Material-informed Gaussian Splatting for 3D World Reconstruction in Digital Twin

MSc. Thesis report

Andy Huynh

Supervisors: H. Caesar, J.M. Silva, S. Tong

Monday 15th November, 2025

Delft University of Technology

TU Delft

SIEMENS

Material-informed Gaussian Splatting for 3D World Reconstruction in Digital Twin

by

Andy Huynh

to obtain the degree of Master of Science

at the Delft University of Technology,

to be defended publicly on Friday December 5, 2025 at 11:00 AM.

Student number: 5400759

Thesis committee:

Dr. H. Caesar,	TU Delft, supervisor
J.M. Silva MSc.,	Siemens Digital Software Industries, supervisor
Dr. S. Tong,	Siemens Digital Software Industries, supervisor
Dr. J.F.P. Kooij,	TU Delft

An electronic version of this thesis is available at `http://repository.tudelft.nl/`.

A published version of this thesis is available at `https://arxiv.org/abs/2511.20348`

TU Delft

Material-informed Gaussian Splatting for 3D World Reconstruction in Digital Twin

Andy Huynh[1,2]

***Abstract*—3D reconstruction for Digital Twin has become crucial, providing controlled and scalable environments for developing and validating advanced sensor algorithms before real-world deployment. Traditional reconstruction approaches have several limitations: they often overlook physical rendering properties of materials, lack detailed textures, and rely on lidar sensors that require complex calibration and perform poorly with capturing retro-reflective and transparent surfaces. We propose a modular, camera-centric, and material-informed 3D Gaussian Splatting (3DGS) pipeline for reconstructing large 3D scenes. Our approach extracts semantic material masks using segmentation models, converts Gaussian representations to explicit mesh surfaces, and automatically projects 2D material labels onto 3D geometry. This combines photorealistic reconstruction with physics-based material assignment for accurate sensor simulation and rendering in modern graphics engines and simulators. Evaluation on an internal Siemens autonomous driving dataset demonstrates that our material-informed approach achieves sensor simulation fidelity comparable to lidar-based ground truth while providing high visual fidelity, as validated through image similarity metrics and lidar reflectivity analysis.**

I. Introduction

Digital Twins [15] are integral to the advancement of sensor technologies, facilitating safe validation of high-risk scenarios and precise control over environmental variables, prior to real-world deployment [12]. This is driven by the advancement of sophisticated algorithms used in modern Advanced Driving Assistance Systems (ADAS) and Artificial Intelligence (AI) systems, involving complex multimodal sensor setups, often relying on lidar and cameras. The concept of accurate translation of real-world scenarios, leading to identical behaviour in both simulation and reality is known as real-to simulation (*real2sim*). Within 'real2sim', 3D world reconstruction captures the spatial and geometrical features of real-world assets, aiming to (I) ensure realistic simulation and (II) generate synthetic sensor data consistent with reality. Existing works [33, 62] proposed lidar-based 3D reconstruction methods for static scenes, focusing on reducing geometric discrepancies for accurate synthetic lidar representation. However, these approaches ignore the physical rendering and visual properties of environments, such as materials and textures. Recent efforts [22] address this gap by introducing an automated material assignment pipeline, that combine lidar-based surface reconstruction with camera-based material segmentation. This has contributed to close the real2sim gap in lidar reflectivity while reducing reliance on resource-intensive tools and expert modelling. Despite these advancements, the resulting visual appearance often remains 'game-like', lacking photorealism and failing to achieve plausible reconstructions under sparse-view conditions.

Recent advancements in novel view synthesis have introduced 3D Gaussian Splatting [26], a lightweight model that provides fast, differentiable rendering and high-quality 3D reconstructions with sparse input views. While originally focused on photorealistic rendering rather than accurate surface geometry, recent works such as MILo [16] and GeoSVR [29] demonstrated that high-fidelity geometric surfaces can be extracted from Gaussian Splatting using only multi-view images. This represents a pivotal step for 3D reconstruction, since modern graphics engines and simulations natively only support explicit surface representations, such as meshes [17]. Furthermore, these camera-based methods provide an alternative to lidar hardware, avoiding expensive equipment, intricate sensor synchronization, and demanding calibration procedures.

In light of the above observations, we propose a modular, camera-centric, pipeline that combines combines photorealistic Gaussian Splatting with physics-based material assignment for reconstructing large 3D scenes. Fig 1 illustrates our hybrid pipeline, that combines (A) Gaussian Splatting with (B) mesh-based material rendering for accurate sensor simulation. We validate our method on the BeCareful project dataset from Siemens [1], evaluating our approach by comparing camera-only reconstructions against lidar-based and fusion approaches.
Our main **Contributions** are:

- **(1)** A modular, camera-only pipeline that integrates Gaussian Splatting for photorealistic geometry reconstruction with automated mesh-based material

[1]Dept. Cognitive Robotics of faculty Mech. Eng. (ME), Delft University of Technology, 2628 CD Delft, The Netherlands. Student no. 5400759

[2]Siemens Digital Industries Software, 3001, Leuven, Belgium.

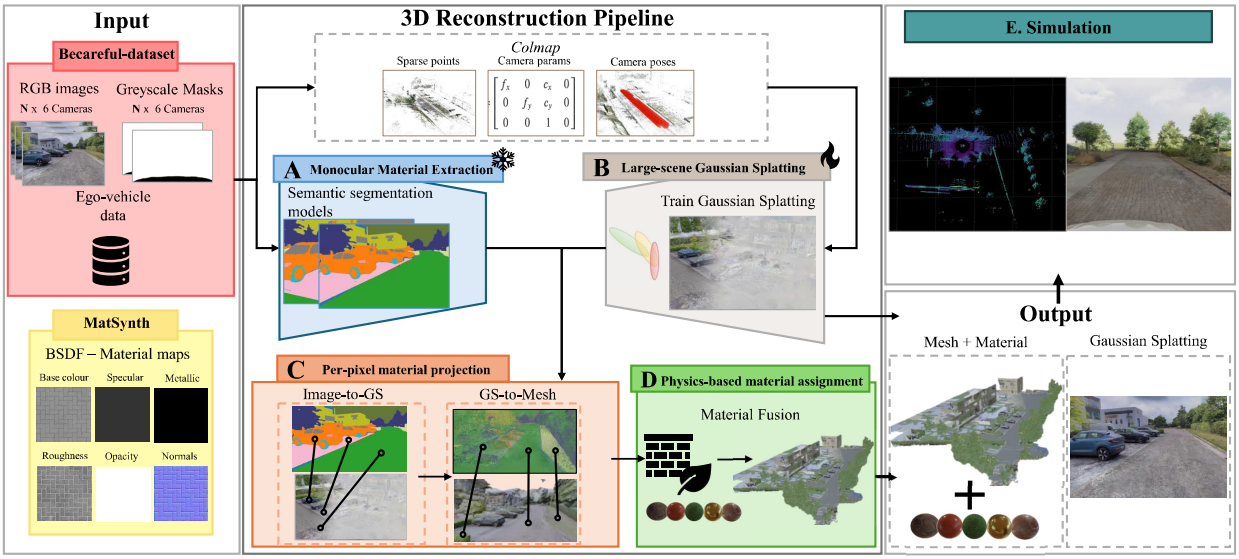


Fig. 1. Overview of our decoupled hybrid reconstruction pipeline. Using only RGB images as input, we subdivide our pipeline into four independent, modular blocks (a) Perform instance segmentation to extract semantic material masks, (b) Adapt a Gaussian Splatting model for large-scene rendering (c) Use semantic masks and the Gaussian Splatting model to project 2D-labelled material masks onto a 3D surface representation (d) Assign a BSDF material to each material-labelled region for accurate physics-based rendering.

assignment, providing an alternative to lidar and complex calibration procedures.

- **(2)** An automated approach for projecting 2D semantic material masks to 3D mesh surfaces through Gaussian Splatting-based reconstruction, facilitating accurate physics-based lidar reflectivity simulation.
- **(3)**: A comprehensive evaluation, comparing camera-only to camera-lidar fusion approaches, demonstrating that our camera-centric, material-informed reconstruction significantly improves simulation rendering quality while achieving sensor simulation fidelity comparable to lidar-based ground truth.

II. Related Work

A. Novel view synthesis

Traditional 3D reconstructions use explicit representations such as point cloud, voxel grids, and polygonal meshes. Methods such as lidar-based reconstruction [21, 30], excel at preserving geometric accuracy and structural integrity. However, these approaches struggle to capture fine visual details and complex surface characteristics. Camera-based alternatives [11, 45] typically offer less geometric accuracy, produce noisier results, and face challenges with transparent and reflective surfaces [42]. The introduction of Neural Radiance Fields (NeRF) [34] has addressed persistent challenges in scene representation, facilitating photorealistic novel view synthesis. Despite developments in scaling NeRF to large scenes [2, 49, 50], NeRF still suffers from high latency and computational overhead, limiting scalability. Additionally, the absence of a discrete surface representation restricts its compatibility with graphics engines and simulations. Recently, alternative approaches such as Gaussian Splatting have aimed to further improve scalability and rendering efficiency [26], prompting several refinements. For instance, Huang et al. [20] proposed collapsing 3D volumes into 2D oriented disks for more accurate geometry, while Zhang et al. [66] and Wei et al. [55] addressed challenges related to semi-transparent assets and shading accuracy, respectively. Wu et al. [57] expanded method compatibility with various camera types and indirect lighting. However, many of these methods remain limited in scene-centric robustness and scalable reconstruction in complex or large environments.

B. Large-scene reconstruction

Building on the foundations of Gaussian Splatting, recent academic work has focused on improving its consistency and accuracy. Many works have pivoted towards small, object-centric models with multi-view imagery, while research towards large-scale, scene-centric outdoor scenarios remain relatively underexplored. Inspired by NeRF-based methods such as Block-NeRF [49] and Mega-NeRF [50], which pioneered parallel large-scale reconstruction via scene partitioning, recent Gaussian Splatting approaches have begun to adopt similar strategies [27, 32]. For these approaches, the scene is divided into smaller, spatially consistent blocks, facilitating

scalable reconstruction. Many models for large scenes employ Level-of-Detail (LoD) techniques [27, 32], allocating higher resolution to visually important or nearby regions while approximating distant areas with lower detail. This improves the overall rendering speed and reducing memory usage. Beyond spatial partitioning, alternative strategies such as temporal segmentation [10]) and distance-based decomposition [46]) have been proposed to further boost efficiency and scalability. Despite these advancements, effective hyper-parameter tuning remains crucial for achieving optimal reconstruction quality in large-scale, Gaussian-based scene reconstruction.

C. Surface Extraction from Gaussian Primitives

While Gaussian Splatting offers high-fidelity visuals and fast rendering, simulation engines require explicit meshes for physics, collisions, and material properties. Recent advances allow extracting surfaces from Gaussians: Guedon et al. [14] aligned 3D Gaussians with surfaces using volumetric fields and Poisson reconstruction [25]. Other approaches applied Signed Distance Functions (SDFs) [8], though they come with dependencies on multi-view data and face lighting limitations. Ray-Gaussian intersections [63] offer unbiased depth estimation, but trade geometric accuracy for added data requirements. MILo [16] integrates mesh extraction into GS training via Delaunay triangulation [23], preserving fine details. Beyond Gaussians, alternative representations : voxel-based methods [44, 48], convex polygons [18] and triangle-based approaches [17], facilitating sharp, engine-ready surfaces. Furthermore lidar-based reconstructions [21, 30] still yield highly accurate geometry.

D. Material extraction

Material and texture recognition has recently attracted significant attention, as it directly influences how assets interact with their environment in simulations through reflection, absorption, and collisions. These factors are critical for validating sensors such as lidar and radar [68]. Bell et al. [3] demonstrated material segmentation for images using fully convolutional networks, framing the task as semantic segmentation into material categories. Cai et al. [6] has demonstrated per-pixel image material segmentation for real-world driving scenes using a Transformer-based framework that also recognises textures. Liang et al. [31] extended this approach by integrating polarization and near-infrared light, thereby strengthening material prediction. Upchurch et al. [51] leveraged 'big data' to create a large-scale, densely annotated material segmentation dataset, encompassing 46 material categories across both indoor and outdoor scenes. Viswanath et al. [53] proposed classifying materials via LiDAR semantic segmentation using calibrated reflectivity, which correlates with material type. Janda et al. [24] utilised a flash lidar sensor, analysing the shape of returned waveforms to learn material properties. Zhu et al. [69] applied large language models to radar data, correlating electromagnetic wave characteristics with material identification.

E. Physics-based materials for rendering

Modern graphics engines use physics-based rendering to simulate surface interactions such as reflection and absorption through mathematical models such as the Bidirectional Reflectance Distribution Function (BRDF) [28]. Approaches for modelling real-world materials can be categorised as (I) analytical models, which are widely available but have reduced accuracy [40]. (II) Measured model offer high accuracy but are infeasible for scalability [56]. (III) Data-driven methods combine both models and use neural networks to learn from lab measured examples. Many engines favour simplified standards such as the Disney-based BSDF [4], trading physical accuracy for ease-of-use, which can limit realism in simulations. Recent work integrates material properties with 3D Gaussians for physics-based rendering [58, 61], providing more organic deformations and relightable scenes. Challenges remain in modelling metallic and reflective/transparent surfaces. Recent works address these limitations by embedding pixel-level properties in Gaussians [36, 60] or apply ray tracing for enhanced realism at the expense of higher computational cost [35]. Mesh-compatible outputs are also being developed [59], making integration with standard engines more seamless.

III. Method

This section will break down our decoupled hybrid reconstruction pipeline, as illustrated in Fig. 1. We first cover how we generate semantic material labels (a), and how we train a large-scale Gaussian splatting model(b). We highlight how we use both outputs to project 2D-labelled images to 3D representations(c). We will cover how we apply individual Physics-based rendering (PBR) materials for our reconstruction(d), followed up by our scenario reconstruction using Simcenter Prescan [47](e).

A. Monocular Material Extraction

This section presents our methodology for extracting per-pixel 2D material labels exclusively from RGB images. To achieve accurate scene representation, it is essential to employ robust material segmentation. Following a comprehensive evaluation of several segmentation models, we adopted RMSNet [6] as our baseline. RMSNet demonstrates strong performance

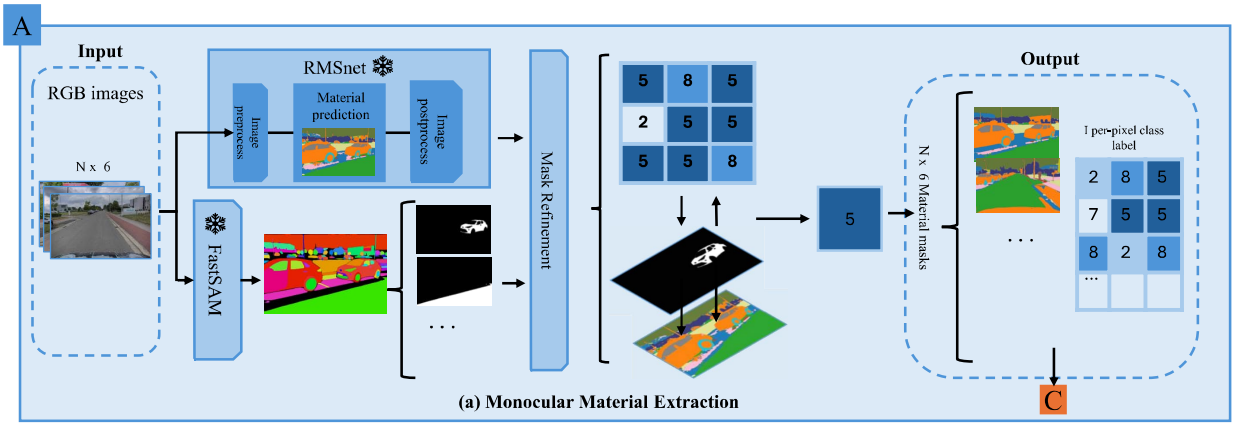


Fig. 2. We use RMSNet [6] to generate semantic material masks and FastSAM to create semantic object masks. Each object mask is overlapped with every semantic material mask, and the most dominant material is assigned to each pixel. This results in a shape-aware material mask.

on material classification and segmentation tasks specialised for autonomous driving applications. We refine the initial semantic masks using foundational models to obtain detailed, pixel-wise labels, as illustrated in Fig.2. Notably, RMSNet operates solely on RGB images, minimising system complexity and eliminating dependency on additional sensor modalities such as lidar or infrared[31, 53]. The network segments material regions by analysing local texture patterns, enabling precise classification for downstream tasks.

Materials often exhibit more fragmented spatial distributions and lack the prominent shape cues characteristic of semantic objects. We observed that traditional approaches failed to capture these shape cues. To address this, we employ Fast Segment Anything (FastSAM) [67], an instance segmentation tool trained on 1.1 billion masks across 11 million images, to segment all semantic objects within a scene. Instance segmentation is applied to each RGB image, to obtain pixel-wise object boundaries. We post-process the masks to remove overlapping masks. Each object mask is then overlaid with the corresponding RMSnet material mask. Using majority voting within each object region, we assign the dominant material label to create material-segmented masks with clearly defined object shapes, as illustrated in Fig. 3. The results is a set of n material segmentation masks, where each pixel inherits its material class according to the SAM-derived object segmentation.

B. Large-scene Gaussian Splatting

For photorealistic 3D reconstruction, we employ Hierarchical 3D Gaussian Splatting (H3DGS) [27]. We have selected H3DGS over alternatives such as CityGaussian [32], which is primarily tailored for aerial imagery, because H3DGS demonstrates great performance on street-level autonomous driving datasets. The reconstruction pipeline starts with COLMAP [45], which performs Structure-from-Motion (SfM) to generate a sparse 3D point cloud from multi-view RGB images. This point cloud serves as the initial positions for the 3D Gaussians. To avoid artifacts caused by the moving ego-vehicle, we mask it in all input images using the segmentation masks. H3DGS initially constructs a global scaffold, then subdivides the scene into spatial chunks derived from this scaffold. However, its hierarchical level-of-detail representation and anti-aliasing mechanisms are not compatible with standard 3D Gaussian Splatting tools, such as common viewers and renderers. We therefore disable these features, keeping only the chunk-based representation in a conventional format to ensure broad compatibility.

C. Per-pixel material projection

This module combines semantic material masks extracted from 2D images with the Gaussian Splatting (GS) model. The process involves two main steps: (I) projecting material masks to 3D Gaussians; (II) transferring these labeled Gaussians to a mesh representation. This produces a reconstructed surface segmented by material labels, as illustrated in Fig.4. Unlike other works relying on camera-to-lidar transformations, our 3D GS model is natively aligned with world coordinates. We employ differentiable rasterization[70] to render 3D Gaussians into 2D representations using the projective transformation $\Sigma' = JW\Sigma W^T J^T$. This maintains correspondence between 3D Gaussians and 2D pixels by tracking the 3D location μ through rasterisation. When

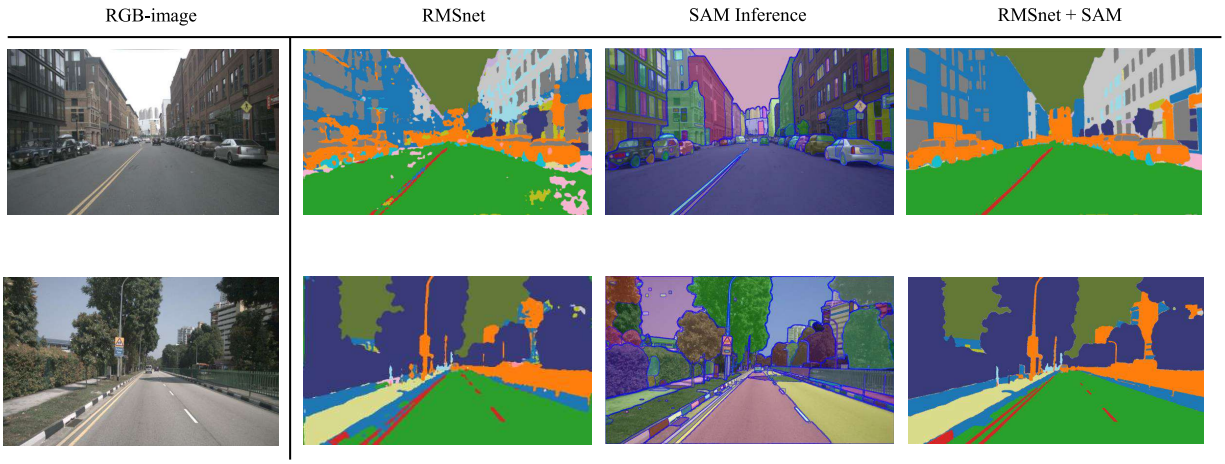


Fig. 3. Leveraging SAM to refine material prediction by defining semantic object shapes on selected nuScenes data [5]

multiple Gaussians project to the same pixel, depth sorting during alpha blending ensures the closest Gaussian label is selected. Inconsistencies in material masks across views, such as homogeneous objects containing noisy labels from multiple material classes, are addressed using SegAnyGS [7] with SAM-based 3D instance segmentation [41]. Majority voting is applied across overlapping masks for label consistency. For mesh compatibility, the GS model is normalised to real-world coordinates and aligned with a mesh, trained from MiLO [16]. Each material-labelled Gaussian is assigned to its nearest mesh triangle via a KNN algorithm [65], and the triangle inherits the Gaussian's label, producing a segmented material mesh as the final output.

D. Physics-based Material assignment

This module applies physics-based material properties in the mesh using BSDF material maps. Each mesh region is assigned a specific set of material-dependent maps. We employ the Principled BSDF shader, following the Disney BSDF standard [4]. This ensures native compatibility with graphics engines such as Blender. The Principled BSDF is parameterised by properties including base colour, specular/diffuse balance, roughness, metallic factor, and transparency to facilitate consistent, lab-verified material representation. Materials are sourced primarily from the Matsynth dataset [52], offering over 4,000 materials across 14 material categories, and NVIDIA vMaterials [37]. Both datasets provide Physics-based rendering parameter data, that are compatible with the lasest graphics engines and simulations. The final output is a reconstructed mesh with material-dependent rendering properties, directly usable by standard graphics engines.

E. Simulation

This section describes our simulation setup for evaluating the PBR-material infused 3D reconstruction, as illustrated in Fig. 1. Scenes from our dataset are reconstructed and integrated into simulation tools such as the Siemens Simcenter Prescan environment to recreate realistic traffic scenarios. The reconstructed models are processed using the Model Preparation tool, which allows assignment of physically measured materials to ensure accurate interaction with sensors, including lidar. Ego-vehicle trajectory data is incorporated via the Simcenter Prescan MATLAB API, ensuring precise positioning and motion within each scenario. This setup allows comprehensive assessment of the scenes under different poses and conditions. The resulting traffic scenarios yield synthetic lidar data with material specific reflectivity information, facilitating effective validation and benchmarking of our 3D reconstruction and material assignment approach.

IV. Experiment

We highlight the primary advantages of our pipeline, designed to be (I) modular for rapid adaptation to emerging methods, and (II) camera-centric, eliminating dependence on sensors such as lidar, to reducing system complexity. To validate the performance of our work, we compare our camera-centric reconstruction against our lidar-camera based approach. We validate the render quality of our Gaussian Splatting using image similarity metrics and validate our sensor accuracy by comparing ground truth lidar reflectivity with our synthetic lidar reflectivity.

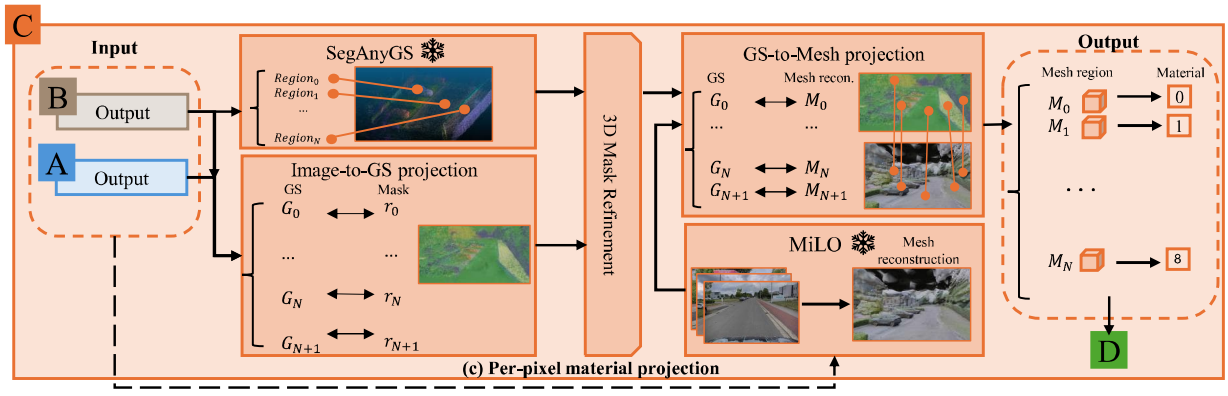


Fig. 4. We leverage material masks(a) and the Gaussian Splatting model(b) and project 2D material-labels through GS to a mesh reconstruction, which are natively supported by graphics engines

A. Dataset

We validate our static 3D reconstruction using a BeCareful project dataset from Siemens [1], collected with a KIA EV6 test vehicle. We have provided an comprehensive overview of our Becareful setup in Appendix A. For the validation, we captured five road scenes in Leuven (BE), each with an average duration of eight seconds, with no dynamic objects or actors present. In addition to point positions, our lidar data records calibrated intensity and reflectivity, representing the returned energy upon contact with the surface, enabling detailed evaluation of material assignment and surface properties.

B. Baseline

In this section, we outline the chosen baselines for validating our 3D Reconstruction methodology and provide key implementation details.

Render quality: We evaluate the render quality by comparing 2D images generated from our Gaussian Splatting approach with ground truth images. We perform this by using established image similarity metrics. For fair benchmarking, we include three leading 3D Gaussian Splatting reconstruction methods [26, 32, 57] as baselines. All models are trained and tested on the five scenes from the BeCareful project dataset, with 50 validation samples each. Performance is assessed by rendering and analysing novel views from each method.

Sensor lidar accuracy: We evaluate the sensor lidar accuracy by focussing on calibrated intensity (reflectivity) values, as these reflect the returned energy influenced by scene materials. We reconstruct each environment in Simcenter Prescan [47], generating synthetic lidar data to compare against the ground truth. As baseline, we include the lidar-camera material assignment methodology proposed by Huynh et al. [22]. To ensure a fair comparison, both approaches are assigned identical material-dependent PBR properties, minimising bias from material differences. This setup allows a direct assessment of how each reconstruction method represents calibrated lidar intensity.

C. Implementation Details

Simulation: We configure the Prescan lidar according to the specifications of the Ouster-OS1 128 [38] used in our dataset. The physics-based sensor returns both the Cartesian coordinates of detected points, and a 'power' output. This output is specified as a non-vendor specific value that represents the received power of the reflected signal.

Hardware: The sensor validation is performed with Siemens Simcenter Prescan 2411. Scene reconstruction leverages MATLAB Simulink R2023B. All experiments are conducted on a Dell Precision 7680 equipped with 64GB RAM and an NVIDIA RTX A4000 ADA Laptop GPU.

D. Metrics

Image Similarity Metrics

To quantitatively assess the render quality of our 3D Gaussian Splatting model, we apply standardised image similarity metrics: Peak Signal-to-Noise Ratio (PSNR) [13], Structural Similarity Index Measure (SSIM) [54], and Learned Perceptual Image Patch Similarity (LPIPS) [64]. We refer to Appendix B for a detailed study how these values are determined. The indicated metrics are widely used in large-scene neural rendering evaluations and collectively validate pixel-level and perceptual accuracy:

- **PSNR** quantifies the noise and distortion between rendered and ground truth images, providing a measure of pixel-wise fidelity.

Fig. 5. **Left:** Our model demonstrates competitive performance, exhibiting minimal deviation from ground truth in rendering quality. It effectively fills masked areas and removes noise present in the ground truth (green). **Right:** Despite these strengths, the model shows limitations when reconstructing highly reflective surfaces (red). Consistency from non-front-facing cameras is also reduced, mainly due to increased image blur in those viewpoints.

- **SSIM** evaluates perceptual quality by comparing luminance, contrast, and structural information, reflecting features important to human perception.
- **LPIPS** compares high-level features (texture, shape, and structure) using deep neural network representations, capturing perceptual similarities that closely align with human visual judgments.

Lidar Reflectivity

We leverage lidar reflectivity (calibrated intensity) as metric for our simulation-based evaluation of our material-aware reconstruction. The simulated 'power' output from Prescan represents the energy reflected back to the sensor upon contact with a surface. To ensure valid material assessment, the simulated lidar power is normalised for both range and incidence angle, yielding accurate surface reflectivity values. This conversion follows established approaches from Prescan case studies[19, 39]. A more detailed description of our calibration procedure is provided in Appendix C.

E. Camera quality: Image similarity

We have evaluated our adapted H3DGS Gaussian Splatting model against other leading 3D Gaussian Splatting approaches by comparing renderings on novel views. We present our qualitative comparison with benchmark models in Appendix D We observe that our model demonstrates remarkable performance in terms of perceptual quality. Specifically, the rendered images retain higher resolution relative to the three other benchmark models. Fig. 5 illustrates how our model is capable to fill masked areas or remove noise that may be present in the ground truth. However, our method still faces challenges with reflective and transparent surfaces, which can result in occasional misalignments with the ground truth. We also observe that our model performs best for the front camera, while side cameras are more affected by motion blur, leading to a reduced render quality. Additionally, complex scenes containing multiple assets can negatively impact render performance. To support these qualitative findings, we conduct a quantitative evaluation. Table I summarises the comparative metrics on novel views. We use three complementary measures (see Sec.IV-D): PSNR, SSIM, and LPIPS, each capturing different aspects of visual quality. The results reveal that although the adapted H3DGS scores lower on PSNR and SSIM than baseline methods, it consistently achieves better results on the LPIPS metric. This is likely due to specific architectural changes, namely, the removal of anti-aliasing and the simplification of hierarchical optimization, which influence RGB composition at the pixel level and increase sensitivity to metrics based on direct pixel-wise similarity. Most importantly, a higher LPIPS indicates greater similarity as perceived by the human eye, making it a particularly relevant metric for our application, where perceptual quality is central.

F. Lidar quality: Reflectivity metrics

We evaluate our camera-centric Gaussian-assigned material reconstruction against the lidar-camera assigned material reconstruction [22], using the reflectivity deviation from the ground truth as our metric. A qualitative comparison between our methodology and the ground truth is provided in Appendix E. Both the lidar-based and our camera-centric models exhibit some deviation in material reflectivity compared to the ground truth. Notably, the vegetation material class consistently shows higher reflectivity than the ground truth, which may be attributed to the rendering properties of the PBR material used, as vegetation can appear in various

TABLE I
IMAGE SIMILARITY METRICS (PSNR ↑, SSIM ↑, LPIPS ↓) EVALUATED ON NOVEL VIEWS FOR ALL BASELINES.

	H3DGS (adapted) - Ours			3DGS			CITYGS (V2)			3DGUT		
	PSNR↑	**SSIM↑**	**LPIPS↓**	**PSNR↑**	**SSIM↑**	**LPIPS↓**	**PSNR↑**	**SSIM↑**	**LPIPS↓**	**PSNR↑**	**SSIM↑**	**LPIPS↓**
d1	16.88	0.299	0.451	19.13	0.476	0.438	20.36	0.486	0.480	19.97	0.465	0.517
d2	17.86	0.358	0.432	19.81	0.525	0.474	20.79	0.526	0.522	21.16	0.505	0.512
d3	18.35	0.375	0.437	19.48	0.515	0.492	20.42	0.524	0.540	21.10	0.502	0.557
d4	19.03	0.504	0.439	20.88	0.659	0.450	21.15	0.676	0.483	21.85	0.666	0.526
d5	19.21	0.503	0.427	21.04	0.650	0.446	21.63	0.690	0.499	22.04	0.662	0.520
Avg	**18.27**	**0.41**	**0.44**	**20.07**	**0.57**	**0.46**	**20.87**	**0.58**	**0.50**	**21.22**	**0.56**	**0.53**

PSNR: Peak Signal-to-Noise Ratio; SSIM: Structural Similarity Index Measure; LPIPS: Learned Perceptual Image Patch Similarity. For each metric, ↑ indicates that higher values are better, while ↓ indicates that lower values are preferred. In each row, the best performance per metric is highlighted in red, and the second-best in yellow. Notably, our model achieves the highest scores in LPIPS, which is particularly significant since LPIPS is most closely aligned with human perceptual similarity—an essential focus of our GS model.

TABLE II
QUANTITATIVE LIDAR REFLECTIVITY EVALUATION BETWEEN CAMERA-GS AND LIDAR-CAMERA

	Camera-GS		Lidar-Camera	
	Mean↓	Median↓	Mean↓	Median↓
d1	10.91	7.16	11.73	7.84
d2	9.37	5.54	10.17	6.70
d3	11.03	6.91	10.47	6.87
d4	9.32	5.77	9.41	6.02
d5	8.98	6.71	8.76	6.36
Avg	**10.05**	**6.48**	**10.14**	**6.78**

We compare both of our models using reflectivity values in the range 0–255. Our Camera-GS reconstruction shows a slight advantage over the lidar-camera methodology.

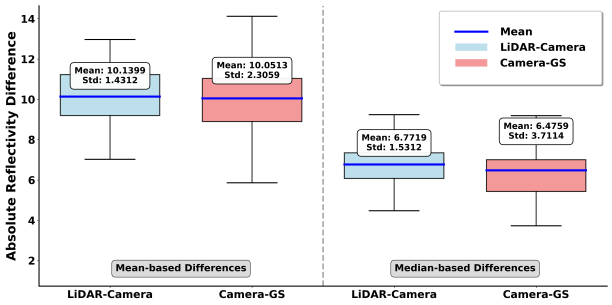


Fig. 6. The absolute reflectivity deviation with respect to the Ground Truth is evaluated for both synthetic lidar from our lidar-camera method and the Camera-GS reconstruction model. Results show that our Camera-GS model has a slight advantage over the lidar-camera approach.

forms. Additionally, it is apparent that the camera-derived mesh contains more noise than the lidar phase, potentially affecting the overall geometric fidelity of the scene. To support our qualitative findings, we perform a quantitative evaluation. Table II and Fig. 6 visualise our reflectivity results. To ensure unbiased results, we evaluate performance across five different datasets. Our results indicate that the camera-based Gaussian Splatting (camera-GS) method achieves reflectivity accuracy that is at least comparable to, and slightly better than, the lidar-camera-based approach. While this advantage is modest, it highlights the effectiveness of our camera-based method, even without lidar assistance. However, the higher standard deviation observed for camera-GS suggests greater variability, likely due to the less refined mesh compared to the lidar-supported model. Overall, these findings demonstrate that camera-GS is a competitive alternative and reinforce its potential for applications where lidar data may not be available or practical.

G. Ablation studies

In this ablation study, we highlight the addition of FastSAM to improve our overall per-pixel accuracy of our semantic material masks, as we have covered in Sec. III-A We quantitatively assess the impact of integrating FastSAM, with results detailed in Table III. To support our qualitative findings, we use the MCubes multi-modal dataset, which contains 500 material-annotated images [31]. A comprehensive analysis is provided in Appendix F. Our evaluation shows that adding FastSAM to our semantic masks increases overall accuracy by 2% compared to using only the RMSnet raw model. We observed that incorporating more masks generally enhances segmentation performance, but this comes at the expense of increased computational time and resource usage. Additionally, we found that using masks to assist with shape cues does not always improve per-pixel accuracy, as fine details—such as road markings—can be overwritten. After carefully considering these factors, we conclude that FastSAM does improve overall per-pixel accuracy. However, this improvement

must be balanced against the higher computational cost associated with more complex object segmentation. For this reason, we chose FastSAM over SAM2. It is important to note, however, that per-pixel accuracy did not improve dramatically; in fact, if an incorrect prediction is assigned to a larger region due to the shape guidance, it can actually lower per-pixel accuracy by spreading the error over a wider area than the original misprediction.

TABLE III
PERFORMANCE COMPARISON ON MATERIAL SEGMENTATION DATASET

Method	Accuracy (%)	mIoU (%)
RMSNet	0.63	0.28
RMSNet + FastSAM	**0.65**	**0.29**
RMSNet + SAM2	0.67	0.31

Performance comparison of per-pixel accuracy scores, illustrating the effect of incorporating SAM to enhance shape cue understanding.

V. CONCLUSION

A. Conclusion

In this paper, we present a novel, camera-centric, and modular approach to 3D reconstruction of large scenes using material-informed 3D Gaussian Splatting. Our hybrid strategy combines Gaussian Splatting with mesh-based material rendering, enabling fast visualizations, accurate geometric reconstruction, and physically realistic material rendering. The modular design of our pipeline allows for the straightforward replacement of individual components with advanced methods as they emerge. We introduce a fully camera-centric workflow that builds the entire 3D reconstruction using only RGB images, reducing reliance on lidar and simplifying the overall process. Our experiments demonstrate that incorporating FastSAM enhances monocular material segmentation accuracy by 2%. Additionally, our pipeline achieves the highest LPIPS score compared to other recent benchmarks for novel view rendering, highlighting its ability to produce high-fidelity, realistic images. In terms of reflectivity, our results show that our approach matches or slightly outperforms lidar-camera-based methods, underscoring the effectiveness of our model. Overall, this work represents a significant leap towards bridging the gap between reality and simulation for Digital Twin.

B. Limitations

Although our pipeline is capable of producing high-fidelity 3D reconstructions, several limitations remain. While FastSAM assists RMSnet in defining object boundaries, it does not resolve mispredicted masks. Training meshes using only camera images represents a significant advancement, but these meshes still lack the geometric precision (noise) of those generated with lidar data. In the realm of physics-based material rendering, we encountered constraints such as limited access to accurate measured materials; moreover, available PBR libraries are primarily designed for visual effects or animation, often prioritising intuition over strict physical accuracy [4], which can introduce errors. Additionally, material specification is challenging, as materials often present diverse appearances, such as varying vehicle colours or different types of concrete and brick. Finally, although we employed a calibration method to convert prescan power units to reflectivity, the results are approximate due to undisclosed lidar sensor properties and calibration error, meaning calculated reflectivity may diverge from actual physical values.

C. Discussion

Our future work will focus on overcoming the identified limitations to make material-informed Gaussian Splatting-based 3D reconstruction more robust. We plan to enhance monocular material extraction further, while adding FastSAM improved the delineation of semantic object regions, it did not fully address incorrect mask predictions. To improve overall material prediction and mitigate intra-class variance, we propose incorporating vision-language approaches [43]. We also aim to reduce pipeline complexity by integrating currently separate components, such as the Gaussian splatting and mesh models, into a fully unified system. Although our camera-based rendering achieves strong results, challenges remain, including lower performance from side cameras and motion-induced image blur, which we intend to address. Finally, while our reflectivity results are nearly on par with lidar-based methods, the geometric accuracy of camera-Gaussian Splatting meshes still lags behind lidar reconstructions, and improving this aspect will be a priority.

REFERENCES

[1] BECAREFUL – Belgian Consortium for Enhanced Safety & Comfort Perception of Future Autonomous Vehicles. Funded by the Flemish Agency for Innovation and Entrepreneurship (VLAIO), Project No. HBC.2021.0939, 2021. Acknowledged in: ExAMPC: the Data-Driven Explainable and Approximate NMPC with Physical Insights.

[2] J. T. Barron, B. Mildenhall, M. Tancik, P. Hedman, R. Martin-Brualla, and P. P. Srinivasan. Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5855–5864, 2021.

[3] S. Bell, P. Upchurch, N. Snavely, and K. Bala. Material recognition in the wild with the materials in context database, 2015.

[4] B. Burley and Walt Disney Animation Studios. Physically-based shading at disney. In *Acm siggraph*, volume 2012, pages 1–7. vol. 2012, 2012.

[5] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom. nuscenes: A multimodal dataset for autonomous driving, 2020.

[6] S. Cai, R. Wakaki, S. Nobuhara, and K. Nishino. Rgb road scene material segmentation. In *Proceedings of the Asian Conference on Computer Vision*, pages 3051–3067, 2022.

[7] J. Cen, J. Fang, C. Yang, L. Xie, X. Zhang, W. Shen, and Q. Tian. Segment any 3d gaussians, 2025.

[8] H. Chen, C. Li, Y. Wang, and G. H. Lee. Neusg: Neural implicit surface reconstruction with 3d gaussian splatting guidance, 2025.

[9] A. Christodoulides, G. K. L. Tam, J. Clarke, R. Smith, J. Horgan, N. Micallef, J. Morley, N. Villamizar, and S. Walton. Survey on 3d reconstruction techniques: Large-scale urban city reconstruction and requirements. *IEEE Transactions on Visualization and Computer Graphics*, pages 1–20, 2025.

[10] X. Cui, W. Ye, Y. Wang, G. Zhang, W. Zhou, and H. Li. Streetsurfgs: Scalable urban street surface reconstruction with planar-based gaussian splatting, 2024.

[11] Y. Furukawa, B. Curless, S. M. Seitz, and R. Szeliski. Towards internet-scale multi-view stereo. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1434–1441, 2010.

[12] L. Gherardini and G. Cabri. The impact of autonomous vehicles on safety, economy, society, and environment. *World Electric Vehicle Journal*, 15(12), 2024.

[13] R. C. Gonzalez and R. E. Woods. *Digital Image Processing*. Pearson Education, 3rd edition, 2008.

[14] A. Guédon and V. Lepetit. Sugar: Surface-aligned gaussian splatting for efficient 3d mesh reconstruction and high-quality mesh rendering. *CVPR*, 2024.

[15] J. Guo and Z. Lv. Application of digital twins in multiple fields. *Multimedia Tools and Applications*, 81(19):26941–26967, 2022.

[16] A. Guédon, D. Gomez, N. Maruani, B. Gong, G. Drettakis, and M. Ovsjanikov. Milo: Mesh-in-the-loop gaussian splatting for detailed and efficient surface reconstruction, 2025.

[17] J. Held, R. Vandeghen, A. Deliege, A. Hamdi, S. Giancola, A. Cioppa, A. Vedaldi, B. Ghanem, A. Tagliasacchi, and M. Van Droogenbroeck. Triangle splatting for real-time radiance field rendering, 2025.

[18] J. Held, R. Vandeghen, A. Hamdi, A. Deliege, A. Cioppa, S. Giancola, A. Vedaldi, B. Ghanem, and M. Van Droogenbroeck. 3d convex splatting: Radiance field rendering with 3d smooth convexes, 2024.

[19] N. Hermidas and M. Phillips. Physics based lidar power calibration. Technical report, Siemens Industry Software Netherlands B.V., The Hague, The Netherlands, May 2021. Published: May 21, 2021.

[20] B. Huang, Z. Yu, A. Chen, A. Geiger, and S. Gao. 2d gaussian splatting for geometrically accurate radiance fields. In *Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers '24*, SIGGRAPH '24, page 1–11. ACM, July 2024.

[21] J. Huang, Z. Gojcic, M. Atzmon, O. Litany, S. Fidler, and F. Williams. Neural kernel surface reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4369–4379, 2023.

[22] A. Huynh. Internship: Real2sim: High-fidelity traffic scenario reconstruction using materials and textures, 2025.

[23] Y. Ito. *Delaunay Triangulation*, pages 332–334. Springer Berlin Heidelberg, Berlin, Heidelberg, 2015.

[24] A. Janda, P. Merriaux, P. Olivier, and J. Kelly. Living in a material world: Learning material properties from full-waveform flash lidar data for semantic segmentation. In *2023 20th Conference on Robots and Vision (CRV)*, page 202–207. IEEE, June 2023.

[25] M. Kazhdan, M. Bolitho, and H. Hoppe. Poisson surface reconstruction. In *Proceedings of the fourth Eurographics symposium on Geometry processing*, volume 7, 2006.

[26] B. Kerbl, G. Kopanas, T. Leimkühler, and G. Drettakis. 3d gaussian splatting for real-time radiance field rendering, 2023.

[27] B. Kerbl, A. Meuleman, G. Kopanas, M. Wimmer, A. Lanvin, and G. Drettakis. A hierarchical 3d gaussian representation for real-time rendering of very large datasets. *ACM Trans. Graph.*, 43(4), July 2024.

[28] M. KURT. A survey of bsdf measurements and representations. *Journal of Science and Engineering*, 20(58):87–102, 2018.

[29] J. Li, J. Zhang, Y. Zhang, X. Bai, J. Zheng, X. Yu, and L. Gu. Geosvr: Taming sparse voxels for geometrically accurate surface reconstruction, 2025.

[30] Z. C. Li, W. Sun, S. Govindarajan, S. Xia, D. Re-

bain, K. M. Yi, and A. Tagliasacchi. Noksr: Kernel-free neural surface reconstruction via point cloud serialization. In *2025 International Conference on 3D Vision (3DV)*, pages 567–574. IEEE, 2025.

[31] Y. Liang, R. Wakaki, S. Nobuhara, and K. Nishino. Multimodal material segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19800–19808, June 2022.

[32] Y. Liu, C. Luo, Z. Mao, J. Peng, and Z. Zhang. Citygaussianv2: Efficient and geometrically accurate reconstruction for large-scale scenes, 2025.

[33] S. Manivasagam, S. Wang, K. Wong, W. Zeng, M. Sazanovich, S. Tan, B. Yang, W.-C. Ma, and R. Urtasun. Lidarsim: Realistic lidar simulation by leveraging the real world, 2020.

[34] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng. Nerf: Representing scenes as neural radiance fields for view synthesis, 2020.

[35] N. Moenne-Loccoz, A. Mirzaei, O. Perel, R. de Lutio, J. Martinez Esturo, G. State, S. Fidler, N. Sharp, and Z. Gojcic. 3d gaussian ray tracing: Fast tracing of particle scenes. *ACM Transactions on Graphics and SIGGRAPH Asia*, 2024.

[36] D. M. Nguyen, M. Avenhaus, and T. Lindemeier. Gs-2m: Gaussian splatting for joint mesh reconstruction and material decomposition, 2025.

[37] NVIDIA Corporation. Nvidia vmaterials library. https://developer.nvidia.com/vmaterials, 2024. Accessed: 2024-06-12.

[38] Ouster, Inc. Os1 lidar sensor - specifications. https://ouster.com/products/hardware/os1-lidar-sensor, 2024. Accessed: 2024-06-08.

[39] V. Petitjean. FIR PSGEN-15323: Reflectivity for Lidar. Technical report, Siemens Prescan NV, 2019. Last Revision Date: 23 October 2019.

[40] M. Pharr, W. Jakob, and G. Humphreys. *Physically based rendering: From theory to implementation*. MIT Press, 2023.

[41] N. Ravi, V. Gabeur, Y.-T. Hu, R. Hu, C. Ryali, T. Ma, H. Khedr, R. Rädle, C. Rolland, L. Gustafson, E. Mintun, J. Pan, K. V. Alwala, N. Carion, C.-Y. Wu, R. Girshick, P. Dollár, and C. Feichtenhofer. Sam 2: Segment anything in images and videos, 2024.

[42] F. Remondino, A. Karami, Z. Yan, G. Mazzacca, S. Rigon, and R. Qin. A critical analysis of nerf-based 3d reconstruction. *Remote Sensing*, 15(14):3585, 2023.

[43] T. Ren, S. Liu, A. Zeng, J. Lin, K. Li, H. Cao, J. Chen, X. Huang, Y. Chen, F. Yan, Z. Zeng, H. Zhang, F. Li, J. Yang, H. Li, Q. Jiang, and L. Zhang. Grounded sam: Assembling open-world models for diverse visual tasks, 2024.

[44] X. Ren, Y. Lu, H. Liang, Z. Wu, H. Ling, M. Chen, S. Fidler, F. Williams, and J. Huang. Scube: Instant large-scale scene reconstruction using voxsplats, 2024.

[45] J. L. Schönberger and J.-M. Frahm. Structure-from-motion revisited. In *IEEE Int. Conf. Computer Vision and Pattern Recognition*, pages 4104–4113, 2016.

[46] X. Shi, L. Chen, P. Wei, X. Wu, T. Jiang, Y. Luo, and L. Xie. Dhgs: Decoupled hybrid gaussian splatting for driving scene, 2024.

[47] Siemens Digital Industries Software. *Simcenter Prescan Manual, Version 2411*, 2024.

[48] C. Sun, J. Choe, C. Loop, W.-C. Ma, and Y.-C. F. Wang. Sparse voxels rasterization: Real-time high-fidelity radiance field rendering. *ArXiv*, abs/2412.04459, 2024.

[49] M. Tancik, V. Casser, X. Yan, S. Pradhan, B. Mildenhall, P. P. Srinivasan, J. T. Barron, and H. Kretzschmar. Block-nerf: Scalable large scene neural view synthesis, 2022.

[50] H. Turki, D. Ramanan, and M. Satyanarayanan. Mega-nerf: Scalable construction of large-scale nerfs for virtual fly-throughs, 2022.

[51] P. Upchurch and R. Niu. A dense material segmentation dataset for indoor and outdoor scene parsing, 2022.

[52] G. Vecchio and V. Deschaintre. Matsynth: A modern pbr materials dataset. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, page 22109–22118. IEEE, June 2024.

[53] K. Viswanath, P. Jiang, and S. Saripalli. Reflectivity is all you need!: Advancing lidar semantic segmentation, 2024.

[54] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.

[55] M. Wei, Q. Wu, J. Zheng, H. Rezatofighi, and J. Cai. Normal-gs: 3d gaussian splatting with normal-involved rendering, 2024.

[56] M. Wojciech, P. Hanspeter, B. Matt, and M. Leonard. A data-driven reflectance model. *ACM Transactions on Graphics*, 22(3):759–69, 2003.

[57] Q. Wu, J. Martinez Esturo, A. Mirzaei, N. Moenne-Loccoz, and Z. Gojcic. 3dgut: Enabling distorted cameras and secondary rays in gaussian splatting. *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

[58] T. Xie, Z. Zong, Y. Qiu, X. Li, Y. Feng, Y. Yang, and C. Jiang. Physgaussian: Physics-integrated 3d

gaussians for generative dynamics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4389–4398, June 2024.

[59] B. Xiong, J. Liu, J. Hu, C. Wu, J. Wu, X. Liu, C. Zhao, E. Ding, and Z. Lian. Texgaussian: Generating high-quality pbr material via octree-based 3d gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 551–561, June 2025.

[60] Y. Yao, Z. Zeng, C. Gu, X. Zhu, and L. Zhang. Reflective gaussian splatting, 2025.

[61] J. Ye, L. Zhu, R. Zhang, Z. Hu, Y. Yin, L. Li, L. Yu, and Q. Liao. Large material gaussian model for relightable 3d generation, 2025.

[62] J. E. Zaalberg. Master thesis: Reducing the sim-to-real gap: Lidar-based 3d static environment reconstruction. Master's thesis, TU Delft - Mechanical Engineering, 2024.

[63] B. Zhang, C. Fang, R. Shrestha, Y. Liang, X. Long, and P. Tan. Rade-gs: Rasterizing depth in gaussian splatting, 2024.

[64] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang. The unreasonable effectiveness of deep features as a perceptual metric, 2018.

[65] Z. Zhang. Introduction to machine learning: k-nearest neighbors. *Annals of translational medicine*, 4(11):218, 2016.

[66] Z. Zhang, B. Huang, H. Jiang, L. Zhou, X. Xiang, and S. Shen. Quadratic gaussian splatting for efficient and detailed surface reconstruction, 2024.

[67] X. Zhao, W. Ding, Y. An, Y. Du, T. Yu, M. Li, M. Tang, and J. Wang. Fast segment anything, 2023.

[68] Y. Zhou, L. Liu, H. Zhao, M. López-Benítez, L. Yu, and Y. Yue. Towards deep radar perception for autonomous driving: Datasets, methods, and challenges. *Sensors*, 22(11):4208, 2022.

[69] J. Zhu, H. Deng, and H. Chen. Can large language models identify materials from radar signals? *arXiv preprint arXiv:2508.03120*, 2025.

[70] M. Zwicker, H. Pfister, J. Van Baar, and M. Gross. Ewa volume splatting. In *Proceedings Visualization, 2001. VIS'01.*, pages 29–538. IEEE, 2001.

APPENDIX A
BECAREFUL VEHICLE SETUP

This section provides a comprehensive description of our Becareful vehicle setup. We visualize the sensor configuration in Fig. 7. The test vehicle is equipped with a 360°LiDAR sensor (Ouster OS1-128), six cameras with a resolution of 2880×1880 (width × height), five radar units, and an inertial measurement unit. To ensure that ego-vehicle parts (such as the hood and rear) do not appear in our images, we post-process the data to mask these areas.

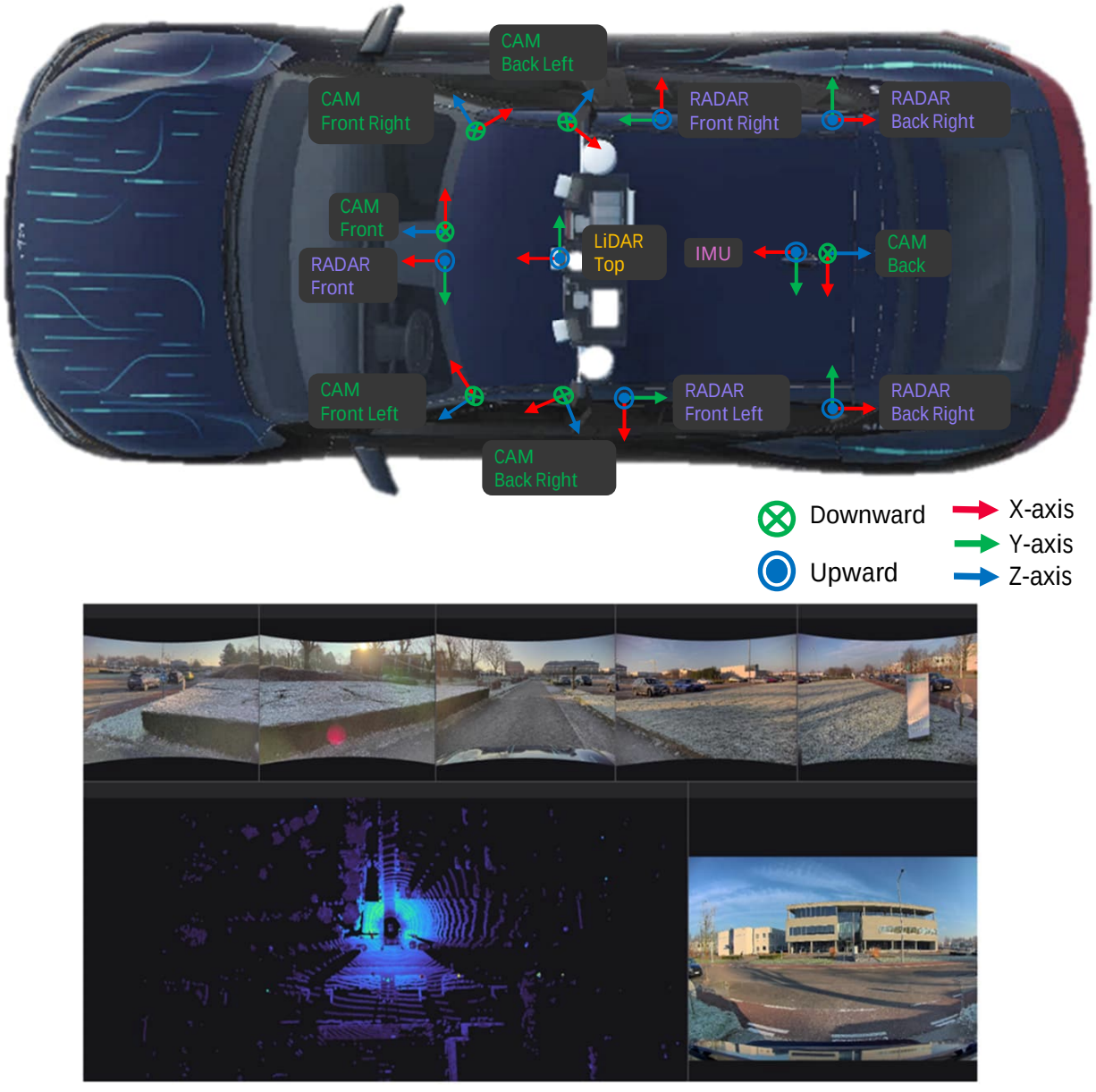


Fig. 7. Sensor configuration KIA EV6 Becareful dataset[1]

APPENDIX B
IMAGE SIMILARITY - METRICS

This section presents qualitative results for novel 3D reconstruction frameworks, highlighting the use of Gaussian Splatting and other methodologies to enhance the visual fidelity of 3D representations within real-to-sim pipelines.

A. Metrics

Standardised metrics are commonly used to quantitatively evaluate the accuracy of neural renderings generated by 3D reconstruction models compared to real-world scenes. These benchmarks assess pixel-level accuracy as well as perceptual similarity, allowing fair comparisons across different datasets and methods. Below, we detail the most widely used metrics.

Peak Signal-to-Noise Ratio (PSNR) [13] is a widely used evaluation metric that compares an original image with its reconstructed version to quantify the noise introduced during processing. Noise is measured by its impact on visual quality: a higher PSNR value indicates lower noise or distortion, while a lower value suggests significant distortion or a high level of noise. The equation for PSNR is shown below.

$$\text{PSNR} = 10 \cdot \log_{10} \left(\frac{MAX_I^2}{\text{MSE}} \right) \tag{1}$$

Mean Square Error (MSE) and its variant, Root Mean Square Error (RMSE), measure the average squared difference between predicted values and the ground truth. These metrics provide a straightforward way to quantify reconstruction errors. The equation is given below.

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \tag{2}$$

This method is often considered a standard approximation for accuracy. Although a higher PSNR generally correlates with higher quality reconstruction, this is not always the case. When estimating image and video quality as perceived by humans, PSNR has been shown to perform poorly compared to other quality metrics.

Structural Similarity Index Measure (SSIM) [54] This perceptual metric evaluates the similarity between two images by examining how their local structures compare. It leverages three components: luminance, contrast, and structure. The metric ranges from 0 to 1, where 1 indicates perfect similarity and 0 means no similarity. The main equation is provided below.

$$\text{SSIM}(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \tag{3}$$

The full equation can be split per individual components

$$\text{Luminance:} \quad l(x, y) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1} \tag{4}$$

$$\text{Contrast:} \quad c(x, y) = \frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2} \tag{5}$$

$$\text{Structure:} \quad s(x, y) = \frac{\sigma_{xy} + C_3}{\sigma_x\sigma_y + C_3} \tag{6}$$

SSIM provides an analytical approach to image quality assessment by mathematically modeling aspects of human visual perception. It quantifies similarity using luminance, contrast, and structure components, focusing on perceived quality rather than raw pixel differences. Although SSIM closely approximates human judgment, it may not perfectly reflect subjective preferences—for example, humans might assign different importance to certain image features than the metric does.

Learned Perceptual Image Patch Similarity (LPIPS) [9, 64] is a perceptual metric that compares images based on high-level features like texture, shape, and structure, using deep representations extracted from pretrained convolutional neural networks (CNNs). Unlike fixed-weight metrics, LPIPS adapts the importance of different feature layers, with learned weights that echo how humans naturally emphasize certain visual aspects over others. As a result, LPIPS typically provides the most accurate representation of human perception of image similarity among commonly used analytical metrics, though it remains an approximation of subjective human judgment.

APPENDIX C
REFLECTIVITY CALIBRATION

This section provides a comprehensive explanation of our calibration process. Simcenter Prescan's physics-based LiDAR outputs a unit called 'Power' [47], which represents intensity in a non-vendor-specific way. However, our research requires vendor-specified intensity values. Due to proprietary restrictions on manufacturer details, an internal investigation [19, 39] has established a methodology to convert Simcenter Prescan power units to vendor-specified 'reflectivity', which is independent of range or incidence angle. The following chapters will elaborate on the underlying theory and the steps involved in this conversion process.

A. Physics-based lidar

The physics-based lidar in Simcenter Prescan can generate synthetic data closely matching that of a real-world lidar, including material physics such as intensity. Intensity represents the amount of energy returned when incident light reflects off a surface. Several material properties influence the intensity value in the physics-based lidar, including the red channel of the base colour, specularity, roughness, clearcoat (associated with vehicle paint), metallic content, and opacity. The received power depends on these material properties, as well as on the range and incidence angle. Figure 8 illustrates how the direction of incident light determines its origin.

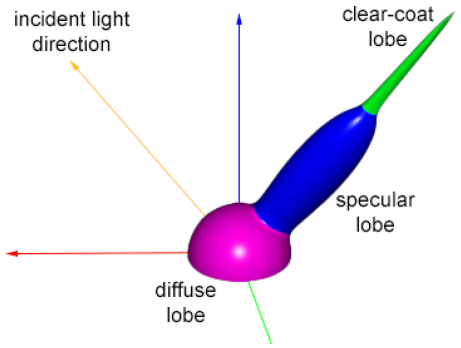


Fig. 8. In the diagram [47], the incident light direction indicates where the light originates. The magenta, blue, and green lobes illustrate how the energy is distributed across different directions. For LiDAR applications, the retroreflective direction—where the light source and view direction coincide—is of particular interest. Additionally, a red base color will significantly increase the power received at the sensor

B. Calibration

LiDARs often measure the reflectivity of objects independently of the laser power or distance involved. For example, our Ouster OS-1 sensor captures both reflectivity and intensity. Its reflectivity values are determined through laser calibration against National Institute of Standards and Technology (NIST)-calibrated reflectivity reference targets at the factory [19]. The reported reflectivity is typically an 8-bit value ranging from 0 to 255. For calibrating our LiDAR's reflectivity, we follow this method to compute the value based on the power output from the physics-based LiDAR sensor (PBL).
We perform a calibration experiment with two planes: one plane is set to have a 'perfect diffuse' material and the other a 'perfect specular' material property. The PBL sensor is positioned at a distance of 5 meters from the two targets. This calibration setup is illustrated in Fig. 9.

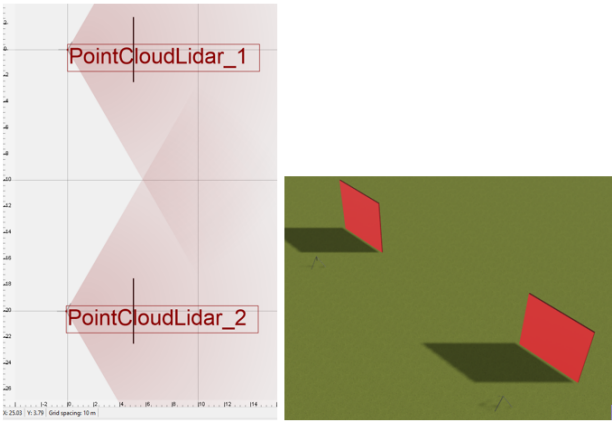


Fig. 9. Depiction of the placement of lidars 5 meters away from the targets with diffuse and specular [19].

The purpose of this calibration process is to distinguish between different surface types using their reflectivity values. Reflectivity can be classified in zones: values from 0 to 100 generally correspond to diffuse surfaces, while values from 101 to 255 indicate specular or retro-reflective surfaces. The calibration results from our experiment in Simcenter Prescan are shown in Fig. 10. In this setup, we use the center beam, as its incidence is perpendicular to the target, which yields the highest reflected power.

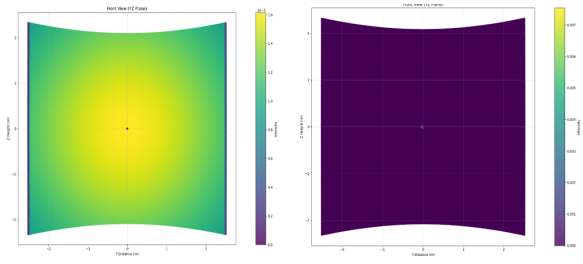


Fig. 10. Power returned from the target with a diffuse texture (left) and the target with a specular texture (right).

TABLE IV
CALIBRATION PARAMETERS FOR DIFFUSE AND SPECULAR TARGETS BASED ON OUSTER-OS1 128 PROPERTIES

Parameter	Value
$P_{d,\text{calibrated}}$	0.00002
$P_{s,\text{calibrated}}$	0.00077
$r_{\text{calibrated}}$	4.99997
$n_{d,\text{calibrated}}$	0.00041
$n_{s,\text{calibrated}}$	0.01933

Utilizing the given values, we seek a way to normalize the power returned by each LiDAR beam to convert power into calibrated reflectivity, making the result independent of laser power and distance to the target. To achieve this, the power received by the sensor in the calibration experiment, $P_{\text{calibrated}}$, is multiplied by the square of the distance to the calibration target, $r_{\text{calibrated}}$, which removes the effect of distance from the reflectivity computation. This results in a calibrated reflectivity, $n_{\text{calibrated}}$.

$$n_{\text{calibrated}} = P_{\text{calibrated}} \cdot {r_{\text{calibrated}}}^2$$

We perform this calculation for both the diffuse and specular targets, yielding $n_{\text{d,calibrated}}$ and $n_{s,\text{ calibrated}}$. Using these reference values, we can compute the normalised reflectivity for any arbitrary beam by first calculating the measured reflectivity as:

$$n_{\text{measured}} = P_{\text{measured}} \cdot {r_{\text{measured}}}^2$$

The conversion to normalised reflectivity is then performed using the following criteria:

$$\text{refl}_{\text{ normalised}} = 100 \times \frac{\text{n}_{\text{measured}}}{\text{n}_{\text{d, calibrated}}}$$
$$\text{if } \text{n}_{\text{measured}} \leq n_{d,\text{ calibrated}}$$

or:

$$\text{refl}_{\text{ normalised}} = 100 + 155 \times \frac{\text{n}_{\text{measured}} - \text{n}_{\text{d, calibrated}}}{\text{n}_{\text{s, calibrated}} - \text{n}_{\text{d, calibrated}}}$$
$$\text{if } n_{d,\text{ calibrated}} \leq \text{n}_{\text{measured}} \leq n_{s,\text{ calibrated}}$$

or:

$$\text{refl}_{\text{normalised}} = 255$$
$$\text{if } n_{\text{measured}} > n_{s,\text{ calibrated}}$$

We proceed by applying the equation to determine our calibrated values. The calibration parameters are listed in Tab. IV. This process yields a piecewise linear function that maps the returned power to normalized reflectivity for any beam. By making the reflectivity value independent of the range and laser power, we provide a more accurate representation of the material properties through the reflectivity measurement.

C. Limitations

As indicated in the reports, the calibration procedures were specifically developed for Velodyne lidars. However, we have found that this methodology provides acceptable accuracy for our Ouster-OS1 lidar, which we use to capture our data. It should be noted that an approximation error may be introduced when converting power to reflectivity using this approach. Additionally, due to limited information disclosed by the manufacturer, certain parameters such as beam power and sensor area had to be assumed based on standard values. As a result, the converted reflectivity values might not be fully accurate compared to real-world measurements.

APPENDIX D
QUALITATIVE IMAGE SIMILARITY EVALUATION

This section provides the complete qualitative evaluation described in Sec.IV-E. The full comparison with other novel benchmark models is shown in Fig.11. Our adapted H3DGS model produces high-quality visuals that closely match the ground truth. We observed that nearby objects, such as vehicles, often present challenges for many benchmark models; this may be because methods like 3DGS generally perform poorly when dealing with sparse or non-object-centric views, due to an insufficient number of viewpoints. Other models, such as Citygs, show improvement but still suffer from image blur in their renderings, which could be due to the fact that Citygs is primarily designed for aerial or sky-captured data. However, despite the strong performance demonstrated by our model, some challenges remain, specifically, image blurring in the sky, difficulties with highly textured objects such as vegetation, and issues with retro-reflective surfaces like windows or traffic signs.

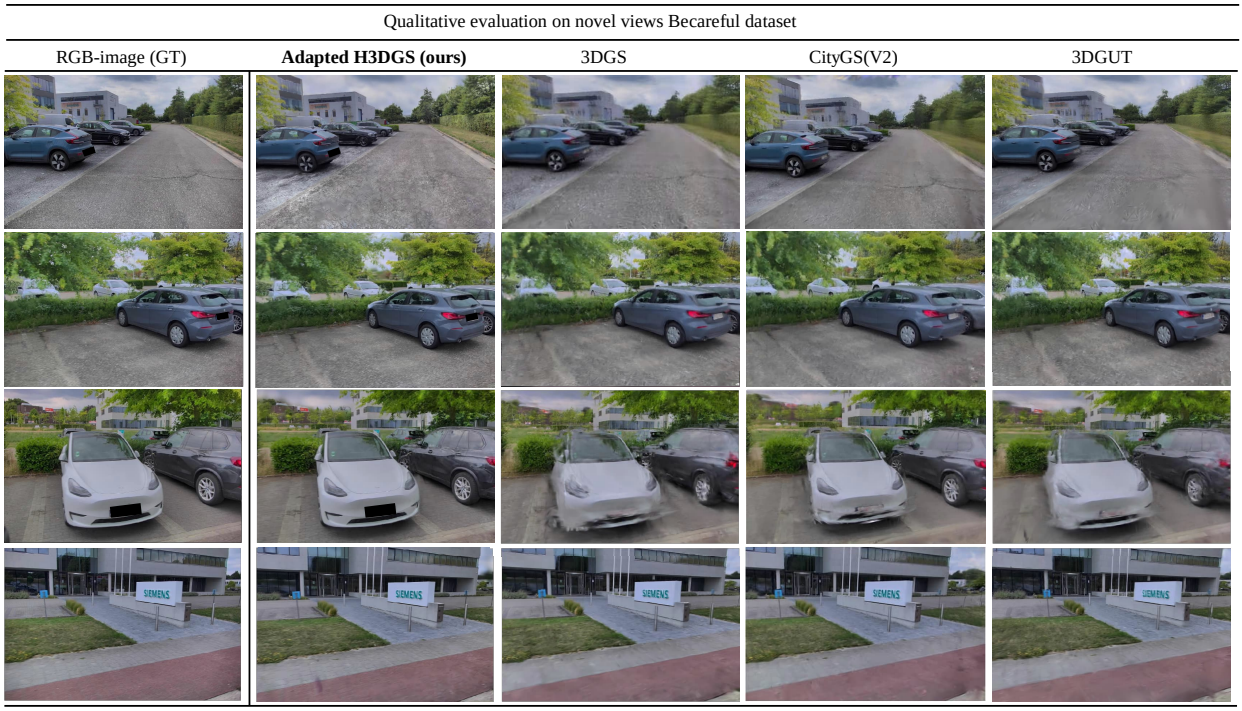


Fig. 11. **Qualitative validation on rendering performance of novel views:** We compare our adapted H3DGS model by a qualitative evaluation with other Gaussian Splatting models and can find that the adapted H3DGS model can be still considered to have the most feasible visual quality despite the Image similarity scores.

APPENDIX E
QUALITATIVE REFLECTIVITY EVALUATION

This section provides the complete qualitative evaluation described in Sec.IV-F. We examine our synthetic lidar data produced by the Physics-based Lidar (PBL), as illustrated in Fig.12. Our initial observation is that the camera-GS mesh exhibits geometrical inconsistencies when compared to the lidar-based mesh reconstruction. We therefore conclude that, with regard to camera-based mesh reconstruction, the geometry remains inadequate; the resulting mesh is still quite rough and contains many floating artifacts caused by Gaussian Splatting, which also tends to include elements like the sky in its reconstruction. In terms of reflectivity, both the lidar-based and camera-GS-based reconstructions show significantly higher reflectivity for vegetation and lower reflectivity for buildings. This pattern can be attributed to intra-class variance as well as the composition of rendering properties, which may not fully align with laboratory-measured material values. Nevertheless, we also observe that both models yield similar reflectivity values for materials such as asphalt. Based on these findings, we conclude that, regardless of geometrical accuracy, both models display general agreement with the ground truth in terms of reflectivity. While some material classes, such as vegetation or concrete, exhibit noticeable differences in reflectivity, this can be explained by intra-class variance (e.g., building concrete often has a lighter color than standard concrete) and the use of PBR-composed materials, as highlighted previously.

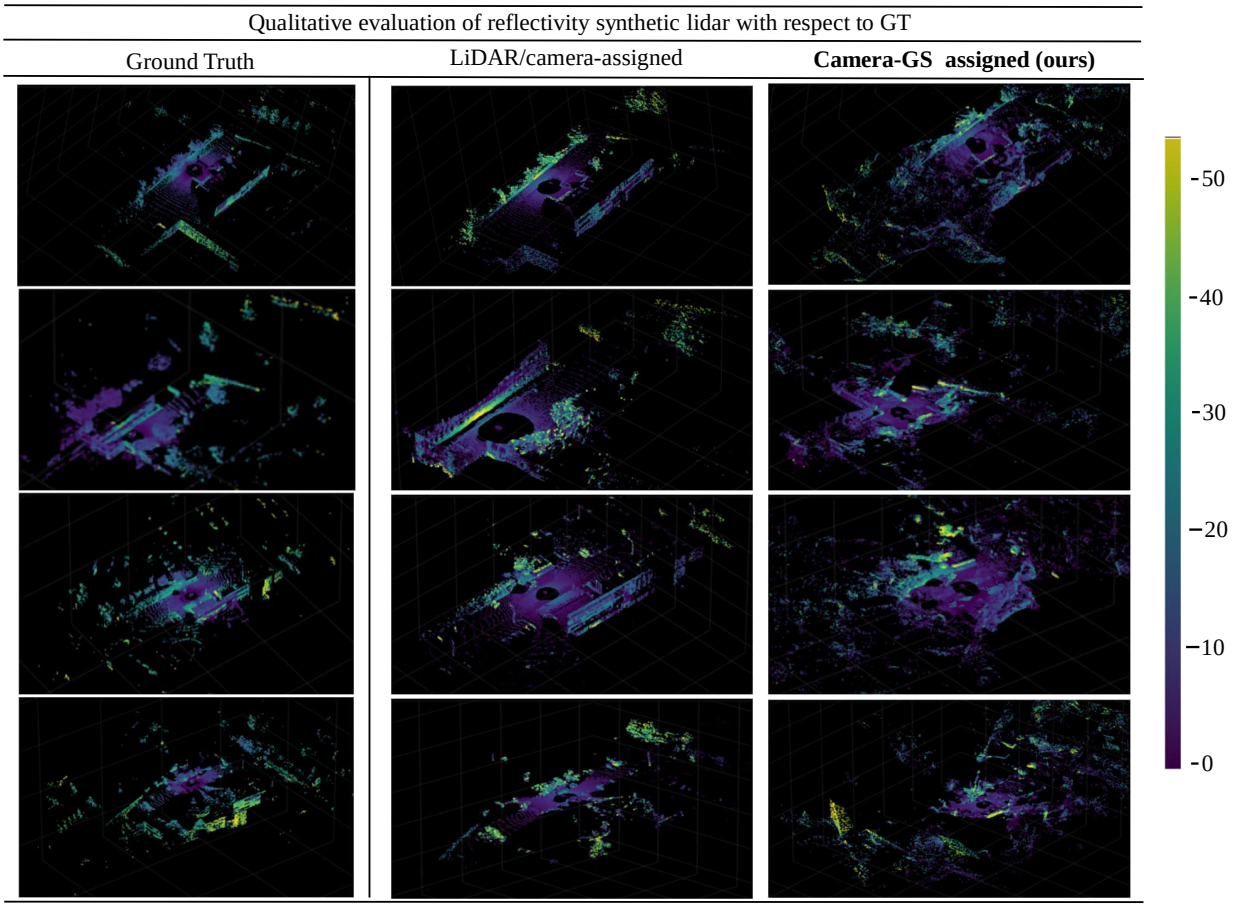


Fig. 12. **Qualitative validation on rendering performance of novel views:** We compare our adapted H3DGS model by a qualitative evaluation with other Gaussian Splatting models and can find that the adapted H3DGS model can be still considered to have the most feasible visual quality despite the Image similarity scores.

APPENDIX F
RMSNET EVALUATION

This section provides a detailed evaluation of integrating SAM to address RMSnet's [6] limitations in capturing shape cues. The goal of this study is to assess the overall improvement in per-pixel accuracy achieved by leveraging SAM. Here, we evaluate the performance of RMSnet and examine the added benefits of using SAM for semantic shape cue assistance.

A. RMSnet evaluation on MCubes

RMSnet was originally trained on the KITTI dataset, which provides 1,000 hand-annotated semantic material masks. Of these, 800 were used for training, with 400 categorised as urban area. Because the ground truth data was manually annotated, semantic material masks are limited in availability, restricting our ability to freely validate the model. Consequently, using the same KITTI dataset for both training and validation would not be feasible.

To address this, we leveraged MCubes, a dataset proposed by Liang et al. [31], which offers 500 material-annotated samples for urban scenes, as our validation set. Our results, shown in Fig. 13, present the attention matrices for both the original KITTI-urban split and the MCubes validation results. We first observed that RMSnet performed exceptionally well on the KITTI-urban dataset, achieving near-perfect prediction accuracy. However, a slight error persists between the 'concrete' and 'plaster' material classes due to their similar textures. When evaluated on the MCubes dataset, RMSnet's performance changed significantly. While some materials remained as dominant as in the KITTI dataset, we found that RMSnet is highly sensitive to changes in the dataset, likely because its training set is relatively small. This case study demonstrates that RMSnet does not perform as consistently on new datasets.

B. SAM evaluation

Our qualitative evaluation of RMSnet, illustrated in Fig.3, demonstrates that adding shape cues from SAM helps define object boundaries for semantic materials. To reinforce these qualitative findings, we conducted a quantitative analysis, comparing the effect of various SAM2 weights and FastSAM, as shown in Fig.14. We observed a correlation between the number of masks and overall per-pixel accuracy: more masks allow for better coverage of fine details (such as windows and road markings), but this comes at the cost of increased computational overhead. However, we also found that too many masks can fragment semantic object masks, which sometimes results in lower performance.

Among the models tested, SAM2 weight Large 2.1 achieved the best results, with an accuracy of 0.67 and IoU of 0.31. Nevertheless, its computational time was at least three times longer than that of FastSAM, which, while slightly less accurate (0.65), is much faster (12.77s). Consequently, we chose to accept a modest loss in performance for significantly reduced computation time, making the system more scalable.

C. Conclusion

This evaluation presents a qualitative overview of adding semantic object masks to RMSnet to assess SAM's contribution to overall per-pixel accuracy. We used the MCubes dataset, which contains 500 material-annotated masks, and found that RMSnet did not perform as well on this dataset as it did on its original KITTI-urban split. To examine the relationship between the number of masks or model size and improvements in per-pixel accuracy for material segmentation, we compared several SAM2 weights and FastSAM. After weighing performance against computational time, we selected FastSAM, accepting a slight decrease in accuracy for significantly faster processing. With this approach, we raised per-pixel accuracy from 0.63 to 0.65.

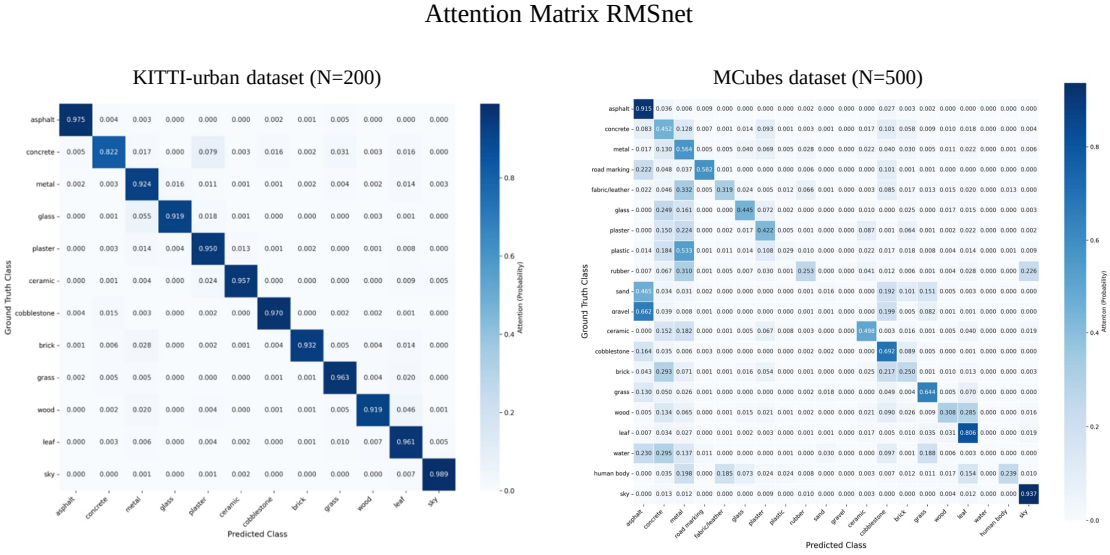
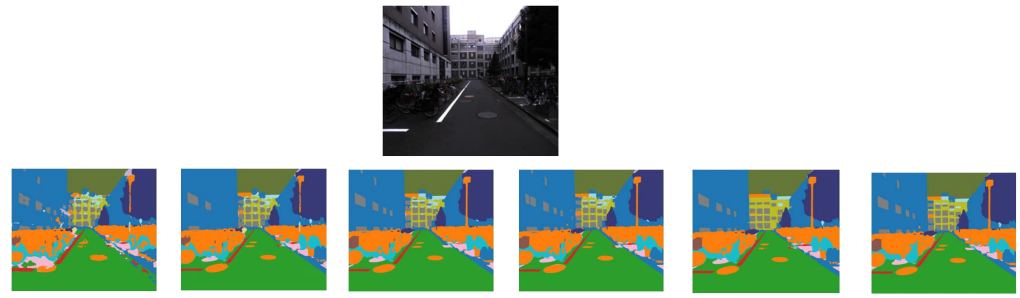


Fig. 13. **Left:** Our attention matrix, based on the validation dataset from the KITTI-urban split (containing 200 samples), demonstrates nearly perfect scores with minimal deviation. **Right:** We perform the same validation using 500 samples from the MCubes dataset and observe that the prediction accuracy drops significantly compared to the KITTI-split dataset.



Model		**RMSnet (vanilla)**	**Base+ (2.1)**	**Base+ (2.1)**	**Base+ (2.1)**	**FastSAM**	**Large (2.1)**
Samples (masks)		0	140	280	591	156	306
Time per image (s)		NaN	13.71s	35.86s	**121.49s**	**12.77s**	42.23s
Statistics	Accuracy*	**0.6345**	0.6487	0.6517	-	0.6510	**0.6719**
	IoU*	0.2817	0.2882	0.2924	-	0.2931	**0.3135**

Fig. 14. We evaluated the standard RMSnet output with additional semantic object masks from FastSAM and tested various weights from SAM2. While SAM2 Large 2.1 achieved the highest scores, it also required significantly more computational resources. Therefore, we found that FastSAM offers the best balance between computational efficiency and per-pixel accuracy.