach specifically designed to model significantly altered vehicle positions. Powered by MADCARS, our large-scale multi-view car dataset, MADRIVE replaces dynamic vehicles in a scene with similar car instances from the database. We believe that MADRIVE could make a step towards modeling multiple potential outcomes for analyzing an autonomous driving system’s behavior in safety-critical situations.

However, despite the promising visual fidelity of future scenario frames, they still differ from the ground truth, as we discuss in Appendix I. Future work could focus on expanding the database with a wider range of car brands, models, and types, as well as enhancing corrupted car videos using recent multi-view diffusion (Zhou et al., 2025).

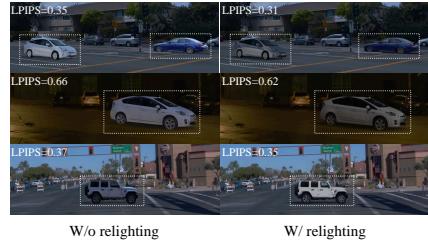


Figure 7: **Relighting ablation.** Rendered hold-out frames without (Left) and with (Right) relighting.

486 REFERENCES
487

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

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

493 Mayank Bansal, Alex Krizhevsky, and Abhijit Ogale. Chauffeurnet: Learning to drive by imitating
494 the best and synthesizing the worst. *arXiv preprint arXiv:1812.03079*, 2018.

495 Paul Besl and H.D. McKay. A method for registration of 3-d shapes. *ieee trans pattern anal mach*
496 *intell. Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 14:239–256, 03 1992.
497 doi: 10.1109/34.121791.

498 Yanrui Bin, Wenbo Hu, Haoyuan Wang, Xinya Chen, and Bing Wang. Normalcrafter: Learning
499 temporally consistent normals from video diffusion priors. *arXiv preprint arXiv:2504.11427*,
500 2025.

501 Brent Burley and Walt Disney Animation Studios. Physically-based shading at disney. In *Acm*
502 *siggraph*, volume 2012, pp. 1–7. vol. 2012, 2012.

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

506 Yohann Cabon, Naila Murray, and Martin Humenberger. Virtual kitti 2. *arXiv preprint*
507 *arXiv:2001.10773*, 2020.

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

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

515 Yurui Chen, Chun Gu, Junzhe Jiang, Xiatian Zhu, and Li Zhang. Periodic vibration gaussian:
516 Dynamic urban scene reconstruction and real-time rendering. *arXiv preprint arXiv:2311.18561*,
517 2023.

518 Ziyu Chen, Jiawei Yang, Jiahui Huang, Riccardo de Lutio, Janick Martinez Esturo, Boris Ivanovic,
519 Or Litany, Zan Gojcic, Sanja Fidler, Marco Pavone, Li Song, and Yue Wang. Omnidre: Omni urban
520 scene reconstruction. In *The Thirteenth International Conference on Learning Representations*,
521 2025. URL <https://openreview.net/forum?id=11x9giME15o>.

522 Bowen Cheng, Ishan Misra, Alexander G. Schwing, Alexander Kirillov, and Rohit Girdhar.
523 Masked-attention mask transformer for universal image segmentation. In *2022 IEEE/CVF Con-*
524 *ference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1280–1289, 2022. doi:
525 10.1109/CVPR52688.2022.00135.

526 Robert L Cook and Kenneth E. Torrance. A reflectance model for computer graphics. *ACM Trans-*
527 *actions on Graphics (ToG)*, 1(1):7–24, 1982.

528 Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig
529 Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of anno-
530 tated 3d objects. In *Proceedings of the IEEE/CVF conference on computer vision and pattern*
531 *recognition*, pp. 13142–13153, 2023.

532 Xiaobiao Du, Haiyang Sun, Shuyun Wang, Zhuojie Wu, Hongwei Sheng, Jiaying Ying, Ming Lu,
533 Tianqing Zhu, Kun Zhan, and Xin Yu. 3drealcar: An in-the-wild rgb-d car dataset with 360-degree
534 views. *arXiv preprint arXiv:2406.04875*, 2024.

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

543

544 Scott Ettinger, Shuyang Cheng, Benjamin Caine, Chenxi Liu, Hang Zhao, Sabeek Pradhan, Yuning
 545 Chai, Ben Sapp, Charles R Qi, Yin Zhou, et al. Large scale interactive motion forecasting for
 546 autonomous driving: The waymo open motion dataset. In *Proceedings of the IEEE/CVF Interna-*
 547 *tional Conference on Computer Vision*, pp. 9710–9719, 2021.

548 Jiyang Gao, Chen Sun, Hang Zhao, Yi Shen, Dragomir Anguelov, Congcong Li, and Cordelia
 549 Schmid. Vectornet: Encoding hd maps and agent dynamics from vectorized representation. In
 550 *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11525–
 551 11533, 2020.

552 Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti
 553 vision benchmark suite. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*,
 554 pp. 3354–3361, 2012. doi: 10.1109/CVPR.2012.6248074.

555

556 Shrisudhan Govindarajan, Daniel Rebain, Kwang Moo Yi, and Andrea Tagliasacchi. Radiant foam:
 557 Real-time differentiable ray tracing. *arXiv preprint arXiv:2502.01157*, 2025.

558 Can Gümeli, Angela Dai, and Matthias Nießner. Roca: Robust cad model retrieval and alignment
 559 from a single image. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition*
 560 (*CVPR*), pp. 4012–4021, 2022. doi: 10.1109/CVPR52688.2022.00399.

561

562 Binbin Huang, Zehao Yu, Anpei Chen, Andreas Geiger, and Shenghua Gao. 2d gaussian splatting
 563 for geometrically accurate radiance fields. In *ACM SIGGRAPH 2024 conference papers*, pp.
 564 1–11, 2024a.

565 Xin Huang, Eric M Wolff, Paul Vernaza, Tung Phan-Minh, Hongge Chen, David S Hayden, Mark
 566 Edmonds, Brian Pierce, Xinxin Chen, Pratik Elias Jacob, et al. Drivegpt: Scaling autoregressive
 567 behavior models for driving. *arXiv preprint arXiv:2412.14415*, 2024b.

568

569 Yingwenqi Jiang, Jiadong Tu, Yuan Liu, Xifeng Gao, Xiaoxiao Long, Wenping Wang, and Yuexin
 570 Ma. Gaussianshader: 3d gaussian splatting with shading functions for reflective surfaces. In *Pro-*
 571 *ceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5322–
 572 5332, 2024.

573 Glenn Jocher and Jing Qiu. Ultralytics yolo11, 2024. URL <https://github.com/ultralytics/ultralytics>.

574

575 James T Kajiya. The rendering equation. In *Proceedings of the 13th annual conference on Computer*
 576 *graphics and interactive techniques*, pp. 143–150, 1986.

577

578 Joanna Kaleta, Kacper Kania, Tomasz Trzcinski, and Marek Kowalski. Lumigauss: High-fidelity
 579 outdoor relighting with 2d gaussian splatting. *arXiv preprint arXiv:2408.04474*, 2024.

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

582

583 Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splat-
 584 tting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139–1, 2023a.

585

586 Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splat-
 587 tting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4), July 2023b.
 588 URL <https://repo-sam.inria.fr/fungraph/3d-gaussian-splatting/>.

589

590 Mustafa Khan, Hamidreza Fazlali, Dhruv Sharma, Tongtong Cao, Dongfeng Bai, Yuan Ren, and
 591 Bingbing Liu. Autosplat: Constrained gaussian splatting for autonomous driving scene recon-
 592 struction. *arXiv preprint arXiv:2407.02598*, 2024.

593

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

594 Ruofan Liang, Zan Gojcic, Merlin Nimier-David, David Acuna, Nandita Vijaykumar, Sanja Fidler,
 595 and Zian Wang. Photorealistic object insertion with diffusion-guided inverse rendering. In *Euro-
 596 pean Conference on Computer Vision*, pp. 446–465. Springer, 2024.

597

598 Chenguo Lin, Panwang Pan, Bangbang Yang, Zeming Li, and Yadong Mu. Diffsplat: Repurposing
 599 image diffusion models for scalable gaussian splat generation. *arXiv preprint arXiv:2501.16764*,
 600 2025.

601 Yichong Lu, Yichi Cai, Shangzhan Zhang, Hongyu Zhou, Haoji Hu, Huimin Yu, Andreas Geiger,
 602 and Yiyi Liao. Urbancad: Towards highly controllable and photorealistic 3d vehicles for urban
 603 scene simulation. *arXiv preprint arXiv:2411.19292*, 2024.

604

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

607

608 Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and
 609 Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In Andrea
 610 Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm (eds.), *Computer Vision – ECCV
 2020*, pp. 405–421, Cham, 2020. Springer International Publishing. ISBN 978-3-030-58452-8.

611

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

615

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

619

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

625

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

629

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

633

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

636

637 Christopher Scarvelis, David Benhaim, and Paul Zhang. Orient anything. *arXiv preprint
 arXiv:2410.02101*, 2024.

638

639 Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurélien Chouard, Vijaysai Patnaik, Paul Tsui,
 640 James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, Vijay Vasudevan, Wei Han, Jiquan
 641 Ngiam, Hang Zhao, Aleksei Timofeev, Scott Ettinger, Maxim Krivokon, Amy Gao, Aditya Joshi,
 642 Yu Zhang, Jonathon Shlens, Zhifeng Chen, and Dragomir Anguelov. Scalability in perception for
 643 autonomous driving: Waymo open dataset. In *2020 IEEE/CVF Conference on Computer Vision
 and Pattern Recognition (CVPR)*, pp. 2443–2451, 2020. doi: 10.1109/CVPR42600.2020.00252.

644

645 Adam Tonderski, Carl Lindström, Georg Hess, William Ljungbergh, Lennart Svensson, and
 646 Christoffer Petersson. Neurad: Neural rendering for autonomous driving. In *2024 IEEE/CVF
 647 Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 14895–14904, 2024. doi:
 10.1109/CVPR52733.2024.01411.

648 Michael Tschanne, Alexey Gritsenko, Xiao Wang, Muhammad Ferjad Naeem, Ibrahim Alabdul-
 649 mohsin, Nikhil Parthasarathy, Talfan Evans, Lucas Beyer, Ye Xia, Basil Mustafa, Olivier Hénaff,
 650 Jeremiah Harmsen, Andreas Steiner, and Xiaohua Zhai. Siglip 2: Multilingual vision-language
 651 encoders with improved semantic understanding, localization, and dense features. *arXiv preprint*
 652 *arXiv:2502.14786*, 2025.

653 Haithem Turki, Jason Y. Zhang, Francesco Ferroni, and Deva Ramanan. Suds: Scalable urban
 654 dynamic scenes. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition*
 655 (*CVPR*), pp. 12375–12385, 2023. doi: 10.1109/CVPR52729.2023.01191.

656

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

661 Jianyi Wang, Kelvin CK Chan, and Chen Change Loy. Exploring clip for assessing the look and feel
 662 of images. In *AAAI*, 2023a.

663

664 Jingkang Wang, Sivabalan Manivasagam, Yun Chen, Ze Yang, Ioan Andrei Bârsan, Anqi Joyce
 665 Yang, Wei-Chiu Ma, and Raquel Urtasun. Cadsim: Robust and scalable in-the-wild 3d recon-
 666 struction for controllable sensor simulation. *arXiv preprint arXiv:2311.01447*, 2023b.

667 Wikipedia contributors. Car colour popularity, 2024. URL [https://en.wikipedia.org/](https://en.wikipedia.org/wiki/Car_colour_popularity)
 668 *wiki/Car_colour_popularity*. [Online; accessed 2024-07-19].

669

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

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

677 Tianhao Wu, Chuanxia Zheng, Frank Guan, Andrea Vedaldi, and Tat-Jen Cham. Amodal3r: Amodal
 678 3d reconstruction from occluded 2d images. *arXiv preprint arXiv:2503.13439*, 2025b.

679

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

684 Tao Xie, Xi Chen, Zhen Xu, Yiman Xie, Yudong Jin, Yujun Shen, Sida Peng, Hujun Bao, and
 685 Xiaowei Zhou. Envgs: Modeling view-dependent appearance with environment gaussian. *arXiv*
 686 *preprint arXiv:2412.15215*, 2024.

687

688 Yunzhi Yan, Haotong Lin, Chenxu Zhou, Weijie Wang, Haiyang Sun, Kun Zhan, Xianpeng Lang,
 689 Xiaowei Zhou, and Sida Peng. Street gaussians: Modeling dynamic urban scenes with gaussian
 690 splatting. In *European Conference on Computer Vision*, pp. 156–173. Springer, 2024.

691 An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li,
 692 Dayiheng Liu, Fei Huang, Haoran Wei, et al. Qwen2. 5 technical report. *arXiv preprint*
 693 *arXiv:2412.15115*, 2024a.

694

695 Jiawei Yang, Boris Ivanovic, Or Litany, Xinshuo Weng, Seung Wook Kim, Boyi Li, Tong Che,
 696 Danfei Xu, Sanja Fidler, Marco Pavone, and Yue Wang. Emernerf: Emergent spatial-temporal
 697 scene decomposition via self-supervision. In *The Twelfth International Conference on Learning*
 698 *Representations*, 2024b. URL <https://openreview.net/forum?id=ycv2z8TYur>.

699

700 Ze Yang, Yun Chen, Jingkang Wang, Sivabalan Manivasagam, Wei-Chiu Ma, Anqi Joyce Yang, and
 701 Raquel Urtasun. Unisim: A neural closed-loop sensor simulator. In *2023 IEEE/CVF Conference*
 702 *on Computer Vision and Pattern Recognition (CVPR)*, pp. 1389–1399, 2023a. doi: 10.1109/
 703 CVPR52729.2023.00140.

702 Zeyu Yang, Hongye Yang, Zijie Pan, and Li Zhang. Real-time photorealistic dynamic scene repre-
 703 sentation and rendering with 4d gaussian splatting. *arXiv preprint arXiv:2310.10642*, 2023b.
 704

705 Ziyi Yang, Xinyu Gao, Wen Zhou, Shaohui Jiao, Yuqing Zhang, and Xiaogang Jin. Deformable
 706 3d gaussians for high-fidelity monocular dynamic scene reconstruction. In *Proceedings of the*
 707 *IEEE/CVF conference on computer vision and pattern recognition*, pp. 20331–20341, 2024c.

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

711 Jason Zhang, Gengshan Yang, Shubham Tulsiani, and Deva Ramanan. Ners: Neural reflectance sur-
 712 faces for sparse-view 3d reconstruction in the wild. *Advances in Neural Information Processing*
 713 *Systems*, 34:29835–29847, 2021.

714 Hongyu Zhou, Longzhong Lin, Jiabao Wang, Yichong Lu, Dongfeng Bai, Bingbing Liu, Yue Wang,
 715 Andreas Geiger, and Yiyi Liao. Hugsim: A real-time, photo-realistic and closed-loop simulator
 716 for autonomous driving. *arXiv preprint arXiv:2412.01718*, 2024a.

717 Hongyu Zhou, Jiahao Shao, Lu Xu, Dongfeng Bai, Weichao Qiu, Bingbing Liu, Yue Wang, Andreas
 718 Geiger, and Yiyi Liao. Hugs: Holistic urban 3d scene understanding via gaussian splatting. In
 719 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*,
 720 pp. 21336–21345, June 2024b.

721 Jensen Jinghao Zhou, Hang Gao, Vikram Voleti, Aaryaman Vasishta, Chun-Han Yao, Mark Boss,
 722 Philip Torr, Christian Rupprecht, and Varun Jampani. Stable virtual camera: Generative view
 723 synthesis with diffusion models. *arXiv preprint arXiv:2503.14489*, 2025.

724 Xiaoyu Zhou, Zhiwei Lin, Xiaojun Shan, Yongtao Wang, Deqing Sun, and Ming-Hsuan Yang. Driv-
 725 inggaussian: Composite gaussian splatting for surrounding dynamic autonomous driving scenes.
 726 In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21634–
 727 21643, 2024c. doi: 10.1109/CVPR52733.2024.02044.

728

729

730

731

732

733

734

735

736

737

738

739

740

741

742

743

744

745

746

747

748

749

750

751

752

753

754

755

756 **A DATA COLLECTION DETAILS**
757

758 The initial database contained $\sim 95,000$ car videos of ~ 100 views on average. The first filtering
759 stage includes the filtering of low quality and overly dark images with the CLIP-IQA model (Wang
760 et al., 2023a), discarding frames with a score < 0.2 . Then, we use Qwen-2.5-VL-Instruct (7B) (Yang
761 et al., 2024a) to respond several questions for each frame:

- 762 • “Does the image depict a car?”
- 763 • “Is the car directly occluded?”
- 764 • “Does the image depict the car interior?”
- 765 • “Does a hand or finger block the view?”
- 766 • “Is the car door open?”
- 767 • “Does the image mainly depict the car window?”
- 768
- 769

770 Based on the responses, we filter out the corresponding frames or, in some cases, entire car instances.
771 Also, if fewer than 45 valid frames remain for a given instance, the entire instance is discarded.
772

773 **B EVALUATION SETUP DETAILS**
774

775 For scene reconstruction evaluation, we selected 12 scenes from the Waymo Open Dataset (Sun
776 et al., 2020), with labels listed in Table 3. This table also provides the correspondence between
777 the original scene labels from the Waymo Cloud Storage and the short names used in our work.
778 We split each scene into training and testing subsets based on time (Table 4) and camera selection
779 (Table 5). Specifically, frames with indices i^{train} , where $i^{\text{train}} \in [i_{\text{start}}^{\text{train}}, i_{\text{end}}^{\text{train}}]$, were used for training.
780 For evaluation, we used frames $i^{\text{test}} \in [i_{\text{start}}^{\text{test}}, i_{\text{end}}^{\text{test}}]$, with all split indices provided in Table 4.
781

782 **Table 3: Waymo scenes used for evaluation of scene reconstruction.**
783

| 784 Label | 785 Scene name |
|--|----------------|
| 785 1231623110026745648_480_000_500_000 | 786 123 |
| 786 1432918953215186312_5101_320_5121_320 | 787 143 |
| 787 1906113358876584689_1359_560_1379_560 | 788 190 |
| 788 10500357041547037089_1474_800_1494_800 | 789 105 |
| 789 10940952441434390507_1888_710_1908_710 | 790 109 |
| 790 16504318334867223853_480_000_500_000 | 791 165 |
| 791 17407069523496279950_4354_900_4374_900 | 792 174 |
| 792 18025338595059503802_571_216_591_216 | 793 180 |
| 793 14183710428479823719_3140_000_3160_000 | 794 141 |
| 794 15834329472172048691_2956_760_2976_760 | 795 158 |
| 795 17647858901077503501_1500_000_1520_000 | 796 176 |
| 796 7799671367768576481_260_000_280_000 | 797 779 |

798 **C PER-SCENE QUANTITATIVE EVALUATION.**
799

800 In addition to the aggregated results in Table 1, we report per-scene metric values in Table 6, Table 7,
801 Table 8, and Table 9, corresponding to MOTA, MOTP, IDF1, and IoU, respectively. We observe that
802 MADRIVE consistently outperforms the baselines across most scenes.

803 **D BASELINE DETAILS**
804

805 **Baselines training and evaluation.** We trained all baselines (Street-Gaussians, HUGS, and Au-
806 toSplat) for $10K$ iterations using the training frames with indices $i \in [i_{\text{start}}^{\text{train}}, i_{\text{end}}^{\text{train}}]$ as specified in
807 Table 4. Additionally, we trained the background models for both the baselines and MADRIVE
808 for $30K$ iterations using all available frames. These pretrained background models were then used
809 during the rendering of the test frames ($i \in [i_{\text{start}}^{\text{test}}, i_{\text{end}}^{\text{test}}]$), on which we compute the metrics reported
in Table 6, Table 7, Table 8, and Table 9.

810
 811 Table 4: Train and test frame splits for Waymo scenes over time. All values, except those in the
 812 leftmost column, indicate frame indices starting from 0.

| Scene name | i_{start}^{train} | i_{end}^{train} | i_{start}^{test} | i_{end}^{test} |
|------------|---------------------|-------------------|--------------------|------------------|
| 123 | 106 | 116 | 117 | 175 |
| 143 | 43 | 53 | 54 | 62 |
| 190 | 115 | 125 | 126 | 137 |
| 105 | 164 | 174 | 175 | 196 |
| 109 | 1 | 16 | 17 | 55 |
| 165 | 7 | 40 | 41 | 111 |
| 174 | 34 | 51 | 52 | 72 |
| 180 | 49 | 55 | 56 | 68 |
| 141 | 60 | 80 | 81 | 117 |
| 158 | 44 | 62 | 63 | 100 |
| 176 | 31 | 42 | 43 | 67 |
| 779 | 50 | 65 | 66 | 84 |

825
 826 Table 5: Train and test frame splits for Waymo scenes based on camera selection.
 827

| Scene name | Train cameras | Test cameras |
|------------|--------------------------------------|------------------------|
| 123 | frontal, frontal left | frontal, frontal left |
| 143 | frontal, frontal left | frontal, frontal left |
| 190 | frontal, frontal left | frontal, frontal left |
| 105 | frontal, frontal left | frontal |
| 109 | frontal, frontal right | frontal right |
| 165 | frontal, frontal left | frontal, frontal left |
| 174 | frontal | frontal |
| 180 | frontal, frontal right | frontal, frontal right |
| 141 | frontal | frontal |
| 158 | frontal | frontal |
| 176 | frontal | frontal |
| 779 | frontal, frontal left, frontal right | frontal, frontal right |

839
 840
 841 **Street-Gaussians.** We used the official implementation available at https://github.com/zju3dv/street_gaussians.
 842
 843

844
 845 **HUGS.** We used the official implementation provided at <https://github.com/hyzhou404/HUGSIM>.
 846

847
 848 **AutoSplat.** As no official implementation is publicly available, we re-implemented the core contributions of AutoSplat on top of the Street-Gaussians codebase.
 849

850 E CHOICE OF REFERENCE MASKS IN THE EVALUATION

851 In our validation setup, we used predictions from tracking and segmentation models on ground-truth
 852 images as targets, since the Waymo dataset lacks segmentation masks and 2D bounding boxes.
 853

854 To evaluate whether cars in the synthesized frames are as identifiable as those in the original frames,
 855 we applied the same detection algorithm to both. Our method outperforms the baseline, primarily
 856 because our system inserts visually coherent cars on test frames by leveraging reconstructed models,
 857 whereas baseline approaches result in degraded or incomplete vehicle representations.
 858

859 However, the inserted cars might be easier to detect. To test this, we conducted an additional experiment.
 860 Specifically, we generated a new set of detector targets by projecting the 3D bounding boxes
 861 provided in the Waymo dataset onto the image plane. We then evaluated the performance of the
 862 detector on both the ground-truth and synthesized (MADrive) frames using the new "ground-truth"
 863 annotation.

864

865

Table 6: Mean MOTA \uparrow results on test frames for all Waymo scenes.

866

867

| Scene name | SG | HUGS | AutoSplat | MADRIVE |
|------------|-------|-------|-----------|---------|
| 123 | 0.687 | 0.685 | 0.327 | 0.887 |
| 143 | 0.650 | 0.513 | 0.600 | 0.825 |
| 190 | 0.787 | 0.795 | 0.904 | 0.858 |
| 105 | 0.906 | 0.656 | 0.742 | 0.906 |
| 109 | 0.242 | 0.448 | 0.605 | 0.925 |
| 165 | 0.684 | 0.461 | 0.788 | 0.883 |
| 174 | 0.809 | 0.886 | 0.830 | 0.936 |
| 180 | 0.611 | 0.528 | 0.695 | 0.778 |
| 141 | 0.667 | 0.607 | 0.163 | 0.767 |
| 158 | 0.423 | 0.233 | 0.681 | 0.639 |
| 176 | 0.727 | 0.562 | 0.176 | 0.912 |
| 779 | 0.661 | 0.296 | 0.545 | 0.779 |

878

879

880

Table 7: Mean MOTP \downarrow results on test frames for all Waymo scenes.

881

882

| Scene name | SG | HUGS | AutoSplat | MADRIVE |
|------------|-------|-------|-----------|---------|
| 123 | 0.073 | 0.093 | 0.099 | 0.079 |
| 143 | 0.114 | 0.461 | 0.095 | 0.203 |
| 190 | 0.088 | 0.112 | 0.115 | 0.144 |
| 105 | 0.073 | 0.262 | 0.222 | 0.118 |
| 109 | 0.093 | 0.132 | 0.094 | 0.122 |
| 165 | 0.125 | 0.202 | 0.119 | 0.149 |
| 174 | 0.075 | 0.886 | 0.078 | 0.093 |
| 180 | 0.150 | 0.231 | 0.194 | 0.195 |
| 141 | 0.119 | 0.261 | 0.237 | 0.179 |
| 158 | 0.087 | 0.128 | 0.123 | 0.119 |
| 176 | 0.093 | 0.246 | 0.167 | 0.072 |
| 779 | 0.176 | 0.443 | 0.305 | 0.180 |

893

894

895

The results in Table 10 show that the predictions on ground-truth images align slightly better with the projected 3D bounding boxes than those on the synthesized MADrive frames. This indicates that our inserted cars do not artificially simplify detection, supporting the validity of our evaluation.

896

897

898

899

900

F ADDITIONAL RETRIEVAL EVALUATION RESULTS

901

902

903

In this section, we provide an additional illustration of our retrieval algorithm. As shown in our Figure 8, introducing a color-based pre-filter enhances the alignment of vehicle colors between retrieved candidates and the target.

904

905

906

907

908

G EVALUATION OF INSERTED MODEL QUALITY

909

910

911

912

913

914

915

916

917

As mentioned earlier, an alternative to our approach would be to generate car models using a pre-trained image-to-3D generative model. However, such methods typically produce low-resolution cars with limited detail and a cartoon-like appearance. To further assess the viability of this alternative, we compared renderings from both approaches against a hold-out set of real car images from MAD-CARS. Specifically, given car crops from driving scenes, we reconstructed models using our retrieval-based approach and a state-of-the-art image-to-3D method, Amodal3R (Wu et al., 2025b). To reduce domain shift, backgrounds were excluded from MAD-CARS. The results in Table 11 support our claim that the retrieval-augmented approach yields cars with higher resemblance to real ones.

918
919 Table 8: Mean IDF1 \uparrow results on test frames for all Waymo scenes.
920
921
922
923
924
925
926
927
928
929
930
931

| Scene name | SG | HUGS | AutoSplat | MADRIVE |
|------------|-------|-------|-----------|---------|
| 123 | 0.804 | 0.806 | 0.475 | 0.940 |
| 143 | 0.787 | 0.709 | 0.750 | 0.904 |
| 190 | 0.880 | 0.887 | 0.950 | 0.924 |
| 105 | 0.952 | 0.780 | 0.877 | 0.951 |
| 109 | 0.390 | 0.619 | 0.754 | 0.961 |
| 165 | 0.806 | 0.612 | 0.894 | 0.936 |
| 174 | 0.894 | 0.940 | 0.907 | 0.967 |
| 180 | 0.753 | 0.709 | 0.820 | 0.871 |
| 141 | 0.805 | 0.698 | 0.278 | 0.866 |
| 158 | 0.605 | 0.377 | 0.829 | 0.797 |
| 176 | 0.847 | 0.720 | 0.316 | 0.955 |
| 779 | 0.793 | 0.532 | 0.739 | 0.881 |

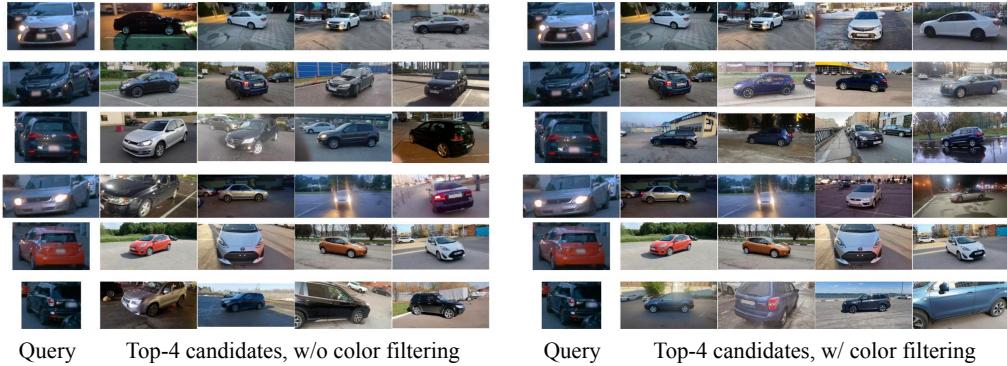
932
933 Table 9: Mean IoU \uparrow results on test frames for all Waymo scenes.
934
935

| Scene name | SG | HUGS | AutoSplat | MADRIVE |
|------------|-------|-------|-----------|---------|
| 123 | 0.753 | 0.608 | 0.500 | 0.866 |
| 143 | 0.485 | 0.243 | 0.510 | 0.779 |
| 190 | 0.707 | 0.519 | 0.740 | 0.846 |
| 105 | 0.671 | 0.425 | 0.439 | 0.731 |
| 109 | 0.499 | 0.246 | 0.419 | 0.832 |
| 165 | 0.633 | 0.459 | 0.647 | 0.730 |
| 174 | 0.695 | 0.581 | 0.655 | 0.829 |
| 180 | 0.475 | 0.238 | 0.582 | 0.814 |
| 141 | 0.607 | 0.196 | 0.226 | 0.765 |
| 158 | 0.404 | 0.135 | 0.498 | 0.862 |
| 176 | 0.499 | 0.187 | 0.263 | 0.886 |
| 779 | 0.247 | 0.153 | 0.273 | 0.874 |

946
947 H ADDITIONAL QUALITATIVE COMPARISONS AND NEW TRAJECTORIES948
949 We provide additional visual results in Figures 9 and 10. We also demonstrate our method’s capabil-
950 ity to render novel views with substantial scene variations. Figure 11 showcases results across four
951 test scenes, where all modifications preserve high image quality.
952953
954 I LIMITATIONS955
956 **Trade-off on seen and unseen data.** Figures 4, 5, 9, 10 show that baseline methods achieve high
957 photometric consistency by optimizing on training frames, while MADRIVE have visual differences.
958959 However, baseline methods significantly degrade on unseen test frames, which is critical for our
960 initial goal. This gap highlights a trade-off of our design: although the visual fidelity is lower at
961 training, our method enables fast and scalable simulation of diverse unseen scenarios.
962963 Importantly, our system is fully automated and requires no human intervention: the retrieval, place-
964 ment, and orientation of car models are all handled automatically. Our validation experiments
965 demonstrate that the generated scenes allow vehicle perception modules to reasonably assess the
966 depicted traffic situations.
967968 **Reconstruction limitations.** To run reconstruction, we estimate camera parameters from the input
969 images. In particular, we run bundle adjustment starting from the initialization obtained with VGQT.
970 At present, errors in camera estimation remain a primary source of reconstruction failures. We ex-
971 pect that continued advances in foundational vision models will substantially reduce this limitation.
972973 State-of-the-art multiview reconstruction methods continue to struggle with reflective and glossy
974 surfaces like cars even up to this day. Accurate modeling of reflections on metallic surfaces on real
975

972
 973 Table 10: Comparison between detector performance on both the ground-truth and synthesized
 974 (MADrive) frames using projected 3D bounding boxes.

| Model | MOTA \uparrow | MOTP \downarrow | IDF1 \uparrow |
|----------------|-----------------|-------------------|-----------------|
| GT frames | 0.879 | 0.270 | 0.928 |
| MADrive frames | 0.861 | 0.340 | 0.908 |



975
 976
 977
 978
 979
 980 Figure 8: **Retrieval illustration.** Top-4 candidates retrieved using SigLIP 2 without (Left) and with
 981 (Right) color filtering.
 982
 983
 984
 985
 986
 987
 988
 989
 990
 991

992 datasets demands more precise representations of illumination - beyond what conventional environment-
 993 maps can provide.

994 J STATEMENT ON LLM USAGE

1000 The authors used the large language model (LLM) only to improve the writing and grammar of the
 1001 text. All the results from the LLM were checked by the authors.

1003
 1004
 1005
 1006
 1007
 1008
 1009
 1010
 1011
 1012
 1013
 1014
 1015
 1016
 1017
 1018
 1019
 1020
 1021 Table 11: Quantitative assessment of car model quality, comparing image-to-3D generative models
 1022 with a retrieval-augmented approach.

| | FID \downarrow | KID $\times 10^3 \downarrow$ |
|----------------|------------------|------------------------------|
| Amodal3R | 81.65 | 51.91 |
| MADrive frames | 62.64 | 39.40 |



Figure 9: **Additional 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**).

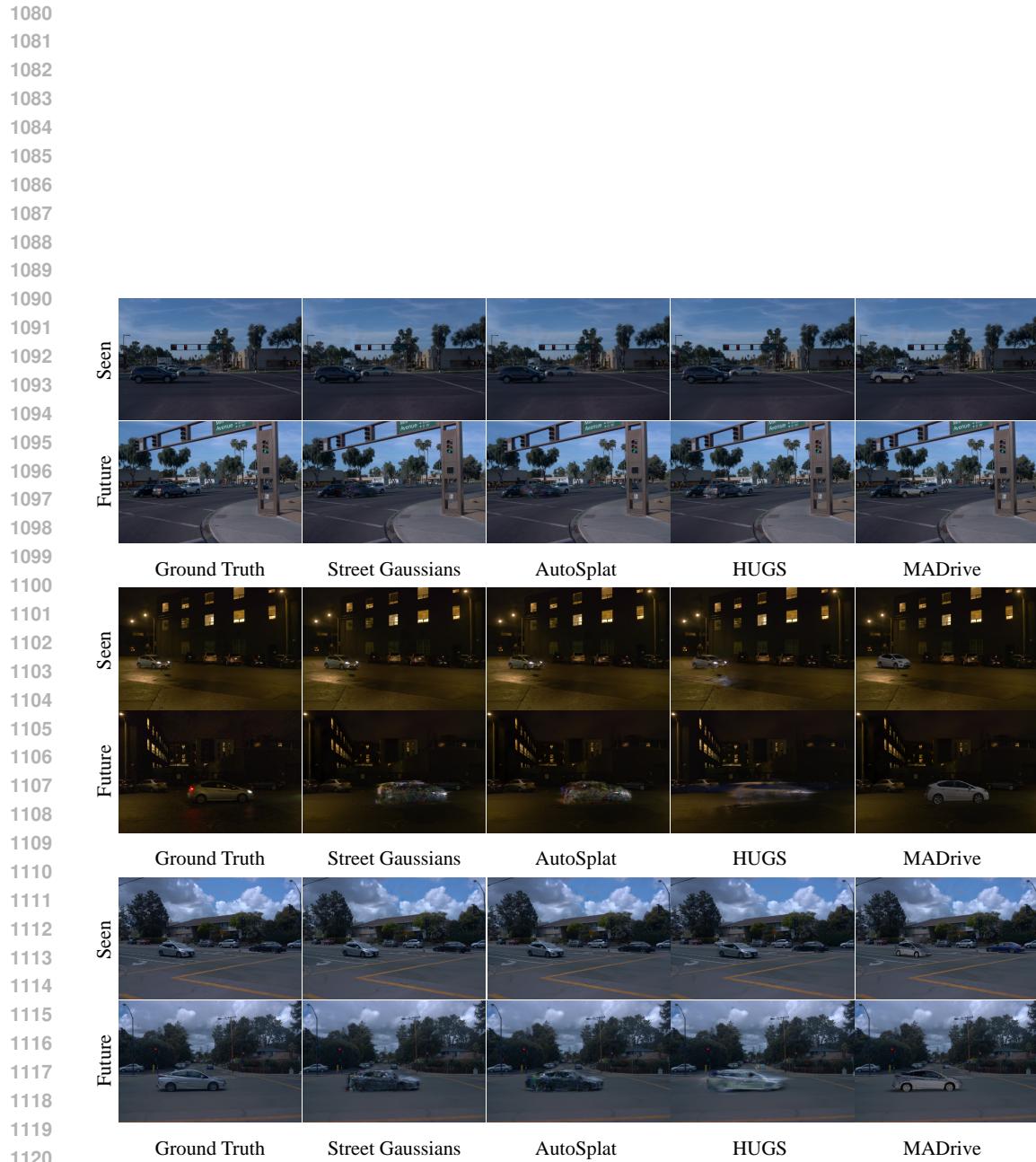


Figure 10: **Additional qualitative comparison** of MADRIVE with non-retrieval-based driving scene reconstruction methods. Reconstruction of the training views (**Top**). Reconstruction of the hold-out (future) views (**Bottom**).

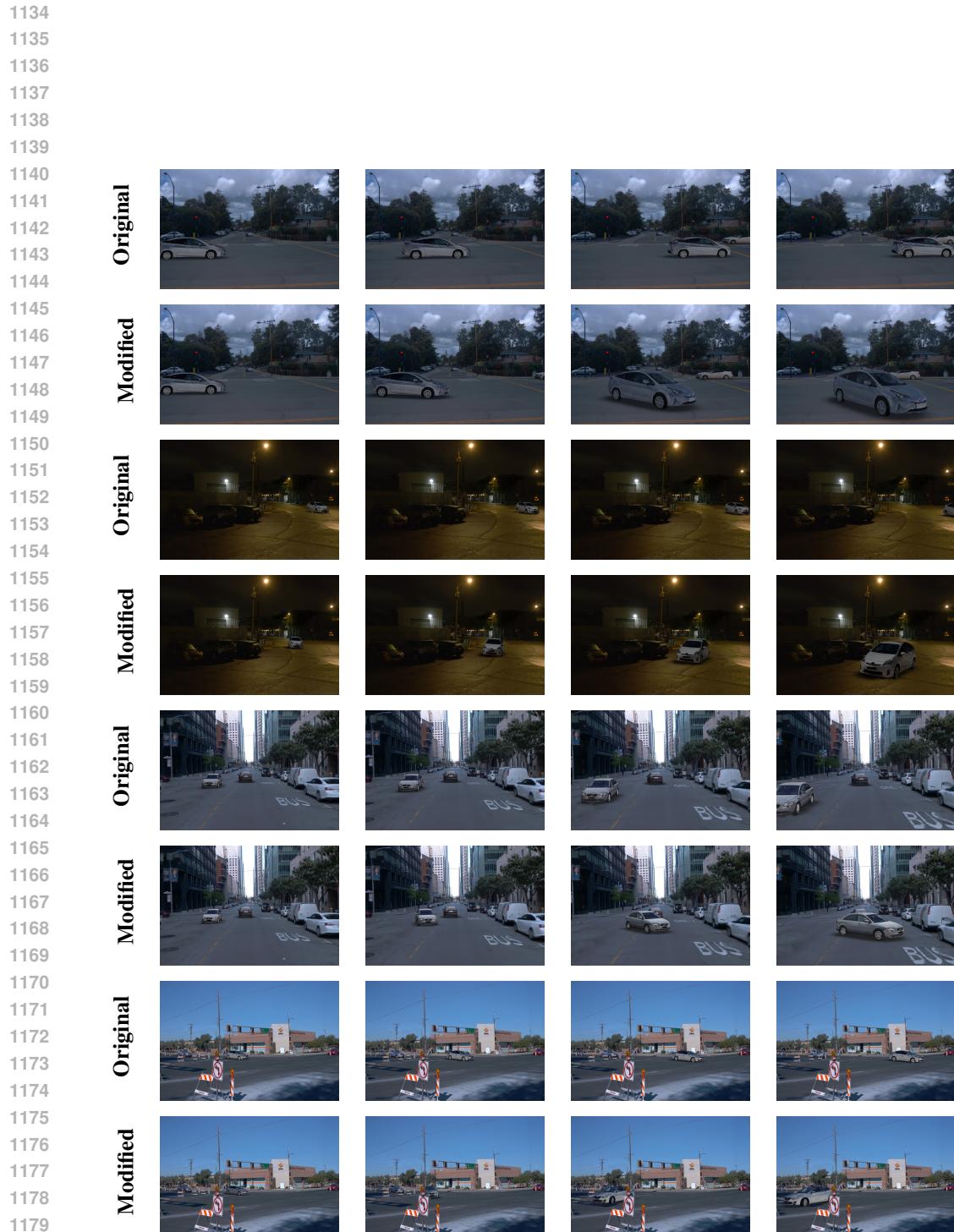


Figure 11: Visualization of original and modified trajectories with MADRIVE. The cars retain high-fidelity appearance even at close distances to the ego camera.

1183
 1184
 1185
 1186
 1187