

UNIVERSITÀ
DI TRENTO
Department of
Information Engineering and Computer Science



FONDAZIONE
BRUNO KESSLER

Doctoral Programme in
Information Engineering and Computer Science

3D reconstruction and scene understanding with 3D Gaussian Splatting representation

Ziyang Yan
PhD Cycle: 38th - Academic Years: 2022-2025

Advisor
Prof. Fabio Remondino
Fondazione Bruno Kessler

May 2026

Abstract

The rapid development of 3D computer vision has made the precise reconstruction and understanding of complex environments essential. At the heart of this challenge lies the need for efficient, expressive 3D representations capable of capturing geometry, appearance, and semantics. While recent neural scene representations have notably improved image-based reconstruction, they are often hindered by high computational costs, limited robustness, and poor scalability for large-scale or interactive applications.

The primary objective of this thesis is to improve the performance of neural scene representations, with a particular focus on 3D Gaussian Splatting, to support both precise 3D reconstruction and comprehensive scene understanding across diverse real-world scenarios. To achieve this goal, we start by introducing NeRFBK, a comprehensive benchmark dataset designed to systematically evaluate radiance field representations against traditional photogrammetry. Next, we investigate the reconstruction of non-collaborative surfaces, such as reflective and transparent objects, which remain challenging for existing 3D reconstruction approaches. By incorporating surface normal supervision and relighting guidance into the 3D Gaussian Splatting framework, we develop a reconstruction pipeline that significantly improves geometric fidelity under sparse views and complex illumination conditions. Furthermore, we extend 3D Gaussian representations to large-scale outdoor scene understanding by introducing RenderWorld, a unified framework that leverages 3D Gaussian Splatting to generate self-supervised 3D occupancy representations from multi-view images, enabling efficient scene reconstruction, semantic reasoning, and motion forecasting in autonomous driving scenarios. Finally, we explore the potential of 3D Gaussian representations for interactive scene manipulation. We propose 3DSceneEditor, a fully 3D-based framework that enables controllable and semantically aware editing of complex indoor scenes by directly operating on Gaussian primitives.

Overall, this thesis provides a series of advancements in Gaussian-based scene representations for 3D reconstruction, scene understanding, and interactive 3D applications. These contributions not only improve the fidelity and efficiency of 3D reconstruction but also advance the capability of machines to interpret and interact with complex 3D environments, bringing machine perception closer to human-level spatial understanding.

Keywords

[3D Reconstruction, 3D Gaussian Splatting, 3D Scene Understanding, Qualitative analyses, Multi-view Synthesis]

Acknowledgments

As I reflect on my PhD journey, it is the remarkable individuals and communities I encountered along the way that stand out most—those with whom I have shared unforgettable memories.

Supervisor. I would like to express my deepest gratitude to my PhD supervisor, Prof. Fabio Remondino, for giving me the invaluable opportunity to pursue my PhD within one of the world's leading research groups. Fabio, your trust and the autonomy you granted me encouraged me to explore research problems independently and creatively. Your exceptional ability to identify impactful research questions, combined with your insightful and constructive feedback, has continuously guided our work in the right direction and ensured the highest research quality.

Colleagues, Friends, and Family. Looking back on these three years, research is only part of the journey. I would not have come this far without the support of incredible colleagues, dear friends, and my loving family.

My heartfelt thanks go to my peers and past members at 3DOM—Elisa, Salim, Raniero, Paweł, Luca, Gabriele, Oscar, Simone, Narges, Nazanin, Vandita, Ashkan, Matteo, Lauris, Mustafa, Giulio, Fabio, Roberto, Michele, Ali, and Ellie. The countless brainstorming sessions, inspiring discussions, coffee breaks, hikes in the mountains of Trentino, and shared meals are memories I will always cherish. For me, 3DOM has been more than a research group—it has been a home where I formed truly special bonds.

I am also deeply grateful to the members of UCF CRCV. Prof. Mubarak Shah, thank you for giving me the opportunity to visit your group. Under your guidance, I gained valuable insights into integrating video understanding with 3D representations, advancing toward 4D modeling. You also introduced me to a new research culture and gave me the chance to experience unforgettable moments in Orlando. My sincere thanks to Dr. Wang, Dr. Shang, Dr. Tian, Raiyaan, Jeffrey, Andong, Soumik, Parth, Kunyang, Ming, Xin, Qingyang, Zonglin, Sirshapan, Rajat, Sabbir, Ahnaf, Qucheng, Sarinda, David, Nyle, Swetha, Guangyu, Adeel, Kevin, and Yancheng—I truly enjoyed both the academic collaborations and the social activities we shared.

Moreover, I would like to thank the wonderful colleagues at CWM BAAI. Dr. Zhao, thank you for giving me the opportunity to work with you and to explore how 3D research can contribute to embodied intelligence. I am also grateful to all group members—Hong, Yuan, Ang, Shaocong, Gen, Chongjie, Nan, Kun, Boyuan, Yuhao,

Weiqing, Houyuan, Zehao, and Zheng—for your constant support and willingness to help whenever I encountered challenges.

Finally, to my parents, your unconditional love and unwavering belief in me have always been my greatest source of strength. Your kindness, patience, and support have shaped who I am today. And to my grandmother, watching from above—I hope I have made you proud.

Collaborators. I would like to express my sincere gratitude to Dr. Yan Wang, Dr. Zhe Wang, and Mr. Yuhang Lu for their invaluable guidance and continuous support on my autonomous driving project, which represents an important part of my PhD research. Dr. Wang, your patience and dedication in guiding me were truly exceptional—every discussion with you was a valuable learning experience. It has been a privilege to have you not only as a collaborator, but also as a trusted friend.

I am also deeply thankful to Dr. Lei Li for your sincere advice and guidance, especially during the early stages of my research in scene understanding. Your support, both academically and personally, has meant a great deal to me.

Furthermore, I would like to express my sincere appreciation to Dr. Tang, Dr. Guo, Ziren, Guo, Lingyu, Yongjie, Feng, Yinghao, Xingyu, Yeling, Jiachen, Mengrui, and Junzhou. Thank you for your trust in me and your willingness to build meaningful and in-depth academic collaborations. Your dedication and hard work have not only contributed significantly to our research, but have also made my academic journey richer and more fulfilling.

Special thanks go to Yihua Shao, my closest and longest-standing collaborator and a treasured friend. Whenever I faced challenges, you consistently provided thoughtful insights and multiple solutions. Over the past two years, we have collaborated on more than ten papers, working side by side through both challenges and successes. Our collaboration has greatly enriched my academic journey and laid a solid foundation for my future in academia.

Finally, I would like to thank Minwen Liao. It has been a pleasure to work with such a talented and dedicated student. I truly appreciate your openness and support throughout our collaboration.

Contents

Abstract i

Acknowledgments iii

1 Introduction 1
- 1.1 Motivation . . . 1
- 1.2 Research Questions and Challenges . . . 3
- 1.3 Contributions . . . 4
  - 1.3.1 Comprehensive Dataset and Evaluation for Radiance Field Representations . . . 5
  - 1.3.2 Improving Reconstruction Quality with Normal Supervision and Relighting . . . 6
  - 1.3.3 Extending 3D Gaussian Splatting to Autonomous Driving Scene Reconstruction and Understanding . . . 7
  - 1.3.4 3D-Only Indoor Scene Understanding and Editing with 3D Gaussian Representation . . . 8
- 1.4 Outline . . . 9
- 1.5 Publications . . . 9

2 Background, State of the Art and Related Works 13
- 2.1 Development of Neural Field Representations . . . 13
  - 2.1.1 Neural Radiance Fields (NeRF) . . . 13
  - 2.1.2 3D Gaussian Splatting (3DGS) . . . 17
- 2.2 Scenarios for GS-based 3D Reconstruction . . . 19
- 2.3 3D Scene Understanding and Editing with 3D Gaussian Splatting . . . 21
  - 2.3.1 3D Scene Understanding . . . 21
  - 2.3.2 3D Scene Editing . . . 22
- 2.4 Method-Oriented Related Works . . . 23
  - 2.4.1 3D Gaussian Splatting for Specular Materials . . . 23
  - 2.4.2 Normal Estimation and Geometric Supervision . . . 23
  - 2.4.3 Relighting and Multi-illumination Synthesis . . . 24
  - 2.4.4 Novel View Synthesis with Diffusion . . . 24
  - 2.4.5 3D Occupancy Prediction . . . 25
  - 2.4.6 World Model in Autonomous Driving . . . 25
  - 2.4.7 3D Representations for Complex Scenes . . . 25

2.4.8 3D Scene Editing 26

3 NeRFBK Dataset Construction 27
3.1 Introduction 27
3.2 The Construction of NeRFBK 28
3.3 Comparison of 3D Reconstruction Datasets 36
3.4 Summary and Relation to Research Question 36

4 Evaluation of Learning-based Methods 39
4.1 Introduction 39
4.2 Analysis and Evaluation of Non-collaborative Surfaces 41
4.2.1 Proposed Assessment Methodology 41
4.2.2 Testing Objects and Methods 43
4.2.3 Results from Multi-View Image Sequences 43
4.2.4 Results from Monocular Depth Estimation 50
4.2.5 Discussion 54
4.2.6 Conclusions of Non-collaborative Surfaces Evaluation 57
4.3 Analysis and Evaluation on Urban Scenes 58
4.3.1 Proposed Assessment Methodology 58
4.3.2 Implementation Details and Data Pre-processing 58
4.3.3 Datasets 61
4.3.4 Metrics 61
4.3.5 Experiment Results 63
4.3.6 Conclusions of Urban Scene Evaluation 69
4.4 Summary and Relation to Research Question 70

5 Improving Surface Reconstruction of Non-Collaborative Objects using Gaussian Splatting 71
5.1 Introduction 71
5.2 Methodology 73
5.2.1 Overall Framework 73
5.2.2 Multi-illumination Data Generation 75
5.2.3 Normal Estimation and Supervision 76
5.2.4 3D Reconstruction with Gaussian Splatting 77
5.3 Experiments 78
5.3.1 Implementation Details 78
5.3.2 Qualitative Experiment 81
5.3.3 Quantitative Evaluation 84
5.3.4 Ablation Study 87
5.4 Conclusions 88
5.5 Limitations and Future Works 89
5.5.1 Summary and Relation to Research Question 89

6 Reconstruct and Understand Autonomous Driving Scenes with 3D Gaussian Splatting 91
6.1 Introduction . . . 91
6.2 Methodology . . . 92
6.2.1 3D Occupancy Prediction with Multi-frame 2D Labels . . . 92
6.2.2 Air Mask Variational Autoencoder (AM-VAE) . . . 96
6.2.3 World Model . . . 97
6.3 Experiments . . . 99
6.3.1 Experimental Setup . . . 99
6.3.2 Main Result . . . 99
6.3.3 Ablation Study . . . 103
6.4 Conclusions . . . 104
6.5 Limitations and Future Works . . . 105
6.6 Summary and Relation to Research Question . . . 105

7 Indoor Scene Understanding and Editing with 3D Gaussian Splatting 107
7.1 Introduction . . . 107
7.2 Methodology . . . 109
7.2.1 Overall Framework . . . 109
7.2.2 Open-Vocabulary Object Grounding . . . 111
7.2.3 3D Gaussian Editing . . . 112
7.3 Experiments . . . 114
7.3.1 Implementation Details . . . 114
7.3.2 Qualitative Evaluation . . . 115
7.3.3 Quantitative Evaluation . . . 117
7.3.4 Ablation Study and Analysis . . . 118
7.4 Conclusions . . . 120
7.5 Limitations and Future Works . . . 121
7.6 Summary and Relation to Research Question . . . 121

8 Conclusions 123
8.1 Core Contributions . . . 123
8.2 Future Works . . . 125
8.2.1 3DGS Future in 3D . . . 126
8.2.2 3DGS future in 4D . . . 127
8.2.3 3DGS Future in Metrology and Geomatics . . . 129

Bibliography 131

A Experimental Details, Additional Visualizations, and User Study of 3DSceneEditor 175
A.1 Implementation details . . . 175
A.1.1 Implementation Details of 3D Scene Representation . . . 175
A.1.2 Implementation Details of Object Grounding . . . 176
A.1.3 Implementation Details of 3D Gaussians Editing . . . 176

List of Tables

3.1 Transparent and Reflective Objects of NeRFBK 30
3.2 Industrial Objects of NeRFBK 31
3.3 Heritage (Indoor and Underground) of NeRFBK 32
3.4 Heritage (Outdoor) (a) of NeRFBK 33
3.5 Heritage (Outdoor) (b) of NeRFBK 34
3.6 Aerial Scenes of NeRFBK 35
3.7 Comparison among NeRFBK with other 3D reconstruction datasets . . 37

4.1 A summary of the objects used in our analyses and available in the NeRFBK dataset. 42
4.2 The evaluated methods for the 3D reconstruction of texture-less, metallic, translucent, and transparent objects. 44
4.3 Cloud-to-mesh comparison of the best-performing methods from each category on the Industrial_A object. 45
4.4 Cloud-to-mesh comparison of the best-performing methods from each category on the Metallic object. 47
4.5 Metrics for the cloud-to-mesh comparisons of the best-performing methods from each category applied to the transparent glass object 52
4.6 Metrics [mm] for the cloud-to-mesh comparisons of the tested MDE methods applied to the Industrial_A object. 54
4.7 A summary of the evaluated methods for the three different non-collaborative industrial objects. 56
4.8 Overview of datasets used for urban scene evaluation. 60
4.9 Evaluation metrics [m] across different methods on the UseGeo, Dortmund and MatrixCity datasets. Best results are highlighted in green, worst results in red. 63
4.10 Visual results of the geometry-based comparisons for the evaluated methods across all datasets. The legends in the figures are set to +/- 3σ [unit: m] considering the metrics of each methods. 64
4.11 Geometric evaluation [m] of fine-structure reconstruction between 3DGS methods and traditional photogrammetry. 66
4.12 Photometric evaluation metrics on three tested datasets. Higher SSIM/PSNR and lower LPIPS are better. 68

4.13 GPU memory usage (MB) for each method during training on the Dortmund and MatrixCity dataset. Since GPU usage is dynamic, we pick the peak values as the result. 69

5.1 Quantitative results of cloud-to-cloud comparison on the *Bottle* object. Our approach achieves the best performance in RMSE (3.16), MAE (2.71), and STD (1.62), indicating superior reconstruction accuracy and stability compared to all baselines, including the normal-only supervision variant. 84
5.2 Quantitative results of cloud-to-cloud comparison on the *Glass* object. Consistent with the results on the *Bottle* object, our method outperforms all competing approaches in terms of RMSE, MAE, and STD, demonstrating robust reconstruction quality and stable performance on transparent objects. 85
5.3 Photometric evaluation metrics on four representative scenes. Higher SSIM/PSNR and lower LPIPS indicate better quality. 87

6.1 3D Occupancy prediction performance on the Occ3D-nuScenes validation set. 100
6.2 4D occupancy forecasting performance. Aux. Sup. denotes auxiliary supervision beyond the ego trajectory. Avg. indicates the average performance over 1s, 2s, and 3s horizons. 101
6.3 Motion planning performance. Aux.Sup.denotes auxiliary supervision apart from the ego trajectory. 102
6.4 Effect of different hyperparameters for the scene tokenizer. The setting denotes the latent spatial resolution, latent channel dimension, and codebook size, respectively. 103
6.5 Efficiency comparison on the nuScenes dataset. 3D Gaussian–based representations significantly reduce memory usage compared to voxel- and BEV-based methods. 104
6.6 Ablation studies of air mask and vae separation. Each value indicates the performance on the validation dataset. 105

7.1 Comparison of SOTA editing methods and our proposed approach: 3DSceneEditor offers more diverse editing capabilities and is currently the first pipeline that supports 3D-only editing. 109
7.2 Quantitative Comparison. 3DSceneEditor outperforms existing methods in both CLIP metrics while requiring less time and GPU memory. . . . 118
7.3 Ablation study of Language-Object Correlation. This table presents the average TIS for tested object pairs and the accuracy of different image-text encoder combinations. The object with the higher TIS is selected as the final choice. 119

List of Figures

1.1 Overview of research questions and corresponding chapters. 5
1.2 Geometry evaluation of neural representations versus traditional photogrammetry. The figure shows a cross-section of the Statue object from the NeRFBK [314] dataset. 5
1.3 Visualization of 3DSceneEditor [312], a 3D-only framework for real-time, text-driven scene editing. The pipeline supports diverse operations including object removal, addition, recoloring, repositioning, and replacement. 8

2.1 The basic visual concept of NeRF scene representation (modified from [177]). 14
2.2 The basic concept of 3D Gaussian Splatting (modified from [110]). . . . 17

3.1 Visual summary of the NeRFBK datasets. Data and more info available at https://github.com/3DOM-FBK/NeRFBK 29

4.1 Examples of objects featuring non-collaborative surfaces: reflective and texture-less (a,b) or transparent and refractive (c,d). 41
4.2 An overview of the proposed procedure to assess the performance of non-collaborative surfaces using different 3D reconstruction methods. 43
4.3 The best accuracy (a) and completeness (b) for all tested methods using the Industrial_A object. 46
4.4 The best accuracy (a) and completeness (b) for all tested methods using the Metallic object. 48
4.5 Cross-section profiles on the Synthetic_Metallic object reconstructed with the best method from each category: (a) The location of the profile on the Synthetic_Metallic object, (b) Mono-NeuS for NeRF, (c) FSGS for Gaussian Splatting, and (d) GBi-Net for MVS. 49
4.6 The best accuracy (a) and completeness (b) methods among all tested approaches for the transparent glass object. 51
4.7 Visualization of MDE results for Industrial_A object. 53
4.8 Visualization of MDE results for Synthetic_Metallic object. 55
4.9 Overall framework for assessing the performance of various Gaussian Splatting methods in urban scene reconstruction 59

4.10 Qualitative graphic-based comparison of the evaluated 3DGS methods across MatrixCity (top) and UseGeo (bottom) datasets. Fine structures such as wires and windows are highlighted with close-ups for clearer visibility. 67

5.1 Overall pipeline of our approach. Given the input image and corresponding camera parameters, Neural Gaffer synthesizes relight images under various lighting conditions. These are used by DSINE to estimate surface normals, which in turn supervise the Specular Gaussian Splatting module. 74

5.2 Architecture of the relighting module. Neural Gaffer is a latent diffusion model conditioned on the input image and rotated lighting maps. . . . 75

5.3 Overview of the DSINE normal estimation process. Given an input image and camera parameters, a lightweight CNN extracts features and generates an hidden state. This hidden state is used to iteratively update surface normals by rotating neighboring pixels' normals, resulting in high-quality normal estimation. 76

5.4 Qualitative comparison on the *Bottle* scene. The baselines fail to recover complete details due to transparency and refraction. Normal supervision slightly improves surface sharpness, but our full method with relighting achieves clearer contours and more complete structure. 79

5.5 Qualitative comparison on the *Glass* scene. Similar to the *Bottle* case, the baseline methods suffer from inconsistent rendering with respect to the ground truth in refractive regions (highlighted by the red and green boxes). In contrast, our full model, incorporating normal estimation and relighting, produces clearer object boundaries and preserves finer structural details. 80

5.6 Qualitative comparison on *DIMO* scenes (containing only 14 images) containing industrial metal parts. In both views, our full model with normal and relighting supervision yields more accurate metal highlights, sharper specular reflections, and clearer background recovery. 81

5.7 Accuracy and completeness evaluation on the *Bottle* scene. Solid lines represent accuracy (distance from reconstructed points to the GT), while dashed lines denote completeness (coverage of GT points by reconstructed points). 82

5.8 Accuracy and completeness evaluation on the *Glass* scene. Solid lines represent accuracy (distance from reconstructed points to the GT), while dashed lines denote completeness (coverage of GT points by reconstructed points). 83

5.9 Visualization of Cloud-to-Cloud comparisons on the *bottle* object. . . . 85

5.10 Visualization of Cloud-to-Cloud comparisons on the *glass* object. 86

5.11 Qualitative ablation results on the *household* object. Despite the low resolution of input images, our method better reconstructs the transparent geometry compared to baselines and the normal-only variant. 88

6.1 General pipeline of RenderWorld. We firstly generate the 3D occupancy labels through an Img2Occ Module (Figure 6.2). Then, using Air Mask Variational Autoencoder (AM-VAE) described in Section 6.2.2, the separated air and non-air voxels are independently encoded into latent representations (i.e., discrete tokens). Finally, these latent representations are processed according to the specifications in Section 6.2.3, and based on this, the voxels and trajectories are predicted, ultimately outputting the predicted occupancy and self-planning. 93

6.2 Training paradigm of 2D-to-3D occupancy prediction Module. Our proposed Img2Occ Module utilizes 2D labels to train the 3D occupancy network, allowing the model to take advantage of detailed 2D pixel-level semantics and depth supervision. 94

6.3 Visualization of the forecasting and planning results of RenderWorld. . 102

7.1 Our 3D-only paradigm, named **3DSceneEditor**, consists of three key steps. First, a pre-trained instance segmentation model is applied to understand the input scene and assign a semantic label to each Gaussian. Followed by an Open Vocabulary Object Grounding module to ground the target objects from the input semantic Gaussians and generate the ROI for target objects. Finally, we edit the specified scene in ROI based on the prompt and render the edited views. 110

7.2 Visualization of our Object Grounding. We first extract the key words from the prompt (**bold fonts** in the figure). Since the positional relationships between objects in 3D space change in different viewpoints, we need to project them onto a static 2D plane to better understand the scene. 111

7.3 Visualization of our object addition pipeline. We generate new objects using a Gaussian-based generative model, guided by keywords extracted from the prompt. With the assistance of the Object Grounding module, these new Gaussians are then integrated into the ROI within the input scene. 113

7.4 Optimization mechanism of object addition and replacement. 114

7.5 Results of 3DSceneEditor. This figure presents additional results across diverse scenes, demonstrating that our method enables precise and varied scene editing for layouts and objects of different scales. 115

7.6 Qualitative comparison with respect to other controllable scene editing approaches. The experimental results in ScanNet++ and Mip-NeRF 360 datasets show that in both indoor/outdoor scene layouts, our 3D-only approach effectively utilizes 3D Gaussian spatial information, maintaining high editing quality independent of input image pixels and consistently preserving scene style, enabling more reliable and controllable editing in intricate settings. 117

7.7 Visualization of Language-Object Correlation. The keyword in the prompt (red font) is extracted and tokenized by a text encoder, then input into the vision-language model (VLM) alongside the image token. The Text-Image Similarity (TIS) is then calculated from the VLM's output. . . . 119
7.8 Visualization of KNN in our optimizion module. By using the KNN to refine the segmentation labels within ROI regions, we effectively reduces noise, and yielding more accurate editing. 120

A.1 Visualization of geometric results for Gaussian sets with varying initializations. The figure illustrates the results of using randomly initialized points versus points sampled from the ground truth (GT). The Gaussian set trained with random initialization shows significant noise in the 3D geometry, making it unsuitable for instance segmentation tasks. In contrast, the Gaussian set sampled from the GT and then trained achieves much higher geometric accuracy with minimal noise, closely aligning with the GT. 176
A.2 Visualization of instance segmentation with different confidence threshold. In this figure, we visualize instance segmentation of a 3D scene represented by a Gaussian set using colorful point clouds as Gaussians lack RGB attributes. Green bounding boxes highlight the missed segmented objects when the confidence threshold is set to $c = 0.8$ compared to $c = 0.3$, while red bounding boxes indicate additional noise or mis-segmentations introduced at the lower threshold. 177
A.3 More visualization result of 3DSceneEditor. 178
A.4 Results of object relighting 179
A.5 Visualization of spatial relation. 179
A.6 More visualization of adaptive optimization 179
A.7 Design of our user study. 181

Chapter 1

Introduction

1.1 Motivation

With the rapid development of 3D computer vision, 3D techniques started to be applied to various aspects of daily human life, including autonomous driving [64], industrial inspection [233], embodied AI [171, 309], AR and VR [225]. Modern AI methods are required to perceive, model, and reason about the 3D world with high accuracy and robustness. The core of these capabilities is the model's ability to reconstruct detailed 3D geometry from input observations and meanwhile understand the semantic and spatial information of complex scenes. Achieving precise 3D reconstruction and comprehensive scene understanding are therefore becoming fundamental prerequisites for enabling intelligent interaction between AI and the physical world [364].

Despite significant progresses in 3D vision in recent years, the generation of detailed and faithful 3D representations of real-world environments remains a challenging problem. Traditional geometry-based reconstruction pipelines [214, 212, 68] have demonstrated strong performance in many practical scenarios, particularly when sufficient image coverage, accurate camera calibration, and favorable imaging conditions are available. However, their performance can degrade in more unconstrained settings, where factors such as sparse viewpoints, complex lighting, or low-texture regions make reliable correspondence estimation difficult [210]. Recent learning-based approaches, such as neural radiance fields and related neural scene representations, have brought substantial improvements to 3D reconstruction, but they often introduce new challenges, including high computational cost, sensitivity to input sparsity, and limited flexibility for downstream applications [213, 316]. As a result, achieving efficient and high-fidelity 3D reconstruction across diverse scenes and conditions is the a central

challenge addressed in this thesis.

In addition to geometry reconstruction, understanding the reconstructed 3D environments at a higher semantic level is also a crucial problem in 3D computer vision. This includes object recognition [264], spatial relationship understanding [43, 235, 18], and enables downstream tasks such as scene editing [312, 31, 278], interaction [240, 220], and manipulation [6, 202]. However, previous 3D scene understanding methods are often designed for specific predefined tasks, such as semantic segmentation or object detection with fixed categories, and typically rely on large amounts of annotated 3D data [186]. These constraints limit their generalization and scalability, especially in layout-complex indoor and outdoor scenarios. Achieving a unified and flexible 3D scene representation for both fine-grained reconstruction and highly generalized scene understanding remains an open research problem.

Recent developments in neural scene representations, such as RenderOCC [192], Panoptic nerf [58] and Instruct-nerf2nerf [73] provide a promising direction for addressing actual 3D reconstruction and semantic enrichment challenges [274]. By modeling geometry, appearance, and semantic information in a continuous and differentiable form, neural representations offer a powerful alternative to classical models. However, many existing approaches are either implicit in nature [178], limiting direct manipulation and interpretability, or require expensive training [2] and dense supervision [329].

The overarching goal of this thesis is to advance the development of neural scene representations, with a particular focus on 3D Gaussian Splatting [110], for both 3D reconstruction and scene understanding in real-world scenarios. While 3DGS has demonstrated strong potential as an efficient and explicit representation for image-based rendering, its application beyond this domain remains an open and actively evolving research direction. Rather than assuming it as a fully mature solution, we treat 3DGS as a promising intermediate representation whose capabilities and limitations are still being explored. The research conducted in this thesis follows a coherent roadmap with four tightly connected objectives (Section 1.2).First, we propose a new dataset for benchmarking and evaluating learning-based 3D reconstruction frameworks under challenging conditions such as sparse views, large scale, and non-collaborative surfaces (Chapter 3). Second, we explore enhanced geometric supervision and novel view synthesis to improve reconstruction quality while preserving fine-grained surface details (Chapter 5). Third, we study how explicit 3D representations can be leveraged to improve outdoor scene reconstruction and support holistic scene understanding using minimal or no 3D annotations (Chapter 6). Finally, we extend 3DGS representations to downstream applications, with a particular focus on interactive and controllable 3D

scene editing in complex indoor environments (Chapter 7).

Overall, in this thesis, we aim to bridge the gap between high-quality 3D reconstruction and rich 3D scene understanding by developing efficient, explicit, and robust 3D Gaussian representations across diverse 3D scenes. In the following sections, we will detail the research questions and challenges addressed throughout the PhD thesis.

1.2 Research Questions and Challenges

In this thesis, we focus on advancing 3D Gaussian Splatting from two closely connected perspectives: 3D reconstruction and 3D scene understanding. Based on this motivation, we formulate the following research questions, which guide the investigations presented in this work:

Research Question 1: How can we establish a high-quality dataset for systematically evaluating the performance of radiance field representations against traditional photogrammetry methods?
Before advancing new methodologies, it is critical to quantify where existing representations fail. While Neural Radiance Fields (NeRF) and 3D Gaussian Splatting (3DGS) offer superior rendering, their reliability in terms of metrological accuracy across different scales remains unclear. By introducing the NeRFBK [314] dataset and conducting comparative studies [316, 315, 213] against traditional photogrammetry, this research aims to identify the specific failure modes of these paradigms in both industrial and geographic contexts. The insights gained here provide the foundational evidence needed to address geometric instabilities in more complex downstream tasks (Answered in Section 1.3.1).

Research Question 2: How can 3DGS decoupling appearance from geometry through physical priors overcome the reconstruction limits of non-collaborative surfaces?
Based on the limitations identified in RQ1, specifically the geometric collapsing of neural representations on transparent or metallic surfaces, this question focuses on enhancing object-level fidelity. For non-collaborative objects, vanilla 3DGS [110] struggles to capture fine-grained, high-frequency geometric details. We therefore investigate a pipeline that incorporates surface normal supervision and object relighting to decouple an object's radiance from its underlying geometry. By introducing additional geometric constraints, our method effectively mitigates the challenges posed by transparent and

reflective surfaces, resulting in denser and higher-fidelity reconstructions (Answered in Section 1.3.2).

Research Question 3: How can 3D Gaussian Splatting be extended into an efficient world representation for large-scale dynamic scene understanding and forecasting?
Moving beyond static objects, we address the challenge of large-scale, complex scene reconstruction and understanding required for autonomous driving. This stage evolves the geometric foundations of RQ2 into a functional World Model. Through the RenderWorld [311] framework, we explore how 3DGS can be utilized to generate self-supervised 3D occupancy labels (with scenes represented as GS-based occupancy enriched with semantic information) and to support 4D motion forecasting. The goal is to overcome the prohibitive computational cost of prior NeRF-based approaches, enabling real-time reconstruction and semantic segmentation that meet the strict efficiency requirements of autonomous driving systems (Answered in Section 1.3.3).

Research Question 4: How can 3D Gaussian Splatting serve as a unified and efficient representation for advanced scene understanding and downstream 3D scene manipulation?
Building on RQ3, we observe that 3DGS naturally provides a point-based, discrete representation, allowing each Gaussian to carry semantic and instance-level information. This property enables RenderWorld to generate semantically enriched occupancy labels directly from 3DGS. Leveraging this insight, we propose 3DSceneEditor [312], a fully 3D-only framework for automated and interactive scene manipulation. User-driven edits—such as object replacement, deletion, etc., are applied directly to Gaussians, avoiding the computational redundancy and multi-view inconsistencies of traditional 2D-to-3D pipelines. This demonstrates that 3DGS can serve as a unified and efficient representation for both scene understanding and real-time 3D scene editing (Answered in Section 1.3.4).

To provide a clear overview of the thesis organization, Figure 1.1 summarizes the relationship between the research questions and their associated chapters.

1.3 Contributions

Based on the aforementioned research questions, the specific contributes of this thesis are detailed as follows.

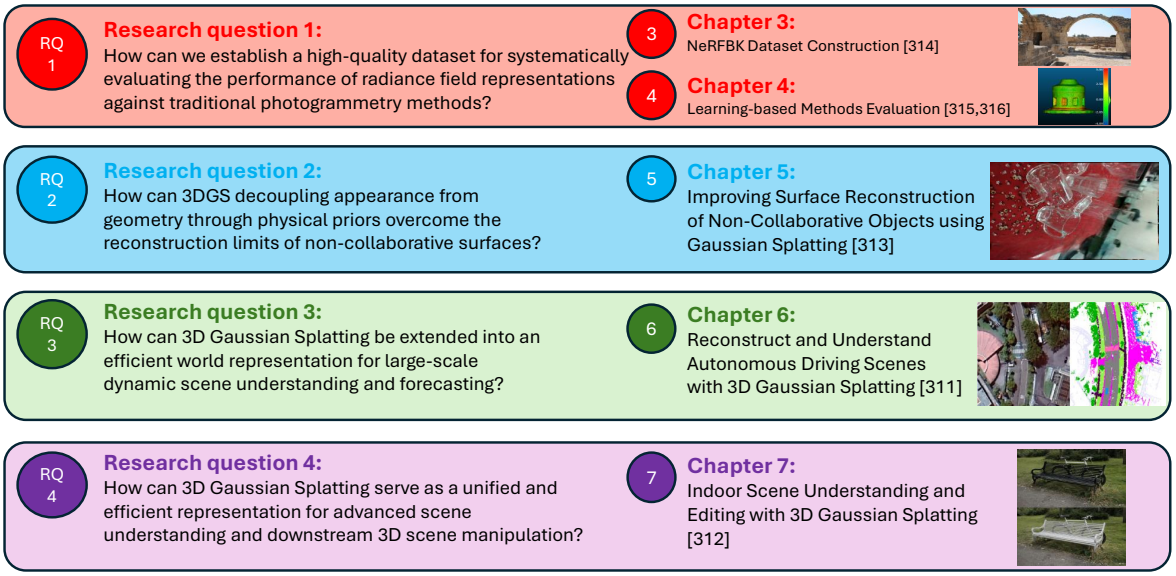


Figure 1.1: Overview of research questions and corresponding chapters.

1.3.1 Comprehensive Dataset and Evaluation for Radiance Field Representations

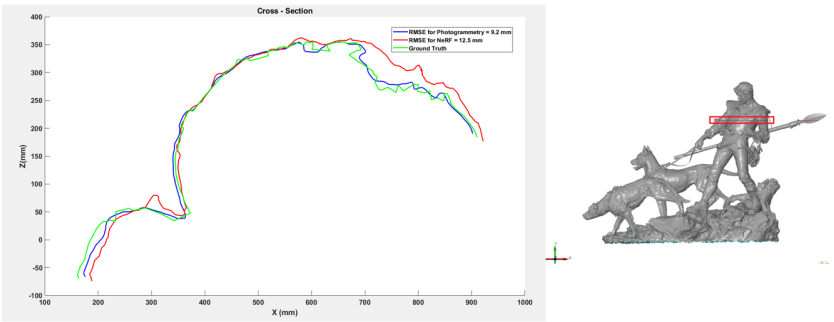


Figure 1.2: Geometry evaluation of neural representations versus traditional photogrammetry. The figure shows a cross-section of the Statue object from the NeRFBK [314] dataset.

Radiance field representations, including Neural Radiance Fields and 3D Gaussian Splatting, have emerged as powerful paradigms for image-based 3D reconstruction due to their ability to model complex appearance effects and to operate without explicit geometric supervision [62, 23]. While these methods have shown strong performance on a variety of object- and scene-level benchmarks, their evaluation is often conducted on narrowly scoped datasets with limited scene diversity, scale variation, or surface com-

plexity. In particular, challenging scenarios such as texture-less, metallic, reflective, or transparent surfaces, as well as multi-scale indoor, outdoor, and aerial environments, remain underrepresented in existing benchmarks [316]. Moreover, the lack of consistent, high-quality ground truth across real and synthetic data further hinders objective comparison and reproducibility. These limitations make it difficult to systematically analyze the strengths, failure modes, and generalization behavior of radiance-field-based reconstruction methods across different application domains.

Our thesis addresses these limitations by introducing NeRFBK (Chapter 3), a comprehensive benchmark dataset designed for the systematic evaluation of radiance field representations in diverse and challenging 3D reconstruction scenarios.
NeRFBK combines real-world and synthetic datasets with accurate ground-truth geometry, high-resolution imagery, and precise camera calibration, covering a wide range of scales, surface properties, and acquisition settings. The dataset spans multiple application domains, including industrial inspection, transparent and reflective objects, cultural heritage documentation, and aerial urban reconstruction. By jointly providing object-level and scene-level data with varying material characteristics and geometric complexity, NeRFBK enables consistent and reproducible evaluation of NeRF- and 3DGS-based methods under realistic conditions. In addition, our thesis presents a systematic experimental analysis conducted on NeRFBK, employing standard photogrammetric metrics and complementary evaluation criteria to assess reconstruction accuracy, completeness, and robustness (Chapter 4). Together, the dataset and evaluation framework establish a unified basis for benchmarking radiance field representations, facilitating fair comparison with traditional image-based methods and supporting the development of more reliable and scalable 3D reconstruction techniques.

1.3.2 Improving Reconstruction Quality with Normal Supervision and Relighting

Precise 3D reconstruction of non-collaborative surfaces remains a fundamental challenge in the 3D vision field, where traditional methods often fail to establish reliable geometric correspondences. Existing NeRF- and 3DGS-based methods achieve high-fidelity reconstructions, yet their performance degrades under sparse input views, uncontrolled illumination, and highly reflective or transparent surfaces [150, 335, 157]. To address these limitations, it is essential to incorporate surface normal guidance and illumination-aware supervision to improve reconstruction fidelity on complex materials.

To this end, we propose a novel 3DGS-based reconstruction framework (Chapter 5)

that integrates surface normal supervision and relighting for enhanced reconstruction of reflective and transparent objects. Our approach first generates multi-illumination relight images using a pre-trained relighting network, and then co-supervises the 3DGS reconstruction pipeline with the corresponding surface normal maps. Extensive experiments demonstrate that our method significantly improves reconstruction fidelity for challenging real-world and synthetic non-collaborative objects compared to baseline approaches, validating the effectiveness of leveraging explicit normal supervision and relighting guidance within the 3DGS framework.

1.3.3 Extending 3D Gaussian Splatting to Autonomous Driving Scene Reconstruction and Understanding

Even though Contribution 2 addressed object-level reconstruction, scaling up to scene-level reconstruction and understanding, such as in autonomous driving, remains challenging because scene representation quality directly impacts motion planning, forecasting, and overall system safety [255]. Existing autonomous driving pipelines typically separate perception, forecasting, and planning [26], relying on 3D target detection via LiDAR and camera fusion [87, 135, 140]. While these methods are effective for object detection, they often fail to capture fine-grained scene information, limiting robustness for downstream planning and decision-making [238]. Furthermore, LiDAR sensors are expensive, and multimodal fusion introduces heavy computational overhead, posing challenges for real-time deployment.

To overcome these limitations, we introduce **RenderWorld** (Chapter 6), a unified framework for autonomous driving scene reconstruction, understanding, and prediction, built upon 3D Gaussian Splatting (3DGS). RenderWorld employs a self-supervised Img2Occ module that generates 3D occupancy labels by anchoring 3DGS and supervising it with multi-view 2D depth and semantic images rendered via Gaussian rasterization. This 3D occupancy representation provides a structured, explicit, and dense encoding of the environment, enhancing downstream forecasting and motion planning. To further improve spatial granularity, we combine 3DGS with a specifically designed Air Mask Variational Autoencoder (AM-VAE), which separately encodes air and non-air voxels. Extensive comparisons with multiple baselines in top-tier venues demonstrate that our approach not only improves scene-level detail through high-quality image rendering but also reduces memory consumption compared with NeRF-based methods.

1.3.4 3D-Only Indoor Scene Understanding and Editing with 3D Gaussian Representation

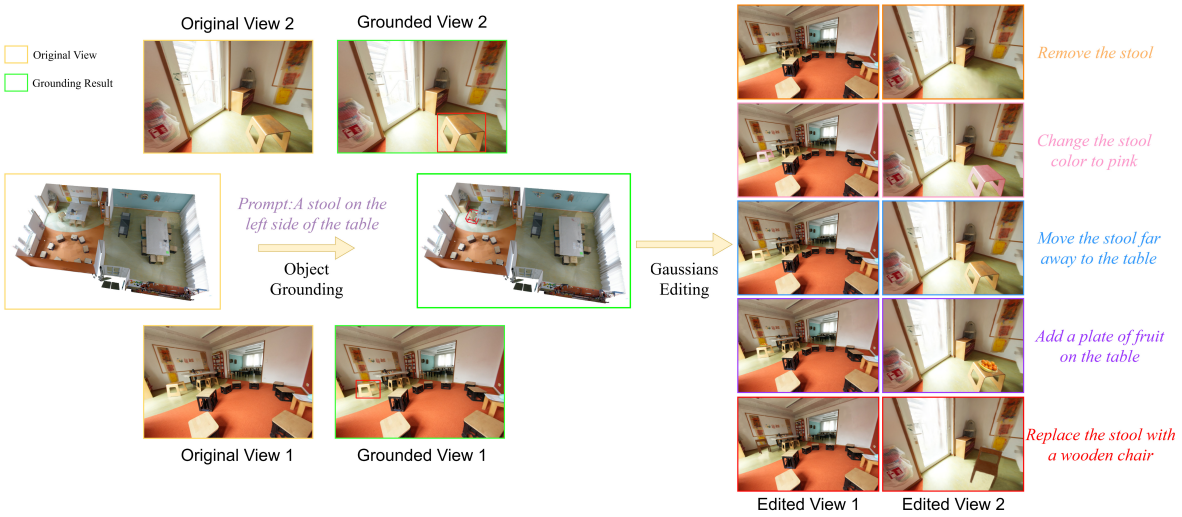


Figure 1.3: Visualization of 3DSceneEditor [312], a 3D-only framework for real-time, text-driven scene editing. The pipeline supports diverse operations including object removal, addition, recoloring, repositioning, and replacement.

Our fourth contribution focuses on indoor scene understanding and related downstream applications, such as interactive editing. The creation and manipulation of complex 3D scenes remain time-consuming and costly, requiring manual efforts for object placement, layout design, and material selection [194]. While generative AI methods using text-to-3D or image-to-3D pipelines can accelerate scene generation, they typically rely on multi-step 2D-to-3D projections, pre-trained NeRF models, or diffusion-based editing frameworks [200, 90, 73], which limit interactive control, precision, and efficiency. NeRF-based methods [223, 242, 287] suffer from high computational cost and low manipulability due to their implicit representations. Existing Gaussian Splatting-based editing approaches [31, 278, 254] improve rendering speed and visual quality but often rely on 2D pre-segmentation, diffusion models, or low-resolution inputs, which constrains scene manipulation flexibility and interactive control. These limitations motivate the development of a fully 3D editing framework capable of real-time, high-fidelity, and user-controllable modifications. Then, We propose **3DSceneEditor** (Chapter 7), a fully 3D controllable editing framework built upon 3DGS. By leveraging the explicit, point-level nature of Gaussian representations, our method encodes semantic information directly at the scene level, enabling precise scene modifications from a single text or

image prompt. The framework combines pre-trained instance segmentation for semantic labeling of Gaussians and the modifications are applied directly to 3D Gaussians, avoiding the inefficiencies and inconsistencies of 2D-to-3D pipelines. Extensive experiments and benchmarks on complex indoor 3D layouts validate the framework's effectiveness in terms of fidelity, efficiency, and user-interactivity.

1.4 Outline

This thesis is organized into 8 chapters.

Chapter 2 provides a comprehensive background and state of the art review of research related to this thesis.

Chapter 3 presents the NeRFBK dataset, which is designed to evaluate neural field representations in comparison with traditional photogrammetry methods. This chapter is based on our publication in the ISPRS Archives 2023 [314].

Chapter 4 presents a systematic evaluation of learning-based 3D reconstruction methods for industrial objects and urban-scale scenes. This chapter is based on our publications in *Metrology* [315] and the ISPRS Archives 2025 [316].

Chapter 5 presents a surface reconstruction method for non-collaborative objects based on 3DGS. The content of this chapter is based on our in submission paper *Colla-GS* [313], which addresses the challenges of reconstructing reflective, transparent, and glossy surfaces through relighting-based surface normal supervision.

Chapter 6 presents RenderWorld, a Gaussian-based world model for autonomous driving that enables 3D occupancy generation, prediction, and motion planning from 2D multi-view images. This chapter is based on our publication at ICRA 2025 [311].

Chapter 7 presents a novel open-vocabulary, zero-shot 3D scene understanding and editing method, named *3DSceneEditor*, built upon 3DGS. This chapter is based on our publication at WACV 2026 [312].

Chapter 8 concludes the thesis by summarizing its main contributions. It also reflects on the role and potential of 3D Gaussian Splatting and discusses promising directions for future research.

1.5 Publications

This thesis builds upon the following 6 publications [314, 315, 316, 311, 312, 313]

NeRFBK: A Holistic Dataset for Benchmarking NeRF-Based 3D Reconstruction (Chapter 3)

Ziyang Yan, Gabriele Mazzacca, Simone Rigon, Elisa Mariarosaria Farella, Pawel Trybala, Fabio Remondino

International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences (ISPRS Archives), 2023

Learning-Based 3D Reconstruction Methods for Non-Collaborative Surfaces — A Metrological Evaluation (Chapter 4)

Ziyang Yan, Nazanin Padkan, Paweł Trybała, Elisa Mariarosaria Farella, Fabio Remondino

Metrology, 2025

Evaluating 3D Gaussian Splatting for Urban Scene Reconstruction (Chapter 4)

Ziyang Yan, Mengrui Yin, Yihua Shao, Fabio Remondino

International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences (ISPRS Archives), 2025

Colla-Gaussian: High-Quality Surface Reconstruction of Non-Collaborative Objects with Gaussian Splatting (Chapter 5)

Ziyang Yan, Xingyu Liu, Yihua Shao, Fabio Remondino

Under Review by Journal of Photogrammetry, Remote Sensing and Geoinformation Science (PFG)

RenderWorld: World Model with Self-Supervised 3D Labels (Chapter 6)

Ziyang Yan, Wenzhen Dong, Yihua Shao, Yuhang Lu, Haiyang Liu, Jingwen Liu, Haozhe Wang, Zhe Wang, Yan Wang, Fabio Remondino, Yuexin Ma

IEEE International Conference on Robotics and Automation (ICRA), 2025

3DSceneEditor: Controllable 3D Scene Editing with Gaussian Splatting (Chapter 7)

Ziyang Yan, Lei Li, Yihua Shao, Siyu Chen, Zongkai Wu, Jenq-Neng Hwang, Hao Zhao, Fabio Remondino

The IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), 2026

Besides the aforementioned papers that constitute the core chapters of this thesis, I collaborated with other FBK-3DOM colleagues or teammates of the 2 institutions I visited during my 1-year period abroad. This produced other scientific publications as follows:

MeshPart: Few-Shot Part-to-Part Articulated Mesh Generation

Ziyang Yan, Yihua Shao, Mengrui Yin, Junzhou Chen, Yinghao Chen, Fabio Remondino, Hao Zhao, Jingcai Guo

In submission

GM-MoE: Low-Light Enhancement with Gated-Mechanism Mixture-of-Experts

Minwen Liao, Hao Bo Dong, Xinyi Wang, Yihua Shao, **Ziyang Yan**

International Conference on Computer Vision (ICCV), 2025

NeAR: Coupled Neural Asset–Renderer Stack

Hong Li, Chongjie Ye, Houyuan Chen, Weiqing Xiao, **Ziyang Yan**, Lixing Xiao, Zhaoxi Chen, Jianfeng Xiang, Shaocong Xu, Xuhui Liu, Yikai Wang, Baochang Zhang, Xiaoguang Han, Jiaolong Yang, Hao Zhao

IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2026

Unifying Appearance Codes and Bilateral Grids for Driving Scene Gaussian Splatting

Nan Wang, Yuantao Chen, Lixing Xiao, Weiqing Xiao, Bohan Li, Zhaoxi Chen, Chongjie Ye, Shaocong Xu, Saining Zhang, **Ziyang Yan**, Pierre Merriaux, Lei Lei, Tianfan Xue, Hao Zhao

Advances in Neural Information Processing Systems (NeurIPS), 2025

Evaluating Monocular Depth Estimation Methods on Industrial Objects

Nazanin Padkan, **Ziyang Yan**, Fabio Remondino

The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences (ISPRS Archives), 2025

A Critical Analysis of NeRF-Based 3D Reconstruction

Fabio Remondino, Ali Karami, **Ziyang Yan**, Gabriele Mazzacca, Simone Rigon, Rongjun Qin

Remote Sensing, 2023

Chapter 2

Background, State of the Art and Related Works

In this chapter, we provide a background overview of the research fields related to this thesis. We first provide an overview of neural field representations in Section 2.1. We then discuss the state of the art of 3D reconstruction using 3DGS and novel view synthesis in Section 2.2, and 3D scene understanding and editing with 3DGS in Section 2.3. In Section 2.4, we review additional related works that are directly relevant to the methodologies proposed in this thesis.

2.1 Development of Neural Field Representations

2.1.1 Neural Radiance Fields (NeRF)

Neural Radiance Fields (NeRF) was first introduced by Mildenhall *et al.* in 2020 [177]. NeRF-based methods normally use a set of images, together with their associated 3D camera positions and view directions (i.e., oriented images), as input and outputs for the volume density and view-dependent emitted radiance [62]. This principle is shown in Figure 2.1. All the NeRF-based approaches use a neural network, which learns the volumetric 3D repre-sentation of an object from multi-view 2D images. Then, by feeding the network a new camera position and view direction, it can perform socalled novel view synthesis, which predicts the emitted color and volume density of the scene seen from the selected pose.

NeRF is a type of neural implicit representation method which encodes a scene using an optimizable continuous function. Assuming that the 3D object is located at the center of the modeled space, for each 3D point coordinate $\mathbf{p} = (x, y, z)$ and its view

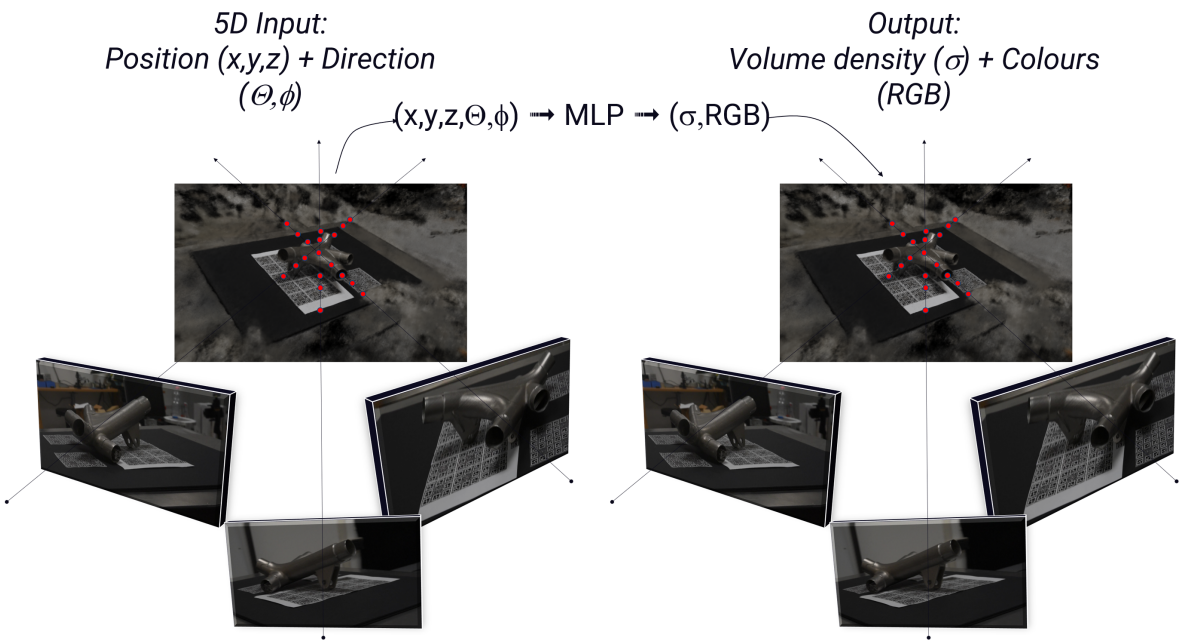


Figure 2.1: The basic visual concept of NeRF scene representation (modified from [177]).

direction $\mathbf{d} = (\theta, \phi)$, we can obtain the following relationship:

$$(\sigma, \mathbf{c}) = f_\theta(\mathbf{p}, \mathbf{d}), \tag{2.1}$$

where σ and $\mathbf{c}$ are the density and color of a point, respectively, and θ represents the parameters of the continuous function f.

The implementation structure of a NeRF mainly consists of an encoder and a decoder. The encoder usually leverages a convolutional neural network (CNN) which is responsible for extracting the spatial position and perspective features of each point in the scene from the input multi-view images and camera parameters. Each convolutional layer in the encoder maps the input data from a low-dimensional space to a high-dimensional space, extracting increasingly complex feature representations.

The decoder is typically represented as a multi-layer perceptron (MLP) network that generates a continuous 3D radiance field from the features extracted by the encoder. It accepts as input the spatial location and perspective features of each point produced by the encoder, and outputs the corresponding color and density values. Each MLP layer in the decoder maps the input to another high-dimensional space and extracts more complex representations.

The original NeRF implementation [177], as well as subsequent derivative methods,

utilize a non-deterministic stratified sampling approach, described as follows. The ray emitted from the camera is divided into N equally spaced bins, and a sample is uniformly drawn from each bin. The color of the ray (i.e., the pixel color) is obtained by volumetric ray marching:

$$C(\mathbf{r}) = \sum_{i=1}^{N} T_i \left(1 - \exp(-\sigma_i \delta_i)\right) \mathbf{c}_i, \tag{2.2}$$

where

$$T_i = \exp\left(-\sum_{j=1}^{i-1} \sigma_j \delta_j\right), \quad \delta_i = t_{i+1} - t_i \tag{2.3}$$

is the interval between adjacent sampled points.

The original method calculates the loss function by comparing a rendered pixel value for camera ray $\mathbf{r}$ with the corresponding ground truth pixel value C^{gt} for all camera rays of the target view with pose $\mathbf{p}$. Thus, the loss function $\mathcal{L}$ is given by:

$$\mathcal{L} = \sum_{\mathbf{r} \in \mathcal{R}(\mathbf{p})} \|C(\mathbf{r}) - C^{gt}\|_2^2, \tag{2.4}$$

where $\mathcal{R}(\mathbf{p})$ is the set of all camera rays for the target pose $\mathbf{p}$.

Multi-View Dependent NeRF

The NeRF approach uses an MLP network to represent a 3D scene as a learnable, continuous volumetric scene function and render the scene by optimizing the scene function. However, the original NeRF implementation can only deal with simple reflection scenarios and struggles with processing complex reflective surfaces of industrial objects. Recently, more studies related to NeRF have started to pay attention to complex surface reconstruction. Nerfren [70] introduced NeRFReN which uses separate trans-mitted and reflected Neural Radiance Fields to process complex reflection scenes. [99] designed Dex-NeRF, which estimates the depth from transparent objects through a transparency-aware depth rendering method based on finding the first sample along the ray whose density is higher than the fixed threshold. Ref-NeRF [269] decomposes specular reflection and diffuse reflection from the target object and uses the viewing vector estimated by MLP to render the scenes. Nevertheless, it results in an enormous increment of parameters and computation. IBL-NeRF [36] ingeniously classifies indoor reflection from an indoor scene by prefiltered radiance fields. The limitation is that it is suitable for large-scale scene rendering rather than single object rendering, and this is especially relevant for the reconstruction of isolated transparent objects with per-

fect-mirror reflection. To improve the performance with less-observed and texture-less areas, MonoSDF [340] applies monocular geometry prediction and utilizes depth and normal cues predicted by monocular estimators. However, the reconstruction results are susceptible to the changes in the quality of the cues. Neuralangelo [138] is the advanced version of Instant-NGP [183], which combines multi-resolution 3D hash grids with neural surface rendering to reduce noise and then facilitate high-fidelity 3D surface reconstruction from large-scale scenes. However, multi-view dependent NeRF requires dozens or even hundreds of images to achieve high-quality scene rendering. The insufficient input of images or lack of images from specific perspectives will often cause geometric shape errors and missing surface areas [47].

Few/Single Shot NeRF

The limitation of multi-view dependent NeRF promotes the research and development of few-shot NeRF and even single shot NeRF [35]. Some of the existing methods achieve this goal by regularizing the geometry of the scene. DS-NeRF [47] utilizes sparse depth outputs from Structure-from-Motion as supervision, while DDP-NeRF [217]further obtains dense depth supervision from sparse inputs through a CNN network. RegNeRF [189] regularizes the geometry and appearance by proposing a depth smoothness loss and a pre-trained normalizing flow color model. Moreover, SimpleNeRF [234] trains two additional models, which, respectively, reduce positional encoding frequencies and remove view-dependent components. Pixelnerf [336] per-forms single-view reconstruction by first extracting the features from the input image through a CNN network and then projecting the points sampled from the camera ray onto the image plane. The novel perspectives are then rendered by applying bilinear interpolation between pixel features to extract the corresponding image feature vector. Applying the diffusion model with NeRF is also a common way to achieve this goal. DiffusioNeRF [299] uses a trained diffusion model that can regularize the distribution of RGB-D patches from perturbed viewpoints. GANeRF [218] learns the patch distribution of the scene using an adversarial discriminator that provides feedback for radiation field reconstruction, thereby improving realism in a 3D consistent manner. ReconFusion [297] utilizes a diffusion prior-based NeRF by utilizing CLIP [206] to embed the feature vectors and PixelNeRF [336] to render a feature map. The latter includes the corresponding camera and geometric information, allowing the diffusion model to predict and generate novel perspectives.

2.1.2 3D Gaussian Splatting (3DGS)

Contrary to NeRF, 3D Gaussian Splatting represents a scene using a set of explicit geometric primitives, where each element is modeled as a parameterized 3D Gaussian. This representation provides a continuous point-based description of geometry and appearance, enabling efficient rendering and optimization. While the idea of splatting originates from classical Elliptical Weighted Average (EWA) techniques, its adaptation to high-fidelity 3D reconstruction and novel view synthesis was recently introduced by Kerbl *et al.* [110], significantly advancing real-time neural rendering.

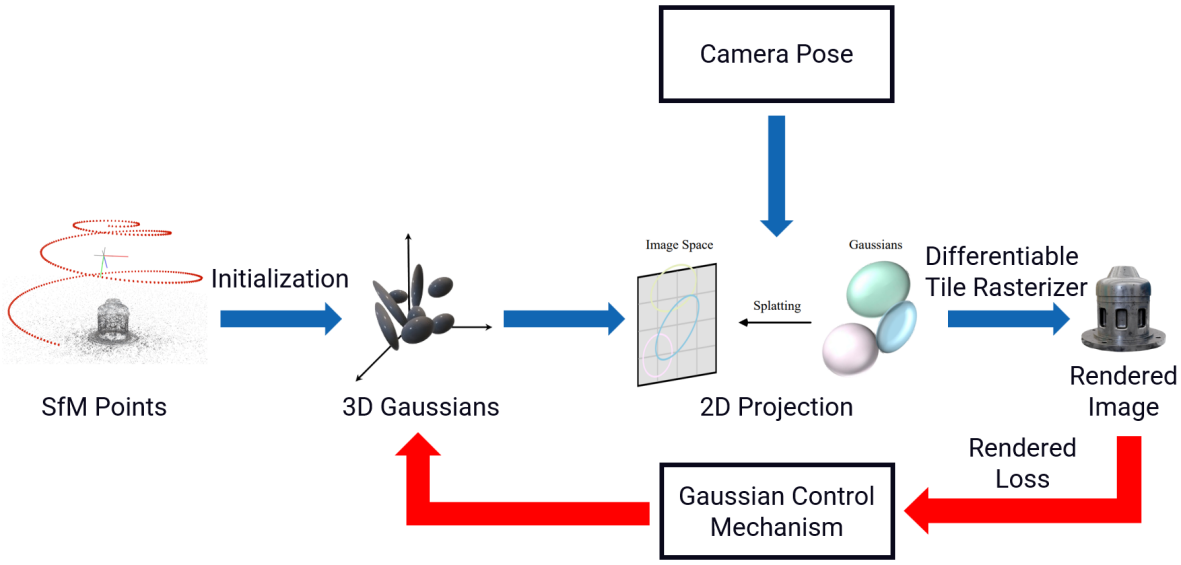


Figure 2.2: The basic concept of 3D Gaussian Splatting (modified from [110]).

Refer to figure 2.2, the construction of a 3DGS model typically starts with camera pose estimation and sparse structure recovery from multi-view images. The resulting point cloud serves as an initialization for the Gaussian primitives. Each Gaussian is defined by a mean position $\mu \in \mathbb{R}^3$ and a covariance matrix Σ that captures its spatial extent and anisotropy. The Gaussian density at a spatial location x is expressed as:

$$G(x) = \exp\left(-\frac{1}{2}(x-\mu)^\top \Sigma^{-1}(x-\mu)\right). \tag{2.5}$$

To maintain numerical stability and enforce positive definiteness, the covariance matrix is factorized into a rotation component R and a diagonal scaling matrix S, such that:

$$\Sigma = RSS^\top R^\top. \tag{2.6}$$

For rendering, each 3D Gaussian is projected into the image plane through the camera transformation. The projected covariance Σ' in screen space is computed via:

$$\Sigma' = JW\Sigma W^\top J^\top, \tag{2.7}$$

where W denotes the world-to-camera transformation and J is the Jacobian of the perspective projection.

Image synthesis is performed using front-to-back alpha compositing. Pixels accumulate color contributions from Gaussians ordered by depth, resulting in the final pixel color:

$$C = \sum_{i=1}^{N} c_i \alpha_i \prod_{j=1}^{i-1} (1 - \alpha_j), \tag{2.8}$$

where c_i represents the color of the i-th Gaussian and α_i corresponds to its opacity-weighted contribution in screen space.

Compared to implicit representations such as NeRF, 3DGS explicitly stores scene elements, allowing highly efficient GPU-based rasterization and real-time rendering. Each Gaussian is parameterized by its position, covariance, color, and opacity, and these parameters are optimized end-to-end using stochastic gradient descent within a fully differentiable rendering pipeline. During training, gradients are propagated from the image domain back to the Gaussian parameters through a differentiable rasterizer.

The optimization objective minimizes the discrepancy between rendered images and ground-truth views by combining a pixel-wise $\mathcal{L}_1$ loss with a perceptual structural similarity term:

$$\mathcal{L} = (1 - \lambda)\, \mathcal{L}_1 + \lambda\, \mathcal{L}_{\text{D-SSIM}}, \tag{2.9}$$

where λ controls the trade-off between low-level color accuracy and perceptual consistency. This formulation enables 3D Gaussian Splatting to achieve high-quality visual results while maintaining interactive rendering performance.

The emergence of 3DGS has revolutionized the research trend of 3D reconstruction. a large number of GS-based research works [122, 88, 59, 371] [50-58] have sprung up in just a few months after the launch of Gaussian Splatting. To further optimize Gaussian Splatting, Mesh-GS [63] proposed a multi-scale Gaussian Splatting method that reduces the aliasing produced in signal sampling by adjusting the size of the Gaussians based on the image resolution. Similar to NeRF, the development of Gaussian Splatting is also moving towards few/single views as the input. [39] proposed a dense depth map generated using a pre-trained MDE model to mitigate the overfitting ren-

dering problem that occurs in the novel viewpoints. SparseGS [305] incorporates depth and diffusion constraints along with an artifact's removal technique to further improve the quality of generated novel perspectives. Using only a few views as the input for SfM, the generated point clouds will be very sparse. FSGS [369] addresses this issue by introducing a Proximity-guided Gaussian Unpooling algorithm to densify the initial sparse point cloud. This method applies KNN to grow external 3D points based on the Euclidean distance from the closest original 3D points. The efficiency of Gaussian Splatting has led to its swift adoption across various domains, such as 3D geometry generation [252, 331], dynamic scene rendering [323, 92, 294], the creation of animatable 3D human models [180, 370, 203, 49, 32], real-time surgical reconstruction [160], and SLAM [130, 310, 347]. At the same time, more and more GS-based methods start to focus on reconstructing non-collaborative surfaces. Scaffold-GS [169] introduces a hierarchical 3D Gaussian scene model with anchor points initialized from SfM to enhance the ability to capture scene local details, especially for reflective, transparent, or texture-less regions. GaussianShader [105] employs a stream-lined shading function on 3D Gaussians to improve the accuracy of normal estimation, thus elevating rendering quality in scenes featuring reflective surfaces. The high efficiency in reconstruction and rendering, together with its outstanding performance, has enabled 3DGS to rapidly replace NeRF as one of the most popular representations in neural field modeling. In the following sections, we will discuss in depth the latest progress of 3DGS in both reconstruction and scene understanding.

2.2 Scenarios for GS-based 3D Reconstruction

In this section, we provide an overview of the state-of-the-art GS-based methods for 3D reconstruction across multiple application scenarios.

UAV-based reconstruction: DroneSplat [250] proposes a 3DGS framework for in-the-wild drone imagery, removing dynamic distractors via adaptive masking and enhancing reconstruction quality through multi-view stereo and voxel-guided optimization. Similarly, [204] introduces a geometry-aware approach for large-scale UAV reconstruction, leveraging dense LiDAR point clouds for geometric supervision and a spatial–temporal spherical harmonics model to improve photometric consistency under varying illumination.

Non-collaborative surfaces: GS-2DGS [258] enhances Gaussian Splatting for reflective object reconstruction by incorporating geometric supervision into 2D Gaussian representations, mitigating noisy surface extraction and improving geometric fidelity

for highly view-dependent glossy surfaces. GlossyGS [120] addresses the inherent ambiguities of glossy surface modeling in 3DGS-based inverse rendering by integrating micro-facet geometry priors and normal distribution regularization, enabling more accurate decomposition of geometry and specular materials. Xie et al. [303] further improve reflective object modeling by combining physically-based BRDF attributes, illumination–geometry decoupling, and anisotropic spherical Gaussian lighting representations, achieving precise specular reflection modeling and improved geometric consistency through visibility-aware optimization and Gaussian pruning.

Autonomous driving: RadarSplat [117] extends 3DGS to radar-based driving scenes by integrating explicit radar noise modeling, enabling realistic radar data synthesis and improved geometric reconstruction under noisy sensing conditions. AutoSplat [112] proposes a constraint-driven 3DGS framework for autonomous driving, enforcing geometric and reflection consistency while modeling temporally-dependent appearance variations to handle dynamic objects and sparse multi-view inputs.

Indoor scenes: GaussianRoom [300] enhances 3DGS for indoor scene reconstruction by leveraging neural SDF guidance and monocular cues to improve geometry accuracy and real-time rendering, especially in large textureless regions. IndoorGS [222] exploits 2D/3D geometric cues and an adaptive density control strategy to achieve more accurate geometry and higher rendering quality in indoor environments.

Urban-scale scenes: Yuan et al. [345] propose a framework for fast, high-fidelity urban scene reconstruction with level-of-detail control, visibility-based image selection, and depth/scale regularization to handle multi-view appearance variations. GS-Diff [179] uses diffusion-guided 3DGS to robustly reconstruct large-scale unconstrained urban scenes from sparse, uneven multi-view inputs, integrating depth priors, dynamic object modeling, and anisotropy regularization to improve geometric and photometric consistency.

Feed-forward pose-free reconstruction: A growing trend in 3DGS research is feed-forward reconstruction without known camera poses, first introduced by Dust3R [282]. FreeSplatter [307] reconstructs high-quality 3D Gaussians from unposed sparse-view images while estimating camera parameters within seconds, enabling efficient object- and scene-level reconstruction. SelfSplat [108] extends this paradigm to pose-free and 3D-prior-free reconstruction, integrating self-supervised depth and pose estimation to achieve accurate, cross-dataset consistent geometry and appearance. EGGS [353] introduces a hybrid 2D/3D Gaussian Splatting representation to enforce multi-view consistency and balance appearance and geometry for novel view synthesis in a feed-forward manner, eliminating the need for known camera poses.

2.3 3D Scene Understanding and Editing with 3D Gaussian Splatting

In this section, we provide an overview of the state-of-the-art GS-based methods for 3D scene understanding and editing.

2.3.1 3D Scene Understanding

Conventional instance recognition methods are typically designed under a closed-set assumption, where classifiers are trained on a fixed and limited set of categories. This setting restricts their generalization ability and practical deployment in real-world scenes and their reliance on implicit semantic representations further constrains flexibility and scalability [295].

Recently, the emergence of visual-language models (VLMs) has significantly promoted open-vocabulary learning within 3DGS frameworks. By exploiting the cross-modal alignment learned in VLMs, these approaches bridge the gap between closed-set and open-set scenarios, while the explicit representation of 3DGS alleviates the limitations imposed by implicit semantic modeling [318]. CAGS [246] addresses cross-view granularity inconsistency in open-vocabulary 3DGS by propagating contextual features through local Gaussian graphs and employing mask-centric contrastive learning to enhance coherent instance reasoning. EgoGaussian [350] and EgoSplat [124] extend 3DGS to egocentric dynamic scene understanding, where EgoGaussian jointly reconstructs scenes and tracks rigid object motion from RGB input alone, while EgoSplat enhances open-vocabulary semantic consistency via multi-view instance aggregation and spatial–temporal transient modeling. GaussianGraph [284] augments 3DGS-based open-world understanding with adaptive semantic clustering and explicit scene graph generation, enabling improved object segmentation and reliable spatial reasoning through 3D consistency verification. SOVGaussian [146] tackles sparse-view open-vocabulary reconstruction by introducing depth-constrained language fields with language-aware depth distillation and language-guided Gaussian pruning to maintain geometry–semantic consistency. OpenSplat3D [198] and ShelfGaussian [358] explore open-vocabulary 3D instance modeling under different supervision paradigms, where OpenSplat3D leverages SAM-guided contrastive feature splatting for label-free instance segmentation, and ShelfGaussian employs off-the-shelf vision foundation models with a multi-modal Gaussian transformer for zero-shot semantic occupancy and planning tasks. Beyond Averages [4] rethinks semantic learning in 3DGS by abandoning per-Gaussian seman-

tic blending and instead modeling object-level "bags of embeddings," enabling robust open-vocabulary object retrieval and consistent 2D–3D semantic propagation.

2.3.2 3D Scene Editing

Scene editing is one of the most common downstream tasks in 3D scene applications, as it enables flexible manipulation of 3D assets to meet the growing demand for personalized customization [366]. Compared with other 3D representations, the explicit nature of 3DGS makes it a strong candidate for scene editing. First, each Gaussian acts as an independent primitive carrying explicit semantic information, allowing editing operations to be directly localized to regions of interest. Second, its efficient rasterization supports fast optimization and background inpainting pipelines during and after editing.

In recent years, many exciting on 3DGS-based editing frameworks are proposed and vigorously promoted the development of this field. 3DGS-Drag [53] and Drag Your Gaussian [205] introduce drag-based geometric editing frameworks for 3DGS, where user-defined control points and score distillation guidance enable intuitive, geometry-consistent and multi-view coherent 3D manipulations beyond pure text-driven texture editing. EditSplat [123] and InterGSEdit [291] focus on improving multi-view consistency in text-guided 3DGS editing, leveraging multi-view fusion, geometry-consistent attention priors, and adaptive attention optimization to reduce local artifacts and enable more controllable, high-fidelity edits. Localized Gaussian Splatting Editing [302] proposes an illumination-aware pipeline that integrates anchor-view diffusion inpainting and depth-guided SDS optimization to achieve locally precise edits with globally consistent lighting. TexGS-VolVis [254] extends 3DGS to textured Gaussian primitives for expressive volume visualization, enabling geometry-consistent style transfer and fine-grained partial scene editing via text- and image-driven guidance. Instruct-4DGS [119] addresses efficient dynamic scene editing by decomposing 4D Gaussian representations into static and dynamic components, performing edits only on static Gaussians and refining them via score distillation to achieve scalable temporal editing. AG^{2}aussian [288] introduces an anchor-graph structured Gaussian representation to enable clean instance-level Gaussian selection, facilitating semantic-aware querying, object removal, and physics-consistent editing.

2.4 Method-Oriented Related Works

While the previous sections reviews 3DGS from a task-oriented perspective, this section is organized to support the subsequent methodology chapters of this thesis. Therefore, we revisit the literature from a method-oriented perspective, focusing on the core technical components that are directly relevant to our proposed approaches.

2.4.1 3D Gaussian Splatting for Specular Materials

3D Gaussian Splatting [110] represents scenes using optimized 3D Gaussians, achieving efficient high-quality rendering through rasterization. However, the original formulation struggles with specular reflections due to its reliance on spherical harmonics for view-dependent appearance modeling.

Recent work has explored various strategies to enhance specular material representation. Appearance enhancement approaches modify the underlying appearance model: GaussianShader [105] incorporates specialized shading functions, while Spec-GS [322] uses anisotropic spherical Gaussian fields instead of spherical harmonics. Geometry-integrated approaches incorporate explicit surface modeling: SpectroMotion [54] combines physically-based rendering with deformation fields for dynamic scenes, and GeoSplatting [327] grounds Gaussians on mesh geometry for material decomposition.

Despite these advances, existing methods either lack geometric supervision (appearance-based) or require pre-computed geometry (geometry-aware) [103], leading to inconsistent normal estimation for specular rendering.

2.4.2 Normal Estimation and Geometric Supervision

Accurate surface normals are essential for modeling specular reflections, as they directly determine the direction of reflected light and the spatial coherence of high-frequency appearance [228, 231]. While classic 3DGS methods lack explicit normal supervision, recent works have demonstrated that introducing normal constraints significantly improves reconstruction quality in reflective regions. GausSurf [276] employs geometry guidance from multi-view consistency in texture-rich areas and normal priors in textureless regions for high-quality surface reconstruction. Similarly, DN-Splatter [266] leverages per-pixel depth estimates to enforce geometric constraints, achieving improved photorealism and geometric fidelity. These approaches highlight the importance of geometric supervision in enhancing 3D Gaussian representations.

2.4.3 Relighting and Multi-illumination Synthesis

Relighting plays an important role in inverse rendering by exposing the surface to diverse lighting conditions, which helps to disambiguate geometry and reflectance [142]. This approach proves particularly crucial for specular materials, where view-dependent appearance variations under uniform lighting often lead to ambiguous surface reconstructions. Although relighting has been widely used in relightable appearance synthesis and BRDF estimation, its potential as a geometric augmentation mechanism for enhancing 3D reconstruction quality remains largely unexplored within the 3D Gaussian Splatting framework.

Recent advances in neural relighting have demonstrated impressive capabilities in indoor scene manipulation [230]. ScribbleLight [37] enables single-image indoor relighting through Albedo-conditioned Stable Image Diffusion, while LumiNet [304] leverages latent intrinsics and diffusion models for comprehensive indoor scene relighting. SpotLight [57] introduces shadow-guided object relighting via diffusion. These works primarily focus on achieving photorealistic relighting effects for visual applications.

2.4.4 Novel View Synthesis with Diffusion

For diffusion-based novel view synthesis, the conditioning strategy differs from the standard text-driven setting commonly adopted in image generation models. Rather than using text prompts, the diffusion process is guided by a conditioning signal constructed from a reference image and the target camera pose [125, 279]. This design explicitly incorporates geometric viewpoint information, allowing the model to generate images that are consistent with the desired viewing direction.

Accordingly, the reverse diffusion process can be written as

$$\mathbf{z}_{t-1} = p\big(\mathbf{z}_{t-1} \mid \mathbf{z}_t,\, \epsilon_\theta\big(\mathbf{z}_t, t, E(\mathbf{I}^0, \mathbf{v})\big)\big)\,, \tag{2.10}$$

where $\mathbf{z}_t$ and $\mathbf{z}_{t-1}$ denote the latent variables at diffusion steps t and $t-1$, respectively. Here, $\mathbf{I}^0$ represents the input reference image, and $\mathbf{v}$ denotes the target camera pose to be synthesized. The encoder $E(\cdot)$ maps the reference image and the desired viewpoint into a conditioning embedding, which is then provided to the denoising network ϵ_θ. This conditioning enables the diffusion model to progressively refine the latent representation toward a novel view that preserves the appearance of the input image while respecting the specified camera geometry.

2.4.5 3D Occupancy Prediction

3D occupancy is gaining attention as a viable alternative to LiDAR perception [349]. Most previous works [94, 19, 140, 355] utilize 3D Occupancy Ground Truth for supervision, which is challenging to annotate. With the widespread adoption of Neural Radiance Fields (NeRF) [213, 314], some methods [192, 349, 359, 93, 13, 161] have attempted to use 2D depth and semantic labels for training. However, using continuous implicit neural fields to predict occupancy probabilities and semantic information often leads to high memory cost [312]. Recently, GaussianFromer [95] leverages sparse Gaussian points as a means of reducing GPU consumption to describe 3D scenes while GaussianOcc [61] utilizes a 6D pose network to eliminate the reliance on ground truth poses, but both of them suffers from a significant drop in overall segmentation accuracy.

2.4.6 World Model in Autonomous Driving

World models [71, 275] are often used for future frame prediction and to assist robots in making decisions [247]. As end-to-end autonomous driving [84], [102] is gradually evolving, world models are also applied for predicting future scenarios and decisions making [79]. Unlike traditional autonomous driving approaches [199], [14], the world model approaches integrate perception, prediction and decision making. Many current approaches perform fusion of camera-LiDAR data and input into world model, which is used to forecast [66], [78] and make motion planning [97]. Among them, OccWorld [361] proposes to utilize 3D occupancy as world model's input. However, OccWorld is less effective at utilizing pure 2D input and struggles to accurately predict future scenes due to information loss during the encoding process.

2.4.7 3D Representations for Complex Scenes

Neural Radiance Field (NeRF) [178], based on implicit representation and volumetric rendering, has been a representative work in the field of 3D reconstruction in recent years. It has been widely used for 3D reconstruction [248, 362, 277, 33, 188, 149], AI generation [175, 90, 24], and 3D editing [342, 121, 74]. However, NeRF-based models require dense and continuous sampling in 3D space for optimization. When dealing with complex scenes like ScanNet [41] or ScanNet++ [330] (each scene with hundreds or even thousands of images), the relatively long training time, high computational demand, and substantial GPU memory requirements reduce user friendliness and make interactive-rate scene editing challenging. Recently, 3D Gaussian Splatting (3D-GS)

[110] has become the leading 3D representation technique, praised for its quick training time and high-quality real-time rendering. Similar to NeRF, beside 3D reconstruction purposes [311, 169, 279, 85, 369], 3D-GS is also being widely adopted for 3D generation [252, 332, 331, 363] and editing [31, 278, 232, 21, 82, 77, 25].

2.4.8 3D Scene Editing

Editing NeRF is inherently challenging due to the complex interplay between shape and appearance [31]. However, the availability to individually edit each Gaussian in 3D-GS provides significant flexibility for scene editing, particularly in indoor environments with intricate layouts. Existing 3D-GS editing methods fall into two main categories. The first relies on 2D diffusion priors or large language models (LLMs) [31, 278, 267, 27, 301, 351], enabling text-driven editing pipelines but often limited in complex scenes by diffusion model capabilities. The second category directly edits 3D scenes, bypassing diffusion models by using 2D masks generated by models like Segment Anything Model (SAM) [115] and Ground SAM [216] for 2D-3D semantic projection. For instance, Gaussian Grouping [328] enhances precision by tracking objects across frames, while SAGA [21] and SAGD [82] assign 3D semantic features through 2D mask projections. FlashSplat [232] simplifies this by treating 2D mask lifting as a linear programming problem, enabling single-step editing. However, these methods lack prompt-based control, requiring manual scene edits, which limits usability.

Chapter 3

NeRFBK Dataset Construction

3.1 Introduction

Generating high-quality 3D models is a central objective of many studies in computer vision and photogrammetry. Nowadays, 3D reconstructions are widely applied across various sectors, including quality control, 3D monitoring, inspections, robotics, virtual and augmented reality, and medical imaging. Over the past decades, improvements in image-based 3D reconstruction algorithms have made it increasingly important to evaluate and compare their performance. In particular, NeRF- and 3D Gaussian Splatting (3DGS)-based algorithms have attracted significant attention from the research community due to their potential in high-fidelity 3D reconstruction.

A benchmark dataset is essential for evaluating and comparing the performance of sensors, platforms, or processing algorithms against reliable ground truth (GT). However, obtaining diverse, high-quality datasets is challenging due to the associated costs, time, and the need for precise annotations. Benchmark datasets should cover various scales, environments, and surface characteristics, and be accompanied by accurate ground truth data.

Since the introduction of vanilla NeRF [178] and 3DGS [110], these view-synthesis-based 3D reconstruction methods, which represent a scene as a radiance field, have sparked a major revolution in the field. Unlike many other neural 3D representations [270, 98], NeRF and 3DGS are self-supervised and can learn a scene from a set of multi-view images and camera poses without requiring 3D or depth supervision. These models have found broad applications in robotics, autonomous navigation, virtual and augmented reality, and industrial inspection [62].

Several studies [329, 213, 315] suggest that NeRF- and GS-based methods have

strong potential, often outperforming traditional image-based techniques such as Multi-View Stereo (MVS) [226, 272, 283, 239] and RGB-D-based methods [263, 262, 147], especially when reconstructing texture-less, metallic, highly reflective, or transparent objects, thanks to the view-dependent nature of the models.

Accurate development, evaluation, and comparison of NeRF- and 3DGS-based methods require high-quality datasets with precise ground truth. However, gathering both real and synthetic data covering diverse surfaces, scales, shapes, and material properties is challenging. To address this, the NeRFBK dataset was introduced, providing a comprehensive collection of multi-scale indoor and outdoor datasets with high-resolution images and videos, camera parameters (intrinsic and extrinsic), and ground truth. NeRFBK includes real and synthetic images spanning various surfaces and sizes, covering three main domains: industry, cultural heritage, and geospatial applications.

The NeRFBK dataset enables systematic evaluation of NeRF- and 3DGS-based algorithms across diverse reconstruction challenges, supporting advancements in 3D reconstruction methods. Its utility is demonstrated through experimental results conducted on the shared data, highlighting the dataset's value for benchmarking and method development.

3.2 The Construction of NeRFBK

The NeRFBK datasets are composed of both real and synthetic scenes (refer to Figure 3.1). For the real-world datasets, high-resolution images are captured in diverse scenarios under varying lighting conditions, using different cameras, scales, and ground sample distances (GSD), together with corresponding 3D ground-truth (GT) data. For the synthetic datasets, images and GT are generated in *Blender* by modeling objects with different shapes and sizes and defining various camera trajectories to simulate image acquisition.

As summarized in Table 1, NeRFBK consists of several categories of datasets, including:

- **Industrial**: Two real metallic objects and two synthetic datasets are provided (see Table 3.2). These objects exhibit complex geometries, poor texture, and reflective surfaces, posing significant challenges for both traditional and learning-based 3D reconstruction methods.
- **Transparent and reflective**: Three real-world and one synthetic texture-less object are included (see Table 3.1). The main challenge in these datasets lies in

image matching, as object appearance strongly depends on shape, background, and illumination. Refraction and specular reflections are prevalent, with light traveling through or being reflected by the surface.

- **Heritage**: Ten different cultural heritage scenarios are available (see Tables 3.3, 3.4, 3.5), including the Temple of Baalshamin in Syria, which was tragically destroyed in 2015. The images for this site were collected from the REKREI online repository [271].

- **Aerial**: Two aerial datasets are provided (see Tables 3.6), one acquired from a UAV flight and another captured using an oblique aerial camera over the city of Dortmund [187].

Each dataset includes image sets, original videos (for selected scenes), and ground-truth point clouds or mesh models. In Chapter 4, a subset of these datasets is used to evaluate NeRF- and GS-based reconstruction methods, with both qualitative and quantitative performance analyses reported.

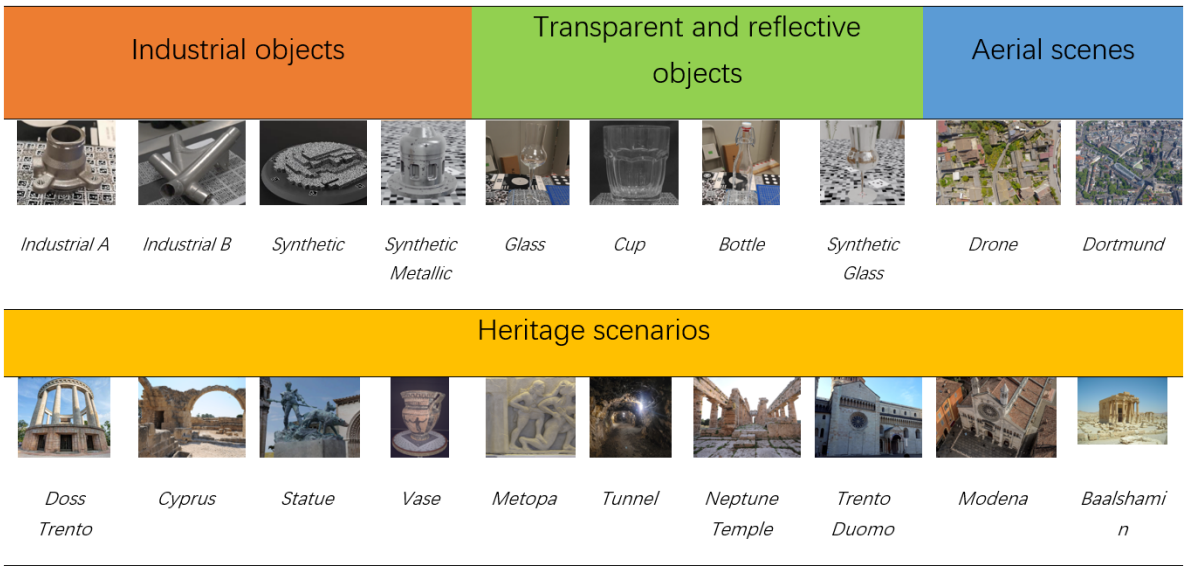


Figure 3.1: Visual summary of the NeRFBK datasets. Data and more info available at https://github.com/3DOM-FBK/NeRFBK

Dataset	Numb. Images	Camera Type & Image Size	Approx. Size (cm)	Description	Video	Synthetic	Ground Truth
Glass	552	Huawei P20 Pro 1920×1080 px	5×5×25	Transparent Highly refractive	Yes	No	
Cup	287	Huawei P20 Pro 1080×1920 px	8×8×10	Complex shape Highly refractive	Yes	No	
Bottle	300	Huawei P20 Pro 1080×1920 px	6×6×30	Complex shape Highly refractive	Yes	No	
Synthetic_Glass	300	Virtual pinhole 1920×1080 px	10×10×22	Transparent Highly refractive	Yes	Yes	

Table 3.1: Transparent and Reflective Objects of NeRFBK

Dataset	Numb. Images	Camera Type & Image Size	Approx. Size (cm)	Description	Video	Synthetic	Ground Truth
Industrial_A	295	Huawei P20 Pro 1080×1920 px	5×5×4	Textureless, Small and Complex, Reflective	Yes	No	
Industrial_B	220	Huawei P20 Pro 1080×1920 px	15×12×4	Textureless, Complex, Reflective	Yes	No	
Synthetic	373	Virtual pinhole 1920×1080 px	11×11×2	Well-textured, Sharp edges	Yes	Yes	
Synthetic_Metallic	300	Virtual pinhole 1080×1920 px	16×16×13	Textureless, Complex, Reflective	Yes	Yes	

Table 3.2: Industrial Objects of NeRFBK

Dataset	Numb. Images	Camera Type & Image Size	Approx. Size (cm)	Description	Synthetic	Ground Truth
Vase	50	Google Pixel2 4024×3016 px	40×40×30	Indoor	No	
Metopa	106	Nikon COOLPIX P90 4000×3000 px	100×80×10	Indoor, museal, detailed relief	No	
Tunnel	4353	RealSense D455 640×480 px	8000×300×200	Underground, Mobile robot based, High frame-rate	No	

Table 3.3: Heritage (Indoor and Underground) of NeRFBK

Dataset	Numb. Images	Camera Type & Image Size	Approx. Size (cm)	Description	Synthetic	Ground Truth
Doss Trento	761	Huawei P20 Pro 1920×1080 px	2500×2500×1500	Outdoor large scale	No	Unavailable
Cyprus Monument	178	Samsung S6 3840×2160 px	1500×1500×500	Outdoor large scale, video	No	
Statue	100	Ideal reflex camera 6016×4016 px	200×100×500	Outdoor large scale, Two scales	Yes	
Neptune Temple	680+214	Nikon D3X 6048×4032 px Canon EOS 550D 5184×3456 px	2500×1500×7000	Terrestrial + UAV images, Outdoor large scale	No	

Table 3.4: Heritage (Outdoor) (a) of NeRFBK

Dataset	Numb. Images	Camera Type & Image Size	Approx. Size (cm)	Description	Synthetic	Ground Truth
Trento Duomo	565	Samsung S6 5312×2988 px	8000×8000	Outdoor large scale, Terrestrial	No	
Modena Cathedral	132+58	Nikon D750 6016×4016 px Canon EOS 600D 5184×3456 px	2500×2500	Terrestrial + UAV images, Large scale architecture	No	
Baalshamin Temple	47	Multiple cameras From 600×399 px to 4288×2848 px	500×1500	Lost object, Sub-optimal baselines, Unordered touristic images, Multiple resolutions	No	Unavailable

Table 3.5: Heritage (Outdoor) (b) of NeRFBK

Dataset	Numb. Images	Camera Type & Image Size	Approx. Size (cm)	Description	Ground Truth
Drone	224	Sony 7952×5304 px	City scale	Outdoor large scale, Built-up and vegetated areas	
Dortmund	59 (from 5 cameras)	IGI Multi-sensor, 5 views 6132×8176 px at Nadir, 50 mm 8176×6132 px at Oblique, 80 mm	City scale	Outdoor large scale, Built-up and vegetated areas	

Table 3.6: Aerial Scenes of NeRFBK

3.3 Comparison of 3D Reconstruction Datasets

As of early 2026, existing datasets for image-based 3D reconstruction purposes, including neural field representations, often fall short in terms of scale, scene diversity, and availability of high-quality annotations, as summarized in Table 3.7. Early datasets such as DTU [101] and NeRF [178] provide only a limited number of scenes and images, mostly captured under controlled conditions, while larger-scale datasets like Scannet [41] and Tanks and Temples [116] offer extensive real-world images but often lack diverse viewpoints or complete ground truth. Datasets such as BlendedMVS [324] and UrbanScene3D [145] focus on diversity, but their ground truth coverage is partial, and lighting variations are limited. OMMO [168] provides a large number of images across multiple scenes with varying lighting, but it is restricted to specific urban scenarios. Mill 19 [265] offers high-quality captures with complete ground truth for a few scenes, yet the dataset is extremely small in scale.

In contrast, our NeRFBK addresses these limitations by providing a balanced combination of scene diversity, high-resolution captures, multiple lighting conditions, and high quality ground truths. This makes it uniquely suited for evaluating both geometric reconstruction accuracy and photorealistic novel view synthesis. Compared with existing datasets, NeRFBK not only scales up the number of scenes and images but also systematically integrates real and synthetic scenarios, providing a comprehensive benchmark for modern 3D reconstruction methods.

3.4 Summary and Relation to Research Question

This chapter addresses the first part of **Research Question 1** by introducing the NeRFBK dataset as a comprehensive benchmark for evaluating neural radiance field representations. By providing multi-scale scenes, diverse surface properties, and accurate ground truth across real and synthetic settings, the dataset establishes a reliable foundation for systematic comparison between NeRF-, 3DGS-, and traditional photogrammetry-based methods. This contribution enables controlled and reproducible analysis of reconstruction performance, which is essential for identifying the strengths and limitations of different 3D representation paradigms.

Dataset	Scenes	Number of Images	Diverse Scenes	Available Camera Poses	Changing Lighting Conditions	Real	Ground Truth
DTU	124	ca. 4,000	No	Yes	Yes	Yes	No
NeRF	18	ca. 3,500	No	Yes	No	Yes	No
Scannet	1,500	ca. 2.5M	No	Yes	No	Yes	Yes
BlendedMVS	28	ca. 5,000	Yes	Yes	No	No	Yes
UrbanScene3D	16	ca. 10,000	Yes	No	No	Yes	Partly
Relighting NeRF	20	ca. 40,000	Yes	Yes	Yes	Yes	No
OMMO	33	ca. 15,000	Yes	Yes	Yes	Yes	No
Mill 19	2	ca. 3,500	Yes	Yes	No	Yes	No
Tanks and Temples	6	ca. 88,000	No	Yes	No	Yes	Partly
NeRFBK	20	ca. 10,000	Yes	Yes	Yes	Yes	Yes

Table 3.7: Comparison among NeRFBK with other 3D reconstruction datasets

Chapter 4

Evaluation of Learning-based Methods

4.1 Introduction

Image-based 3D reconstruction plays a pivotal role in both industrial and urban applications. Traditional methods such as photogrammetry and photometric stereo have long been employed for industrial inspection [89, 109], quality control [83], and reverse engineering [195] due to their precision [132], portability [1], flexible [174], and cost-effectiveness [76]. However, these methods struggle with non-collaborative surfaces, including shiny metals, glass, or transparent materials, where featureless regions, refraction, and specular reflections lead to noisy or incomplete reconstructions [3, 72, 181, 62]. Achieving high-quality, consistent reconstruction in such scenarios remains an open challenge.

Recently, learning-based methods such as Neural Radiance Fields (NeRF) [178] and 3D Gaussian Splatting (3DGS) [110] have emerged as powerful alternatives. NeRF represents a scene as a continuous volumetric function that maps 3D coordinates and viewing directions to color and density, enabling photorealistic novel view synthesis. 3DGS, in contrast, explicitly encodes the scene as a set of 3D Gaussian ellipsoids, which can be efficiently rendered via rasterization, offering fast convergence, high-quality reconstruction, and GPU-friendly performance [312]. Both approaches have demonstrated impressive results in indoor, object-level, and small-scale outdoor scenarios [239, 334].

Extending these methods to industrial and urban-scale environments introduces several challenges. For industrial objects, especially non-collaborative surfaces, sparse textures and reflective or transparent materials still limit reconstruction accuracy. Existing studies often consider only single-object scenarios with limited evaluation metrics, leaving the reliability and practical applicability of learning-based methods largely

unexplored. In urban-scale reconstruction, additional difficulties arise from memory consumption, computational overhead, and geometric fidelity. The unordered Gaussian representation in 3DGS can degrade novel-view quality, while mesh-coupled or opacity-based enhancements may increase memory usage and reduce fine-detail capture [67, 341]. Sparse, occluded, or highly dynamic regions in street-level and aerial captures further complicate fair evaluation, and large-scale training pipelines remain prohibitive for time-critical or resource-constrained applications [144, 159].

In summary, these challenges highlight the need for systematic evaluation and analysis of learning-based 3D reconstruction methods across both industrial and urban-scale settings. Understanding the strengths and limitations among different learning-based methods for texture-less, reflective, or large-scale scenes is essential for guiding future research, improving reconstruction robustness, and enabling practical deployment in real-world scenarios.

To facilitate such evaluation, we introduced the NeRFBK [314] dataset in Chapter 3. Building upon this dataset, the present chapter conducts a comprehensive evaluation of learning-based reconstruction methods under challenging conditions.

In this chapter, we review and systematically evaluate a diverse set of 3D reconstruction approaches for both industrial object inspection and urban scene modeling. The evaluated methods were selected based on the representative techniques available at the time this study was originally conducted. For industrial objects, we focus on challenging surfaces, including texture-less, reflective, shiny, and transparent materials (Figure 4.1), some of which are included in the NeRFBK [314] dataset. For urban scenarios, we extend the evaluation to large-scale outdoor reconstructions, considering challenges such as sparse observations, occlusions, and high memory demands inherent to city-level 3D modeling.

We use standard photogrammetric metrics and additional measures appropriate for large-scale scenes to quantify reconstruction completeness, geometric accuracy, and precision. This enables a comprehensive comparison across methods and scales. The main objectives of this chapter, based on [316, 315], are:

(i) To report and categorize the currently available learning-based methods for reconstructing industrial objects and urban environments, especially those with challenging surface or scene characteristics;

(ii) To objectively evaluate the performance of NeRF, MVS, MDE, 3DGS, and generative AI methods across both small-scale industrial objects (Section 4.2) and large-scale urban scenes (Section 4.3);



(a) (b) (c) (d)

Figure 4.1: Examples of objects featuring non-collaborative surfaces: reflective and texture-less (a,b) or transparent and refractive (c,d).

(iii) To provide a clear summary of the advantages, limitations, and potential applications of these methods for industrial 3D metrology and urban-scale reconstruction tasks.

4.2 Analysis and Evaluation of Non-collaborative Surfaces

In this section, we present a critical evaluation of the learning-based 3D reconstruction methods by objectively measuring their capability in dealing with non-collaborative surfaces. The evaluated methods were selected from representative approaches available at the time when the experiments were originally conducted. To accomplish this, some industrial objects of different sizes and surface characteristics are considered, including texture-less, metallic, translucent and transparent (Table 4.1). The proposed evaluation strategy and metrics aim to support researchers in understanding the strengths and limitations of each approach.

4.2.1 Proposed Assessment Methodology

The assessment procedure is shown in Figure 4.2. The 3D reconstructions from three different datasets are compared with the available ground truth (GT) data in order to derive quantitative metrics.

All collected images or videos required camera poses in order to generate a 3D reconstruction, either with MVS, GS, or NeRF-based methods. Starting from the available unoriented images, camera poses were retrieved, for all datasets using COLMAP. Then, the selected learning-based methods were applied to generate dense 3D geometries. For

	Industrial_A	**Synthetic Metallic**	**Synthetic Glass**
			
No. of images, resolution	290 images 1280 × 720 px	300 images 1080 × 1920 px	300 images 1080 × 1920 px
Ground truth (GT)	Triangulation-based laser scanning	Synthetic data	Synthetic data
Characteristics	Texture-less; small and complex	Texture-less; complex; reflective	Transparent; highly refractive

Table 4.1: A summary of the objects used in our analyses and available in the NeRFBK dataset.

evaluating the MDE method, a subset of images (4–16) for each object taken from different viewpoints was selected and used as the input into the candidate networks. Then, the predicted depth maps and known camera parameters were used to create 3D point clouds. Finally, all produced point clouds were co-registered and rescaled with re-spect to the available ground truth (GT) data in Cloud Compare using an Iterative Closest Point algorithm [9], and a quality evaluation was performed. To provide an unbiased evaluation of geometric accuracy, different well-establish photogrammetric criteria were applied [76, 116], including best plane fitting, cloud-to-cloud comparison, profiling, accuracy, and completeness.

For some selected objects, profiling was additionally conducted by extracting a cross-section from the 3D data to highlight complex geometric details of the reconstructed surface. An inspection of profiles allowed us to evaluate the performance of a method in preserving geometric details, such as edges and corners, and avoiding smoothing effects. Cloud-to-cloud (C2C) comparisons refer to the measurement of the Euclidean distance between corresponding closest points in the evaluated and GT point cloud.

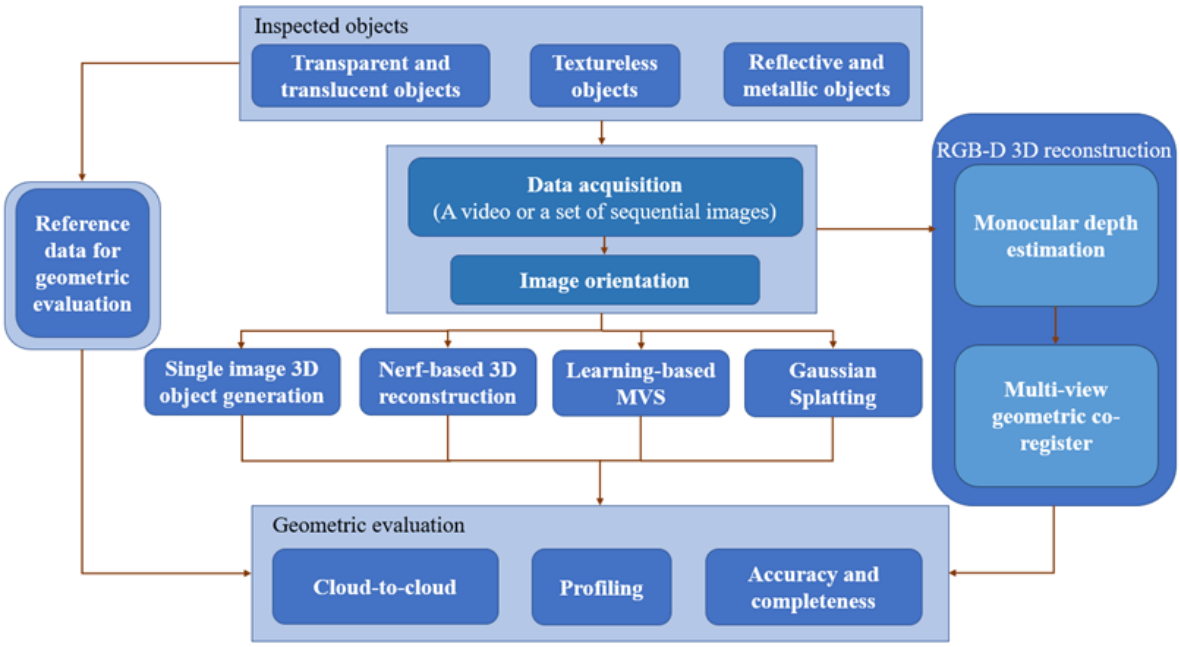


Figure 4.2: An overview of the proposed procedure to assess the performance of non-collaborative surfaces using different 3D reconstruction methods.

4.2.2 Testing Objects and Methods

To achieve the study objectives, two datasets of industrial objects and one of a transparent glass available in the NeRFBK [314] dataset were used (Table 4.1). They feature objects of distinctive characteristics and surface types, with different lighting conditions, materials, sensor types, scales, and resolutions.

We considered four categories (NeRF, Gaussian Splatting, learning-based MVS, and MDE) and a total of 26 methods (Table 4.2) chosen among open-source codes available in Github repositories. The NeRF methods were integrated in NeRFStudio [249] and SDFStudio [339], whereas all other tools are available from Github repositories.

All experiments were performed with a single NVIDIA GeForce A40, A6000, or RTX3080TI GPU. To enable efficient comparison, only the top-ranked methods in each category are afterwords presented

4.2.3 Results from Multi-View Image Sequences

Industrial_A Object

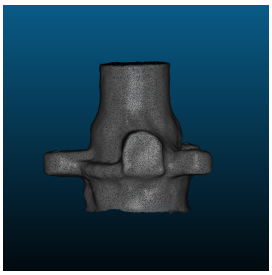
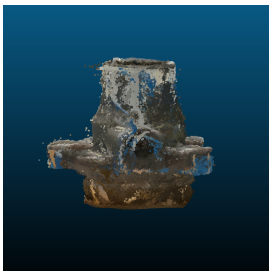
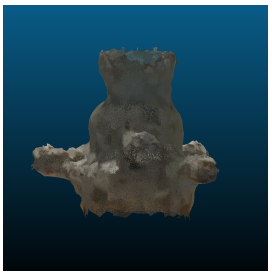
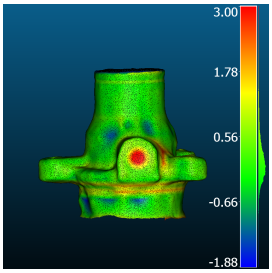
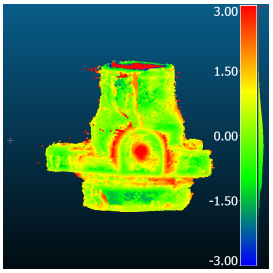
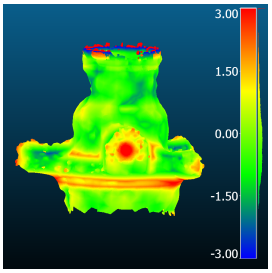
Out of the 22 considered methods based on multi-view image sequences (Table 4.2), all methods, with some exceptions with five NeRF-based approaches (Mono-Unisurf, NeuS-

Category	Methods
NeRF	Instant-NGP [183], Mono-NeuS [339], MonoSDF [340], Mono-UniSurf [339], Nerfacto [352], Neuralangelo [138], NeuS [281], Nerfacto (w/ depth) [352], Nerfacto (w/o depth) [352], UniSurf [190], VolSDF [325]
Gaussian Splatting	FSGS [369], GaussianShader [105], Gaussian Splatting [110], Scaffold-GS [169]
Learning-based MVS	DI-MVS [104], ET-MVSNet [158], GBi-Net [176], GeoMVSNet [357], KD-MVS [51], MVSFormer [20], MVStudio [185], TransMVSNet [50]
MDE	ZoeDepth [10], MiDaS [12], Depth Anything [320]

Table 4.2: The evaluated methods for the 3D reconstruction of texture-less, metallic, translucent, and transparent objects.

facto, Unisurf, VolSDF, and Instant-NGP that failed to reconstruct at least 60% of the object), successfully reconstructed the object's geometry. The comparison results are re-ported in Table 3 for the best approach of each evaluated 3D reconstruction method. Neu-ralangelo achieved the best results among the NeRF methods, and MVSFromer ranked second among all methods. Their RMSDs were 0.57 mm and 0.85 mm, respectively. From a visual inspection (Table 4.3) it is evident that the surface reconstructed by Neuralangelo is highly consistent with the GT, as evidenced by the predominance of dark green points in the comparison. In contrast, MVSFormer shows extensive red regions, indicating signifi-cant errors, along with some scattered noise near the surface. Gaussian Splatting, the best Splatting method, exceeded 1 mm in the RMSD, achieving worse results than half of the tested NeRF and MVS methods due to its noticeably uneven surface, which significantly deviated from the GT.

Figure 4.3 shows the accuracy and completeness of all tested methods on the Industrial_A dataset. Similarly, for the convenience of comparison, we consolidated the top-performing methods in terms of accuracy and completeness in each category for com-prehensive analysis. In terms of accuracy, Neuralangelo outperforms the other methods, followed by MVStudio. KD-MVS is the winner in terms of completeness, achieving higher than 85% and 90% re-call within 1 mm and 2 mm, respectively, far ahead of other methods. However, KD-MVS is inferior to other methods in completeness, since its output point cloud is high density, but also contains a substantial noise.

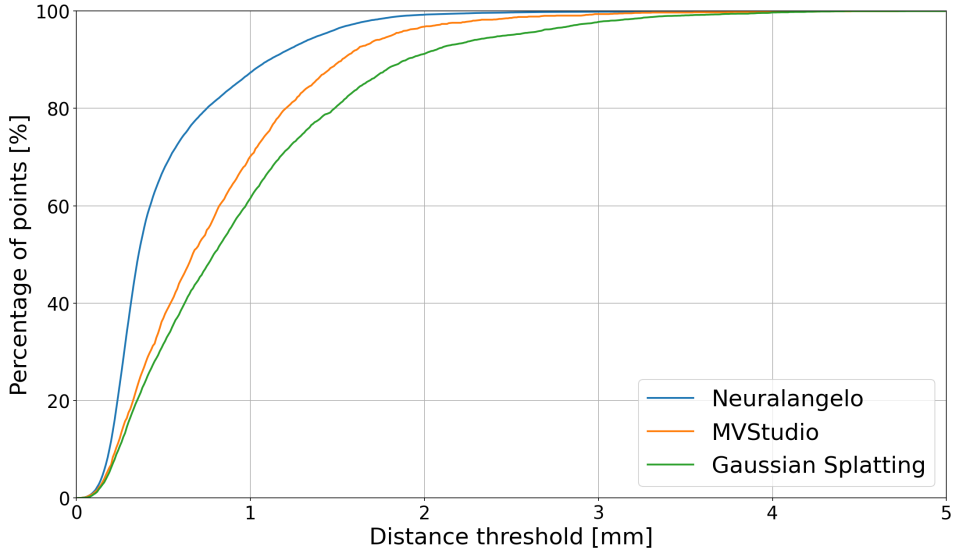
	NeRF	**Learning-Based MVS**	**Gaussian Splatting**
3D Geometry			
Comparison result [mm]			

Method		Neuralangelo	MVSFormer	Gaussian Splatting
Metric [mm]	RMSE	0.57	0.85	1.11
	MAE	0.43	0.69	0.89
	STD	0.37	0.49	0.66
	Mean_E	0.13	-0.19	0.14

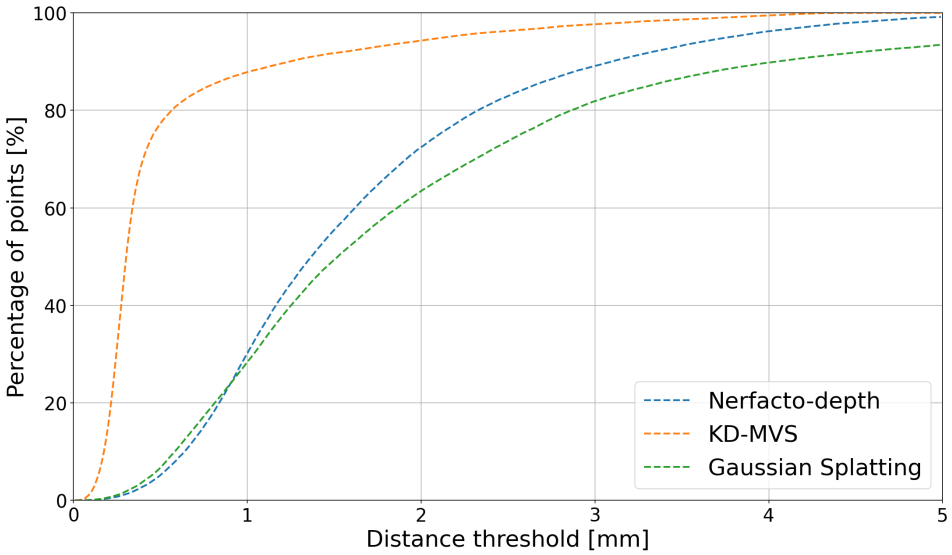
Table 4.3: Cloud-to-mesh comparison of the best-performing methods from each category on the Industrial_A object.

Metallic Object

The Synthetic Metallic dataset contains 300 images (1080 × 1920 pixels) of a synthetic, reflective, texture-less, and metallic object created in Blender. We applied the same processing steps as reported in Section 4.2.3 to Industrial_A object. All approaches successfully reconstructed the geometry of the object, and the comparison results are shown in Table 4.4. Among the MVS methods, GBi-Net achieved the best results, with KD-MVS and MVSFormer ranking second and third. The RMSD of GBi-Net was 0.7 mm, almost twice as low as the first-ranked NeRF-based method (Mono-Neus). FSGS achieved a relatively inferior quality in the experiment, with the RMSD equal to 1.92 mm. Overall, the worst performing method was Instant-NGP, which generated a very noisy result, with an RMSD of 5.8 mm. The top-performing results of the accuracy and completeness of each category applied to the Synthetic Metallic dataset are presented

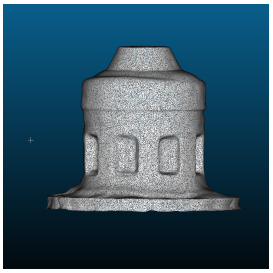
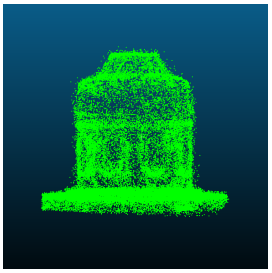
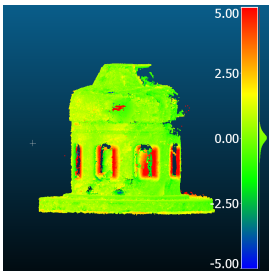
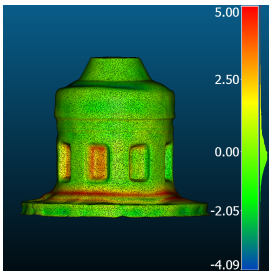
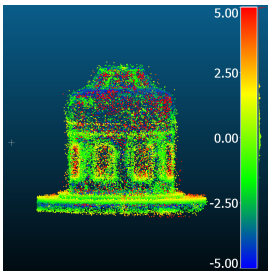


(a) Accuracy of Industrial_A



(b) Completeness of Industrial_A

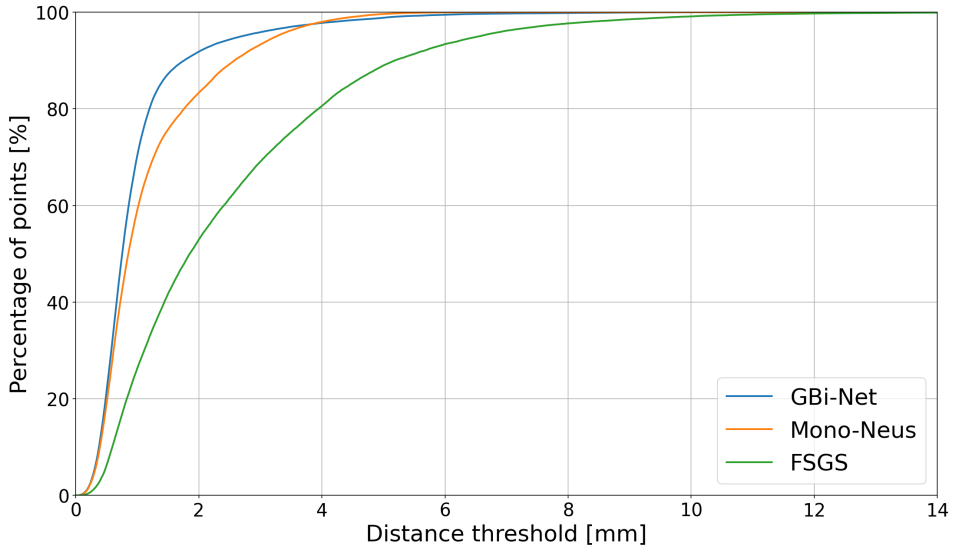
Figure 4.3: The best accuracy (a) and completeness (b) for all tested methods using the Industrial_A object.

	Learning-Based MVS	**NeRF**	**Gaussian Splatting**
3D Geometry			
Comparison result [mm]			

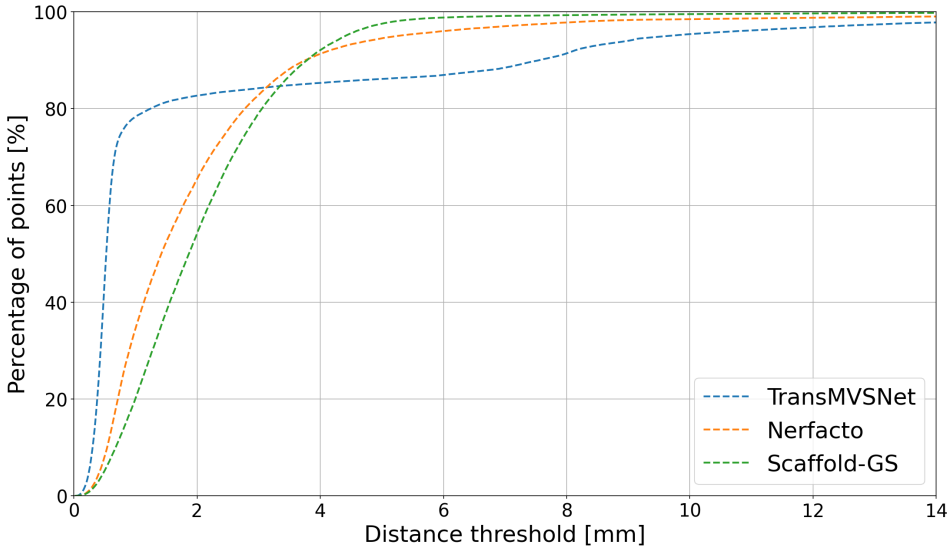
Method		GBi-Net	Mono-Neus	FSGS
Metric [mm]	RMSE	0.70	1.38	1.92
	MAE	0.61	1.10	1.49
	STD	0.35	0.83	1.22
	Mean_E	0.00	0.32	-0.41

Table 4.4: Cloud-to-mesh comparison of the best-performing methods from each category on the Metallic object.

in Figure 4.4. GBi-Net achieved the highest accuracy among all methods, followed by Mono-Neus and FSGS. This can be attributed to GBi-Net; as an MVS-based method, it benefits from depth supervision, allowing it to capture more precise geometric details when reconstructing objects with regular sizes. While minor estimation errors remain in the concave regions at the object's center due to depth estimation inaccuracies, its overall reconstruction accuracy was significantly higher than Mono-Neus (which exhibited geometric distortion at the object's base and chassis connection) and FSGS (which produced excessive noise points above and on top of the object). About completeness, TransMVSNet demonstrated outstanding results within a 1 mm distance threshold but ultimately was surpassed by Nerfacto and Scaffold-GS as the distance threshold exceeded 3 mm. Similarly to the findings in Industrial_A, the Gaussian Splatting



(a) Accuracy of Metallic Object

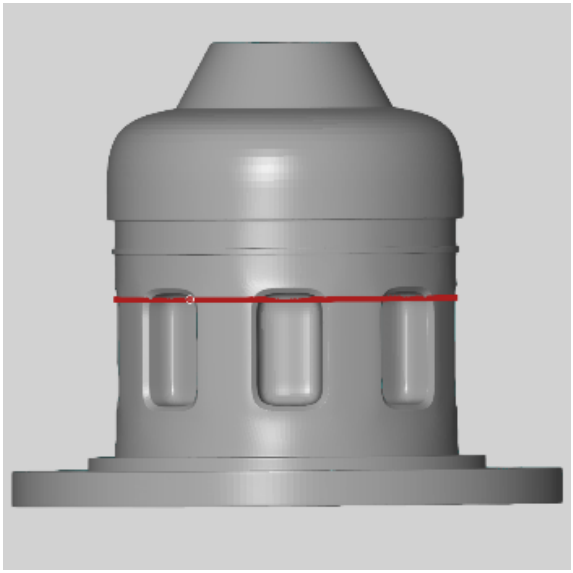


(b) Completeness of Metallic Object

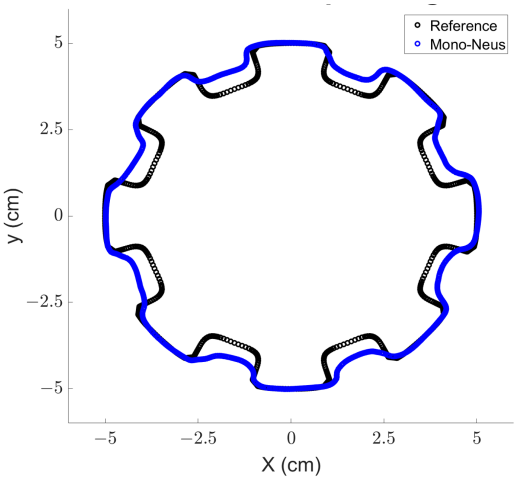
Figure 4.4: The best accuracy (a) and completeness (b) for all tested methods using the Metallic object.

methods exhibited lower accuracy and completeness in a low threshold range (<=2 mm) compared to NeRF and MVS. This discrepancy could be attributed to a limited number of initialized Gaussian points sampled from highly reflective surfaces.

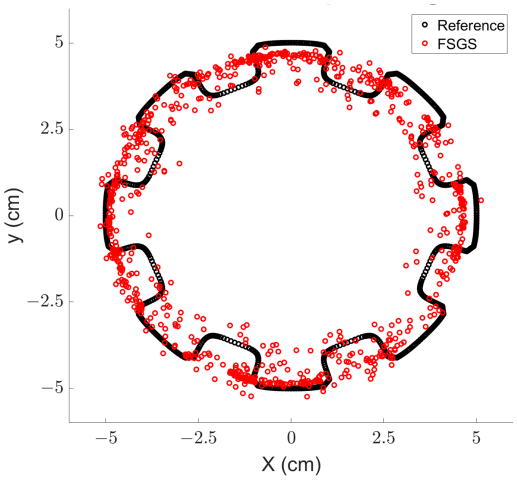
In addition to cloud to mesh comparisons and accuracy and completeness analyses, some cross-sections were extracted from the best-reconstructed geometries in each category (Mono-Neus for NeRF, FSGS for Gaussian Splatting, and GBi-Net for MVS)



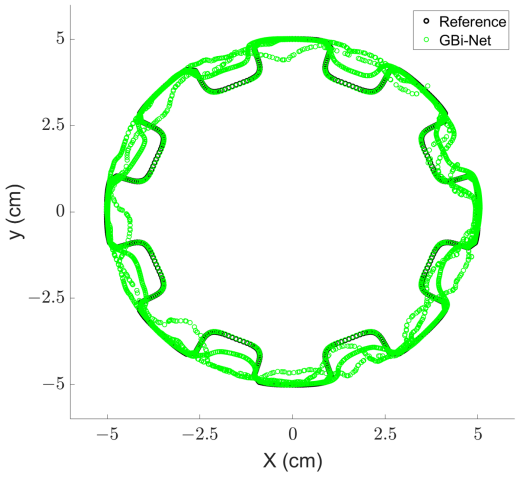
(a) The location of the profile



(b) Cross-section profile for NeRF



(c) Cross-section profile for Gaussian Splatting



(d) Cross-section profile for MVS

Figure 4.5: Cross-section profiles on the Synthetic_Metallic object reconstructed with the best method from each category: (a) The location of the profile on the Synthetic_Metallic object, (b) Mono-NeuS for NeRF, (c) FSGS for Gaussian Splatting, and (d) GBi-Net for MVS.

to check whether small geometric details could be reconstructed. The section location and the profiles are shown in Figure 4.5. The MVS profile (green point) resulted in a better match with ground truth than the others (red, purple, and blue lines) due to the depth-based nature of MVS. However, the result was full of noise on the surface. The result of NeRF (blue point) was slightly inferior to MVS. The geometric features

of the cavities of the object were not accurately reconstructed. Compared to the other methods, FSGS (red points) had trouble in the geometric reconstruction of both the convex and concave parts.

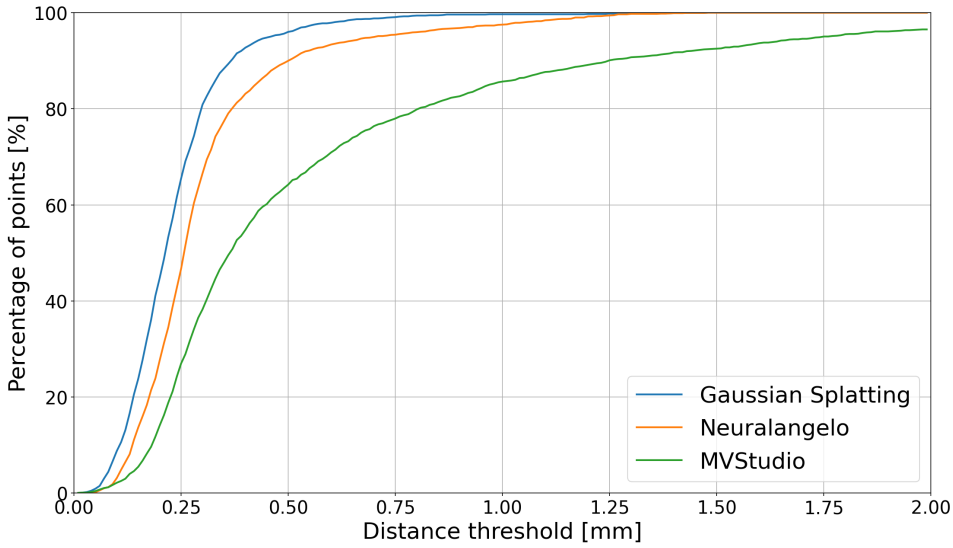
Transparent Object

To evaluate the ability to deal with transparent and refractive surfaces, the methods reported in Section 4.2.1 were tested with the Synthetic_Glass dataset. Table 4.5 reports the results for the top-performing methods of each category: Gaussian Splatting achieved the best results, with 1.54 mm in RMSD, 1.22 mm in MAE, and 0.93 mm in STD. Neuralangelo and MVStudio ranked second and third with RMSD of 2.29 mm and 3.14 mm, respectively. These results also demonstrated that Gaussian explicit representation is more effective for transparent objects, whereas MVS-based methods struggle due to their reliance on depth estimation. This phenomenon was also corroborated by our MDE experiments (Section 4.2.4), where all MDE methods failed to reconstruct the geometry of the Synthetic_Glass object.

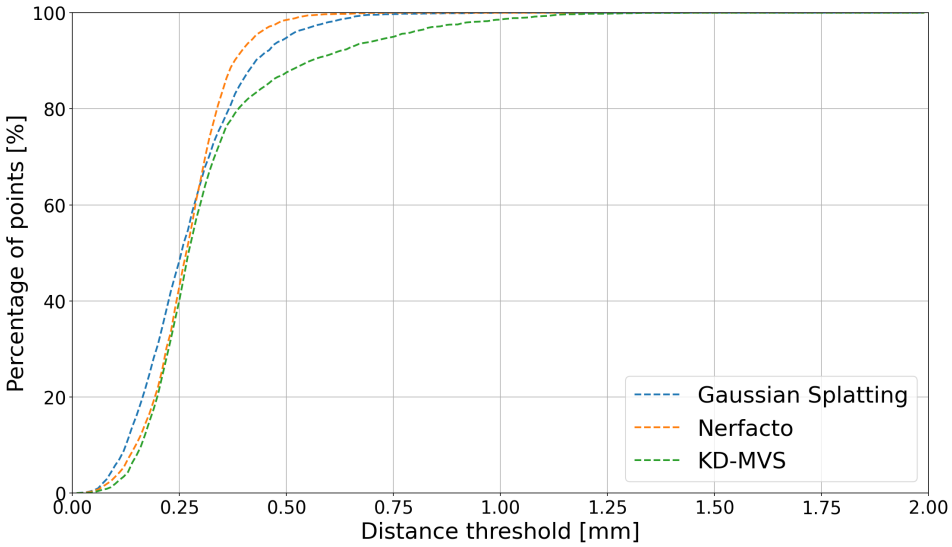
For MVS methods, GeoMVSNet and GBi-Net failed to reconstruct the geometry, whereas MVStudio consistently outperformed the other methods across all metrics, despite the fact that its completeness was generally low. Accuracy and completeness results are presented in Figure 4.6. In contrast to the performance observed on industrial objects, Gaussian Splatting surpassed all competitors in accuracy, while in terms of completeness, the best method resulted from Nerfacto. It is worth mentioning that Gaussian Splatting not only outperformed all other methods in terms of accuracy but also ranked second in cross-category comparison of completeness, highlighting its notable ability to handle transparent objects. In contrast, even if Nerfacto performed worse in accuracy, it achieved better results than other methods in completeness, indicating the capability to reconstruct transparent surfaces in a denser but noisier way.

4.2.4 Results from Monocular Depth Estimation

Different from other 3D reconstruction methods, an object's 3D shape can also be ob-tained from a single RGB image by applying the MDE method. Assuming to have different viewpoints of the object, MDE outputs are inferred depth maps per viewpoint; hence, point clouds can then be generated, e.g., using the Open3D library, and finally all clouds can be co-registered to create a unique 3D reconstruction of the object. As Zoedepth provides metric depth estimates, while other methods infer relative depths, we rescaled the estimated depths of the MiDaS and Depth Anything methods by em-

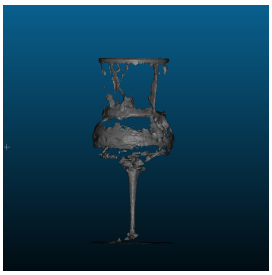
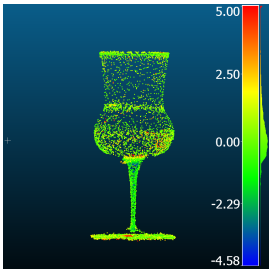
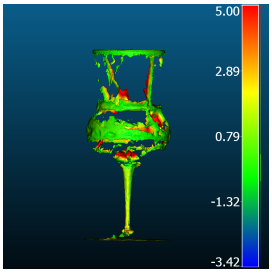
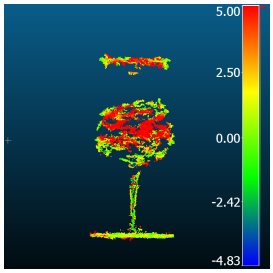


(a) Accuracy of Transparent_Glass Object



(b) Completeness of Transparent_Glass Object

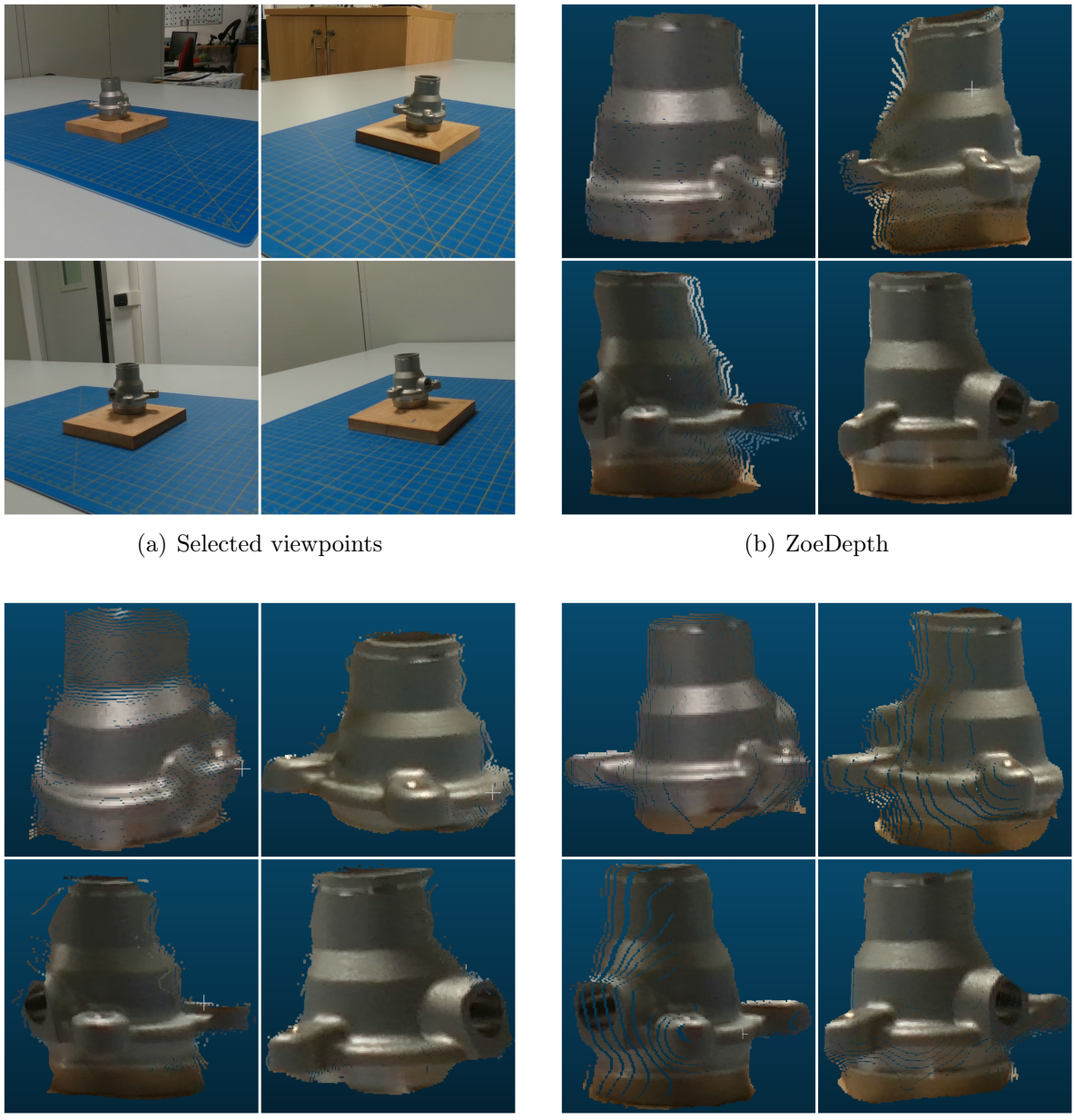
Figure 4.6: The best accuracy (a) and completeness (b) methods among all tested approaches for the transparent glass object.

	Gaussian Splatting	**NeRF**	**Learning-Based MVS**
3D Geometry			
Comparison result [mm]			

Method		Gaussian Splatting	Neuralangelo	MVStudio
Metric [mm]	RMSE	1.54	2.29	3.14
	MAE	1.22	1.72	1.69
	STD	0.93	1.51	1.43
	Mean_E	0.44	1.19	0.93

Table 4.5: Metrics for the cloud-to-mesh comparisons of the best-performing methods from each category applied to the transparent glass object

ploying linear regression to establish a linear relationship between pixel values in the depth image and their respec-tive distances in meters. The reported results refer to Industrial_A and Synthetic_Metallic objects, as no MDE methods could derive successful results on the transparent glass. The results for Industrial_A are presented in Figure 4.7 and Table 4.6. Clearly, ZoeDepth achieved better outcomes compared to MiDaS and Depth Anything. Although MiDaS attained the lowest error in View_01, its notably high standard deviation indicates relative algorithmic instability. However, it is worth noting that the accuracy of these results may be questionable due to significant geometric distortion observed in some of the generated point clouds. This distortion could potentially lead to inaccurate geometric matching when applying ICP for point cloud co-registration.



(a) Selected viewpoints

(b) ZoeDepth

(c) MiDaS

(d) Depth Anything

Figure 4.7: Visualization of MDE results for Industrial_A object.

Method	**ZoeDepth**			**MiDaS**			**Depth Anything**		
Metric [mm]	RMSD	MAE	STD	RMSD	MAE	STD	RMSD	MAE	STD
View_01	1.67	1.22	1.14	0.89	0.68	0.58	1.95	1.30	1.44
View_02	1.41	1.09	0.90	1.46	1.12	0.94	1.67	1.22	1.14
View_03	1.19	1.01	0.86	2.08	1.56	1.38	1.28	0.99	0.82
View_04	1.35	1.11	0.76	1.77	1.16	1.32	1.17	0.88	0.77
Average	1.41	1.11	0.92	1.55	1.13	1.06	1.52	1.10	1.04
Standard deviation	0.20	0.09	0.16	0.51	0.36	0.37	0.36	0.20	0.31

Table 4.6: Metrics [mm] for the cloud-to-mesh comparisons of the tested MDE methods applied to the Industrial_A object.

We also present qualitative results on the Synthetic_Metallic dataset, as shown in Figure 4.8. Among the evaluated methods, Depth Anything achieves the most accurate 3D geometric reconstruction. In contrast, MiDaS produces reconstructions with pronounced geometric distortions, while ZoeDepth exhibits a clear inward concavity along the object's longitudinal geometry.

4.2.5 Discussion

Table 4.7 summarizes the experimental results for each considered method and object. Synthetic_Glass presented the most significant challenge, failing for eight NeRF-based methods, two learning-based MVS, and all MDE methods. Industrial_A also posed some challenges, and six NeRF-based methods failed to produce correct 3D data. Conversely, Synthetic Metallic proved to be the easiest object, with all methods being able to reconstruct its geometry.

The results reported in Section 4.2 indicate that none of the AI-based methods always outperformed the others in all tested scenarios, although in each category, certain approaches emerge as clear winners. For NeRF-based methods, Mono-Neus stands out as the undisputed champion, achieving first place in both Industrial_A and Synthetic_Metallic objects despite its shortcomings in dealing with transparent objects. However, determining a definitive winner in learning-based MVS proves challenging as no single approach demonstrates outstanding performance across multiple object types.



(a) Selected viewpoints



(b) ZoeDepth



(c) MiDaS



(d) Depth Anything

Figure 4.8: Visualization of MDE results for Synthetic_Metallic object.

Category	Method	Synthetic_Metallic	Industrial_A	Synthetic_Glass
NeRF	Instant-NGP	✓	✗	✗
	Mono-NeuS	✓	✓	✗
	MonoSDF	✓	✗	✗
	Mono-UniSurf	✓	✗	✗
	Nerfacto (w/ depth)	–	✓	✓
	Nerfacto (w/o depth)	✓	✓	✓
	Neuralangelo	✓	✓	✓
	NeuS	✓	✓	✗
	NeuS-Facto	✓	✗	✗
	UniSurf	✓	✗	✗
	VolSDF	✓	✗	✗
Gaussian Splatting	FSGS	✓	✓	✓
	GaussianShader	✓	✓	✓
	Gaussian Splatting	✓	✓	✓
	Scaffold-GS	✓	✓	✓
MVS	DI-MVS	✓	✓	✓
	ET-MVSNet	✓	✓	✓
	GBi-Net	✓	✓	✗
	GeoMVSNet	✓	✓	✗
	KD-MVS	✓	✓	✓
	MVSFormer	✓	✓	✓
	MVStudio	✓	✓	✓
	TransMVSNet	✓	✓	✓
MDE	Depth Anything	✓	✓	✗
	MiDaS	✓	✓	✗
	ZoeDepth	✓	✓	✗

Table 4.7: A summary of the evaluated methods for the three different non-collaborative industrial objects.

In terms of accuracy, no approach can be unequivocally deemed as the winner due to the challenge of achieving consistently stable and excelling performance across all test scenarios. Nerfacto emerged as a frontrunner in completeness within the tested scenarios, even though it had shortcomings in accuracy performance. Although learning-based MVS achieved better results in reconstructing geometric details on the surface, it tended to introduce more noise into the results compared to NeRF-based methods and Gaussian Splatting. However, these methods typically produce results with less noticeable noise on the surface of the generated object. In general, NeRF-based methods outperform other approaches in objects with small sizes and asymmetric surfaces. This suggests their suitability for application in micro-industrial object inspection, particularly those susceptible to noise interference. Learning-based MVS can output a dense and very accurate result for medium or large objects with intricate surface structures, which will not have a significant impact on the final metrics due to partial noise. The MVS methods are well suited for applications that require dense point clouds but do not demand real-time processing and visualization. They are particularly appropriate for applications in aerospace component engineering, heritage restoration, and city-level scene reconstruction. However, learning-based MVS is sensitive to transparent and highly refractive surfaces, leading to substantial errors in depth estimation and resulting in a proliferation of noise points on glass objects. For scenes involving transparent surfaces, Gaussian Splatting proved to be more suitable due to its ability to mitigate such effects. Additionally, Gaussian Splatting's explicit representation enables manual control over the number of generated points. This flexibility makes it particularly advantageous for applications requiring real-time performance, fast transmission, and storage efficiency, such as autonomous driving, AR/VR, and rapid geometry editing.

4.2.6 Conclusions of Non-collaborative Surfaces Evaluation

This section investigated the feasibility of employing learning-based methods to handle non-collaborative surfaces and presented a comprehensive metrological analysis using diverse types of learning-based 3D reconstruction methods. Quantitative and visual comparison tests among NeRF, MVS, Gaussian Splatting and MDE were performed to understand the advantages and disadvantages when dealing with non-collaborative surfaces. The research employed complex, texture-less, metallic, reflective, and transparent objects, coming from both real and virtual scenarios. The quality of the generated 3D data was assessed using various evaluation approaches and metrics, including noise level, geometric accuracy and completeness. We verified the possibility of utilizing multi-view

datasets but also just one or a few images for the 3D reconstruction of non-collaborative surfaces, even in the absence of prior camera information, paving the way for future research in this area.

4.3 Analysis and Evaluation on Urban Scenes

4.3.1 Proposed Assessment Methodology

We design our evaluation pipeline (shown in Figure 4.9) following the methodology presented in [315]. Since 3DGS requires camera poses as input, we adopt the exterior parameters provided by each dataset, which were estimated using COLMAP. The selected Gaussian Splatting methods are then applied to generate dense 3D geometries. All generated point clouds are co-registered to the ground-truth (GT) reference using an Iterative Closest Point (ICP) algorithm in CloudCompare. Finally, qualitative and quantitative evaluations are performed based on established photogrammetric criteria to ensure unbiased assessment. Comprehensive results will be presented in the following experimental section of this chapter.

4.3.2 Implementation Details and Data Pre-processing

We implement all evaluations in PyTorch and CUDA, conducting experiments on a single NVIDIA Tesla H100 GPU. For the five 3DGS approaches examined in this work, CityGaussian and BlockGaussian adopt a block-based training strategy, while Octree-GS, Gaussian Opacity Fields (GOF), and Wild-Gaussians employ a holistic training approach. Due to the high GPU memory consumption of block-based training, we downsample the image resolution to prevent out-of-memory errors and apply depth regularization (e.g., Depth Anything V2 [321]) to refine the sparse point clouds generated in COLMAP. This improves the quality of the initial models and compensates for the potential restruction loss caused by image downsampling. The remaining methods do not require special pre-processing; their main distinctions lie in training and Gaussian initialization. CityGaussian and BlockGaussian train in partitions for efficiency, Octree-GS enables adaptive multi-level optimization, and GOF and Wild-Gaussians enhance training with opacity learning and robustness to sparse data.

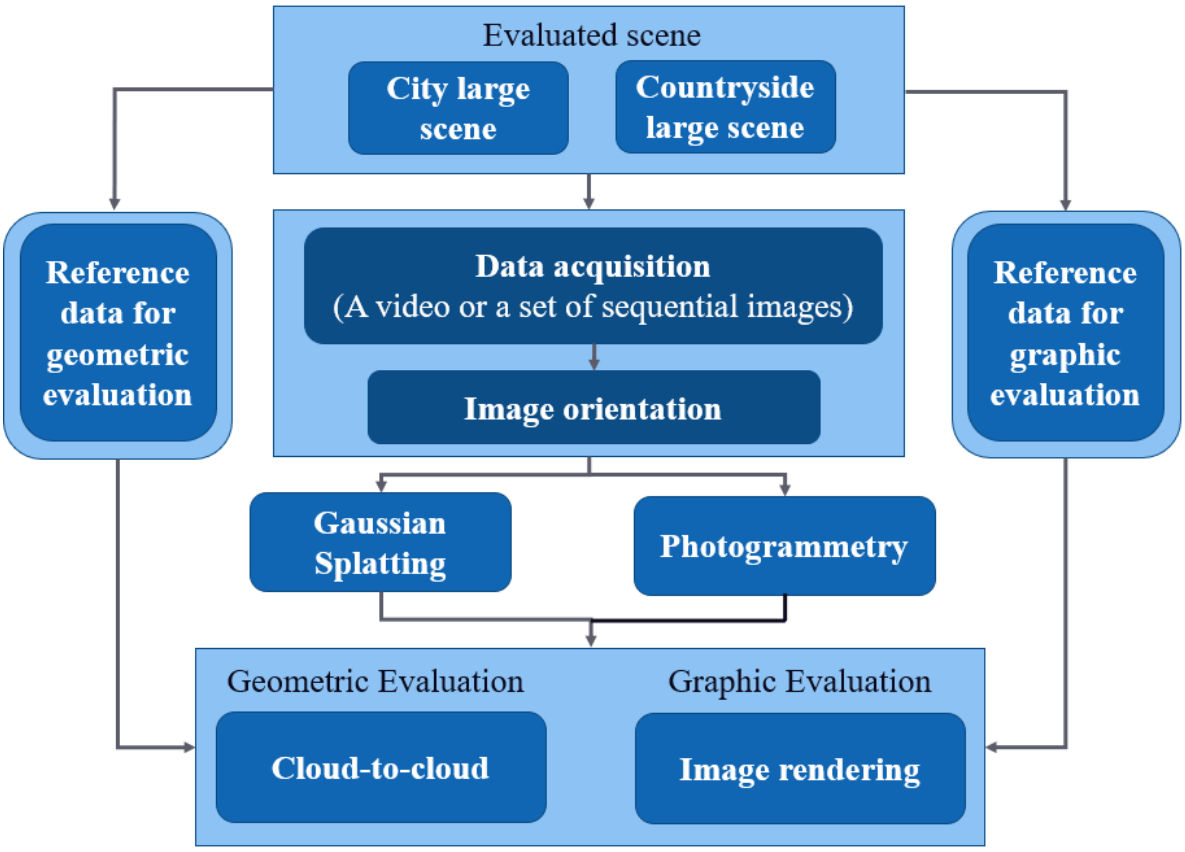


Figure 4.9: Overall framework for assessing the performance of various Gaussian Splatting methods in urban scene reconstruction

Dataset	Sample Image	Numb.of images	Camera type & image size	Description / GDS	Ground Truth (GT)
UseGeo - Dataset_1[1]		224	Sony ILCE-7RM3, 7952x5304 px	High-resolution built-up and vegetated areas (1.7 cm GSD)	Helicopter-based Laser Scanner
Dortmund[2]		59	IGI PentaCam, 6132x8176 px (N), 8176x6132 px (O)	Outdoor large scale, built-up and vegetated areas (12 cm GSD)	Airborne-based Laser Scanner
MatrixCity Block_2[3]		535	Virtual camera 1920x1080 px (synthetic images)	Urban large-scale, metropolitan areas (6.5cm GSD)	MVS / Photogrammetry

[1] https://github.com/3DOM-FBK/usegeo
[2] https://github.com/3DOM-FBK/NeRFBK
[3] https://city-super.github.io/matrixcity/

Table 4.8: Overview of datasets used for urban scene evaluation.

4.3.3 Datasets

To evaluate Gaussian-based reconstruction methods in urban scenarios, we select three representative urban datasets from NeRFBK [314] and MatrixCity [136] (see Table 4.8), covering both real and synthetic acquisitions. **UseGeo** consists of a single high-resolution scene captured with a Sony ILCE-7RM3 camera at 7952×5304 pixels, depicting built-up and vegetated areas with a fine ground sample distance (GSD) of 1.7 cm. Ground truth geometry was acquired using a helicopter-mounted laser scanner, providing precise 3D references for evaluation. **Dortmund** includes 59 large-scale outdoor scenes captured with an IGI PentaCam at 6132×8176 px for nadir and 8176×6132 px for oblique views, representing urban and vegetated regions at 12 cm GSD, with ground truth obtained via airborne laser scanning. While **MatrixCity Block 2** is a synthetic dataset of 535 images generated with a virtual camera at 1920×1080 px, modeling metropolitan areas at 6.5 cm GSD; depth maps were generated using Multi-View Stereo and photogrammetric simulation. Collectively, these datasets form a comprehensive benchmark for assessing 3D reconstruction performance across varying scene scales, camera setups, and urban environments, enabling robust evaluation of geometric accuracy, completeness, and visual fidelity in large-scale scene reconstruction.

4.3.4 Metrics

To quantitatively assess the performance of different methods, consistent with standard practices in the industrial 3D reconstruction community, we employ statistical metrics for cloud-to-cloud evaluation. These include the mean error (Mean_E, or $\bar{X}$) (Eq. 4.1), standard deviation (STD) (Eq. 4.2), root-mean-square deviation (RMSD) (Eq. 4.3), and mean absolute error (MAE) (Eq. 4.4).

Here, the mean error provides the average deviation between reconstructed and reference surfaces, RMSD captures the overall deviation magnitude, MAE reflects absolute positional differences, and STD evaluates the spread of the reconstruction errors.

$$\bar{X} = \frac{1}{N}\sum_{j=1}^{N} X_j \tag{4.1}$$

$$\text{STD} = \sqrt{\frac{1}{N}\sum_{j=1}^{N}(X_j - \bar{X})^2} \tag{4.2}$$

$$\text{RMSD} = \sqrt{\frac{1}{N}\sum_{j=1}^{N} X_j^2} \tag{4.3}$$

$$\text{MAE} = \frac{1}{N}\sum_{j=1}^{N} |X_j| \tag{4.4}$$

where N is the total number of observed points, and X_j denotes the distance from the j-th reconstructed point to its closest corresponding point on the reference surface.

Additionally, we compute accuracy and completeness, also referred to as precision and recall [316, 191], defined as the fraction of points within a specified distance threshold T_h from the ground truth:

$$\text{Accuracy} = \frac{1}{S}\sum_{i=1}^{S} \mathbf{1}(\text{DisT}_i < T_h) \tag{4.5}$$

$$\text{Completeness} = \frac{1}{T}\sum_{i=1}^{T} \mathbf{1}(\text{DisS}_i < T_h) \tag{4.6}$$

Here, DisT_i is the distance from the i-th reconstructed point to its nearest ground-truth point, DisS_i is the distance from the i-th ground-truth point to its nearest reconstructed point, and S and T denote the total number of reconstructed and ground-truth points, respectively.

To evaluate the rendering quality, we further employ commonly used photometric metrics: SSIM, PSNR, and LPIPS. These are computed as follows:

- PSNR (Peak Signal-to-Noise Ratio):

$$\text{PSNR} = 10\log_{10}\frac{L^2}{\text{MSE}}, \quad \text{MSE} = \frac{1}{N}\sum_{i=1}^{N}(I_i - \hat{I}_i)^2 \tag{4.7}$$

  where I_i and $\hat{I}_i$ are the pixel values of the ground truth and reconstructed image, N is the total number of pixels, and L is the maximum pixel value (e.g., 255).

- SSIM (Structural Similarity Index Measure):

$$\text{SSIM}(I, \hat{I}) = \frac{(2\mu_I\mu_{\hat{I}} + C_1)(2\sigma_{I\hat{I}} + C_2)}{(\mu_I^2 + \mu_{\hat{I}}^2 + C_1)(\sigma_I^2 + \sigma_{\hat{I}}^2 + C_2)} \tag{4.8}$$

  where $\mu_I, \mu_{\hat{I}}$ are the mean pixel values, $\sigma_I^2, \sigma_{\hat{I}}^2$ are variances, $\sigma_{I\hat{I}}$ is the covariance,

and C_1, C_2 are constants to stabilize the division.

- LPIPS (Learned Perceptual Image Patch Similarity):

$$\text{LPIPS}(I, \hat{I}) = \sum_l \frac{1}{H_l W_l} \sum_{h,w} ||w_l \odot (\hat{y}^l_{hw} - y^l_{hw})||_2^2 \tag{4.9}$$

where y^l and $\hat{y}^l$ are feature maps of layer l from a pre-trained network, H_l, W_l are spatial dimensions, and w_l are learned channel weights.

These metrics together provide a comprehensive assessment of both 3D reconstruction and rendering quality.

Dataset	Method	RMSE	MAE	STD	Mean_E
UseGeo	Photogrammetry	0.23	0.18	0.14	0.20
	BlockGaussian	3.98	3.08	3.51	3.27
	Octree-GS	3.39	2.52	2.28	2.69
	CityGaussian	5.46	3.87	3.85	4.42
	GOF	4.40	3.06	3.16	3.30
	Wild-GS	–	–	–	–
Dortmund	Photogrammetry	0.88	0.68	0.55	0.75
	BlockGaussian	3.50	2.71	2.48	2.92
	Octree-GS	1.63	1.07	1.23	1.29
	CityGaussian	1.77	1.11	1.38	1.48
	GOF	3.14	2.13	2.32	2.68
	Wild-GS	6.85	4.97	4.71	5.75
MatrixCity	BlockGaussian	1.68	1.18	1.19	1.38
	Octree-GS	0.73	0.53	0.49	0.61
	CityGaussian	1.80	1.27	1.28	1.41
	GOF	3.43	1.82	2.90	2.79
	Wild-GS	–	–	–	–

Table 4.9: Evaluation metrics [m] across different methods on the UseGeo, Dortmund and MatrixCity datasets. Best results are highlighted in green, worst results in red.

4.3.5 Experiment Results

Urban-scale datasets pose unique challenges for 3D reconstruction, especially due to complex building structures, large variations in scale, and reflective or textureless surfaces. We evaluated five Gaussian Splatting methods: BlockGaussian [298], Octree-GS [215], CityGaussian [159], Gaussian Opacity Fields [341], and Wild-GS [306] on three

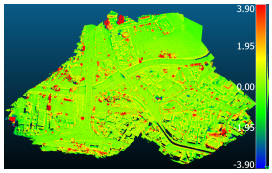
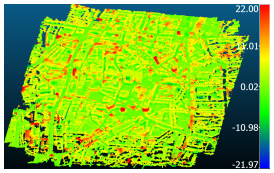
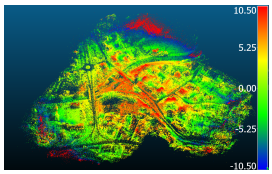
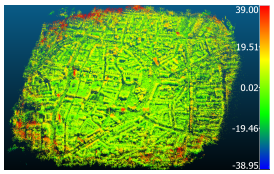
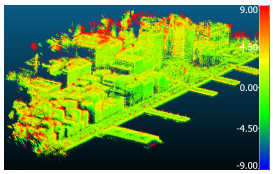
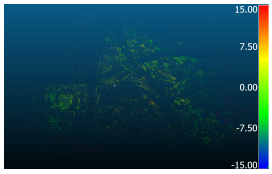
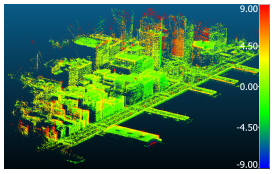
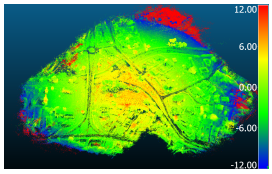
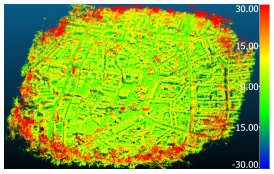
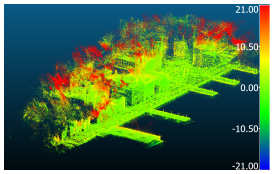
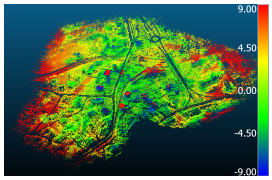
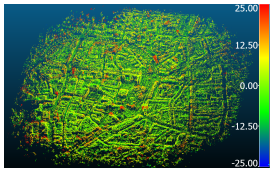
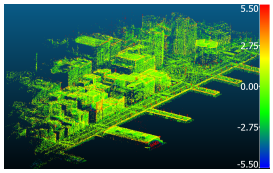
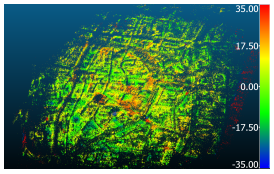
Method	UseGeo	Dortmund	MatrixCity
Photogrammetry			Not available
BlockGaussian [298]			
CityGaussian [159]			
Gaussian Opacity Fields [341]			
Octree-GS [215]			
Wild-GS [306]	✗		✗

Table 4.10: Visual results of the geometry-based comparisons for the evaluated methods across all datasets. The legends in the figures are set to +/- 3σ [unit: m] considering the metrics of each methods.

representative urban datasets, namely Drone, Dortmund, and MatrixCity Block_2. We presents the qualitative results of derived 3D reconstructions across the three datasets in Table 4.8 while the quantitative comparisons of geometric and graphic are reported in Table 4.9.

Qualitative Evaluation

As shown in Table 4.10, we subjectively annotate whether each reconstruction is deemed successful, defined as exhibiting no significant geometric discrepancies with respect to the GT. The qualitative results further indicate that Photogrammetry achieves the most accurate reconstructions, as most regions are correctly recovered (highlighted in green). In contrast, 3DGS methods tend to generate noisy points along scene boundaries (marked in red), which negatively affect their overall reconstruction performance.

Quantitative Evaluation

UseGeo - Dataset It is particularly challenging due to its high-resolution nadir imagery (1.7 cm GSD) covering mixed urban and rural environments. Among all evaluated methods, Photogrammetry achieved the most accurate reconstruction, with RMSE = 0.23 m, MAE = 0.18 m, STD = 0.14 m, and Mean_E = 0.20 m, demonstrating its superior geometric fidelity. Octree-GS ranked second, achieving the best performance among 3DGS methods with RMSE = 3.39 m, MAE = 2.52 m, STD = 2.28 m, and Mean_E = 2.69 m. BlockGaussian and GOF produced acceptable results, whereas CityGaussian exhibited the poorest performance, with RMSE = 5.46 m and Mean_E = 4.42 m. Wild-GS failed to generate valid outputs.

Dortmund It highlights reconstruction performance on dense urban blocks with high structural complexity. Photogrammetry achieved the highest accuracy, with RMSE = 0.88 m, MAE = 0.68 m, STD = 0.55 m, and Mean_E = 0.75 m, demonstrating superior geometric precision and stability across diverse structural patterns. Among 3DGS approaches, Octree-GS again achieved the best results, with RMSE = 1.63 m and MAE = 1.07 m, confirming the effectiveness of its hierarchical representation. CityGaussian reached a competitive RMSE of 1.77 m but showed instability reflected by higher variance. GOF and BlockGaussian achieved moderate accuracy but presented larger errors in regions with intricate geometry, while Wild-GS produced the poorest performance, with RMSE exceeding 6.8 m and Mean_E = 5.75 m. These findings further emphasize traditional photogrammey has better accuracy in urban geometric reconstruction than 3DGS methods.

MatrixCity - Block 2 It focuses on block-level reconstruction within a synthetic city. Octree-GS once more achieved the best results across all metrics with RMSE equal to 0.73m, MAE equal to 0.53m, STD equal to 0.49m, and Mean_E equal to 0.61m, which confirms its robustness across different urban scales. BlockGaussian and CityGaussian provided reasonable reconstructions with RMSE between 1.7m and 1.8m, but with higher noise compared to Octree-GS. GOF performed poorly with RMSE equal to 3.43m and STD equal to 2.9m, and Wild-GS failed to converge.

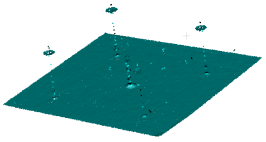
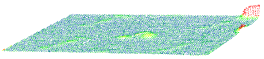
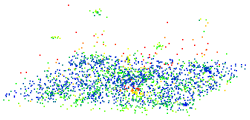
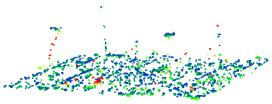
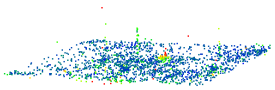
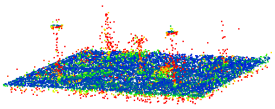
GT		Photogrammetry	BlockGaussian
			
RMSE			1.59
MAE		Failed	1.17
STD			1.07
Octree-GS	**CityGaussian**	**GOF**	**Wild-GS**
			Failed
0.48	0.43	0.70	
0.41	0.34	0.47	Failed
0.26	0.26	0.52	

Table 4.11: Geometric evaluation [m] of fine-structure reconstruction between 3DGS methods and traditional photogrammetry.

Geometric Evaluation of Fine-structures

To evaluate the fine-structure reconstruction capability of GS-based methods compared with traditional photogrammetry, we selected several detailed regions from the Dortmund dataset. As shown in Table 4.11, CityGaussian achieved the best overall performance, with an RMSE of 0.43m, MAE of 0.34m, and STD of 0.26m, demonstrating its strong ability to preserve fine geometric details. Octree-GS followed closely, achieving an RMSE of 0.48, MAE of 0.41m, and STD of 0.26m, indicating consistent accuracy and stability. GOF produced moderate results, as its dense outputs introduced excessive attached noise points. In contrast, Photogrammetry failed to reconstruct those small structures due to the limited GSD of the images.

Qualitative and Quantitative Evaluation of Rendered Images

We qualitatively and quantitatively evaluate the rendering quality of the 3DGS methods across three datasets. As shown in Figure 4.10 and Table 4.12, BlockGaussian consistently achieves superior performance in photometric metrics, obtaining the highest SSIM and PSNR values while maintaining the lowest LPIPS, indicating sharp structural preservation and faithful texture recovery. Octree-GS and CityGaussian perform comparably overall, but their relative strengths vary across datasets. On the UseGeo dataset, Octree-GS slightly outperforms CityGaussian while on MatrixCity Block_2, CityGaussian shows superior metrics. In contrast, GOF exhibits unstable performance across datasets. It tends to generate noisy points near scene boundaries, which can cause severe distortions in certain rendering views, leading to lower photometric scores compared to BlockGaussian and the other two methods. Wild-GS exhibits the weakest performance. Its rendered images show severe RGB mismatches with the GT, along with multiple local distortions. Fine texture and structural details are obviously lost, resulting in significantly lower SSIM and PSNR, as well as higher LPIPS, which further confirms its limited generalization and unsuitability for urban scene reconstruction.



Figure 4.10: Qualitative graphic-based comparison of the evaluated 3DGS methods across MatrixCity (top) and UseGeo (bottom) datasets. Fine structures such as wires and windows are highlighted with close-ups for clearer visibility.

Dataset	Method	SSIM ↑	PSNR ↑	LPIPS ↓
UseGeo	BlockGaussian	**0.8229**	**28.28**	**0.242**
	Octree-GS	0.6820	24.45	0.265
	CityGaussian	0.5154	21.01	0.853
	Gaussian Opacity Fields	0.3693	17.20	0.627
	Wild-GS	–	–	–
Dortmund	BlockGaussian	**0.9626**	**32.90**	**0.049**
	Octree-GS	0.8299	26.56	0.150
	CityGaussian	0.8554	28.10	0.144
	Gaussian Opacity Fields	0.9092	30.29	0.094
	Wild-GS	0.7072	25.08	0.335
MatrixCity	BlockGaussian	**0.9647**	**37.46**	**0.146**
	Octree-GS	0.7374	21.49	0.338
	CityGaussian	0.9007	32.64	0.214
	Gaussian Opacity Fields	0.8134	30.07	0.390
	Wild-GS	–	–	–

Table 4.12: Photometric evaluation metrics on three tested datasets. Higher SSIM/PSNR and lower LPIPS are better.

Quantitative Evaluation of GPU Memory Usage

Table 4.13 summarizes the peak GPU memory usage of all methods during training. On Dortmund, GOF and Wild-GS consume more memory, while BlockGaussian, Octree-GS, and CityGaussian remain more efficient. For UseGeo and MatrixCity, the memory demand increases for all methods due to more training images, with GOF and Wild-GS again showing the highest consumption. Octree-GS achieves notable efficiency through its hierarchical octree structure, which minimizes redundancy. BlockGaussian further reduces memory and computation by partitioning scenes into smaller blocks. CityGaussian saves memory via aggressive Gaussian pruning but sacrifices reconstruction completeness. In contrast, Wild-GS requires substantial memory on large datasets, limiting its scalability. Overall, these results reveal a trade-off between computational efficiency and reconstruction quality. Octree-GS and CityGaussian perform best in global and fine-structure reconstruction, respectively, while BlockGaussian excels in rendering metrics. Considering both geometry and photorealism, Octree-GS emerges as the most balanced method, demonstrating the strength of hierarchical splatting for urban-scale 3D reconstruction.

Method	UseGeo	Dortmund	MatrixCity
BlockGaussian	24108	5626	14520
Octree-GS	**18964**	6043	11403
CityGaussian	23814	**4025**	**6212**
GOF	42096	17727	39746
Wild-GS	32845	8027	25195

Table 4.13: GPU memory usage (MB) for each method during training on the Dortmund and MatrixCity dataset. Since GPU usage is dynamic, we pick the peak values as the result.

4.3.6 Conclusions of Urban Scene Evaluation

This section presented a systematic evaluation of recent 3DGS methods for large-scale urban scene reconstruction, comparing them with traditional photogrammetry and LiDAR data as ground truth. Our study examines 3DGS approaches in terms of scalability, generalization, and efficiency in complex urban environments, where reconstruction accuracy, detail fidelity, and memory usage are critical. Experiments on five 3DGS methods show that BlockGaussian, CityGaussian, GOF, and Octree-GS successfully reconstruct all evaluated datasets, whereas Wild-GS produces valid output only on the Dortmund dataset. In general, no 3DGS method dominates in all aspects of reconstruction and rendering. Traditional photogrammetry achieves the highest accuracy in global geometric reconstruction, but struggles with fine or reflective structures. CityGaussian performs best in fine-structure reconstruction, BlockGaussian achieves best results in image rendering, and Octree-GS emerges as the most balanced method, achieving strong geometric accuracy alongside competitive rendering quality. These results provide valuable insights for researchers by offering guidance on method selection, computational resource allocation, and performance trade-offs across diverse application scenarios. We believe that these findings will facilitate future research on urban-scale 3D reconstruction leveraging 3DGS techniques, ideally integrated with conventional photogrammetric MVS approaches.

4.4 Summary and Relation to Research Question

This chapter completes **Research Question 1** by providing a systematic evaluation of representative learning-based 3D reconstruction methods on the NeRFBK dataset. Through extensive experiments on both non-collaborative objects and large-scale urban scenes, we identify the strengths and limitations of NeRF-, MVS-, MDE-, and 3DGS-based approaches under different reconstruction conditions. In particular, while learning-based methods demonstrate strong performance in appearance modeling and local detail preservation, they still exhibit limitations in geometric consistency, scalability, and robustness under challenging surfaces and large-scale environments. These findings, together with the dataset introduced in Chapter 3, provide a comprehensive answer to RQ1 by establishing both a benchmark and a structured empirical analysis of current reconstruction paradigms.

Chapter 5

Improving Surface Reconstruction of Non-Collaborative Objects using Gaussian Splatting

5.1 Introduction

Reconstructing and rendering high quality 3D scenes from images and videos has long been a central goal in photogrammetry, computer vision and computer graphics [128]. This capability underpins a wide range of applications, including AR and VR, 3D content creation, and embodied intelligence [315].

Traditional 3D representations, such as meshes and point clouds [184, 15, 285], have been widely used for reconstruction. More recently, neural field representations have significantly advanced this area by enabling continuous and differentiable scene modeling. Among them, Neural Radiance Fields [177] and 3D Gaussian Splatting [110] have demonstrated remarkable performance in novel view synthesis and high-fidelity reconstruction.

Despite these advances, reconstructing non-collaborative surfaces remains a fundamental challenge. These surfaces, such as transparent, specular, or texture-less regions, exhibit complex light interactions that violate standard assumptions of multi-view consistency, making it difficult to establish reliable correspondences and recover accurate geometry [213, 169, 327]. In particular, view-dependent effects and insufficient geometric constraints often lead to degraded reconstruction quality under sparse views or uncontrolled illumination.

While NeRF-based approaches model scenes using volumetric rendering, they often

struggle with efficiency and robustness in such challenging scenarios. On the other hand, 3DGS provides an explicit and highly efficient representation, enabling real-time rendering with competitive visual quality. However, the original 3DGS formulation lacks explicit modeling of surface geometry and reflectance properties, making it less effective for handling specular or reflective materials. In particular, the absence of reliable normal supervision and multi-illumination cues limits its ability to capture physically consistent appearance variations.

In this chapter, we present a novel surface reconstruction approach built upon Gaussian Splatting that effectively addresses its limitations in handling specular, glossy, and transparent surfaces. Our method introduces a surface normal supervision mechanism that using the normals maps obtained from multi-illumination relight images to enhance reconstruction quality. Specifically, we first apply a pre-trained relighting network to generate images under diverse illumination conditions which can effectively enhance the detail (geometry and texture) of target object. These relight images are then processed through a surface normal estimation network to extract high-quality normal maps, which are used for geometric supervision during the Gaussians Splatting training pipeline. This comprehensive pipeline enables object detail enhancement the model to better capture complex view-dependent effects and significantly improves reconstruction fidelity for transparent and reflective scenarios.

The key contributions of our approach, presented in [313], are summarized as follows:

- **Enhanced rendering for reflective and transparent objects:** Our method significantly improves the rendering and reconstruction quality of reflective and transparent objects such as metallic surfaces and glass materials, which are poorly handled by compared baselines. This improvement is achieved through the use of surface normal guidance and illumination-consistent supervision.

- **Densification of 3D Gaussians:** Compared to baselines, our approach generates a denser and more accurate point cloud scene representation. This leads to improved surface coverage and enhanced detail preservation, especially beneficial for scenes containing complex geometries and high-frequency regions.

- **Joint integration of relighting and normal supervision:** We combine a relighting network for consistent illumination with predicted surface normals to guide geometry-aware radiance learning, and introduce a specialized loss that balances supervision from real and generated novel views. This combination en-

hances view-dependent effects, improves geometric fidelity, and maintains high photometric realism.

5.2 Methodology

Our method enhances 3D reconstruction of specular objects by combining relighting-based augmentation, robust normal estimation, and normal-supervised optimization within the 3DGS framework. The methodology is structured into four key components: We first establish the overall pipeline and the unified loss formulation that combines photometric and geometric supervision (Section 5.2.1). We then describe our multi-illumination data generation strategy using Neural Gaffer to synthesize diverse lighting conditions from sparse input views (Section 5.2.2). We then describe the robust normal estimation process using DSINE on relight images and the multi-view consistency filtering mechanism (Section 5.2.3). Finally, we introduce our enhanced Specular Gaussian Splatting formulation(Section 5.2.4), which integrates explicit normal supervision with view-dependent specular rendering. Figure 1 illustrates the overall pipeline.

5.2.1 Overall Framework

We model the complete pipeline as a composition of three functional modules (see Figure 5.1): relighting R, normal estimation N, and specular 3D Gaussian Splatting G. Given an input image $I_{\text{in}} \in \mathbb{R}^{H\times W\times 3}$ and associated camera parameters $P = (K, T)$, the final outputs are a rendered image I_{out} and a rendered normal map N_{out}. The entire process is formulated as follows:

$$I_{\text{out}},\, N_{\text{out}} = G(N(R(I_{\text{in}})),\, I_{\text{in}},\, P) \tag{5.1}$$

where:

- $R(I_{\text{in}}) = \{I_l\}_{l=1}^{L}$ synthesizes L relight images under diverse illumination conditions;
- $N(\{I_l\}) = N_{\text{supervised}} \in \mathbb{R}^{H\times W\times 3}$ predicts dense surface normals in the world coordinate frame;
- G takes the estimated normals $N_{\text{supervised}}$, the original image I_{in}, and camera parameters P to optimize and render the final image and normal map.

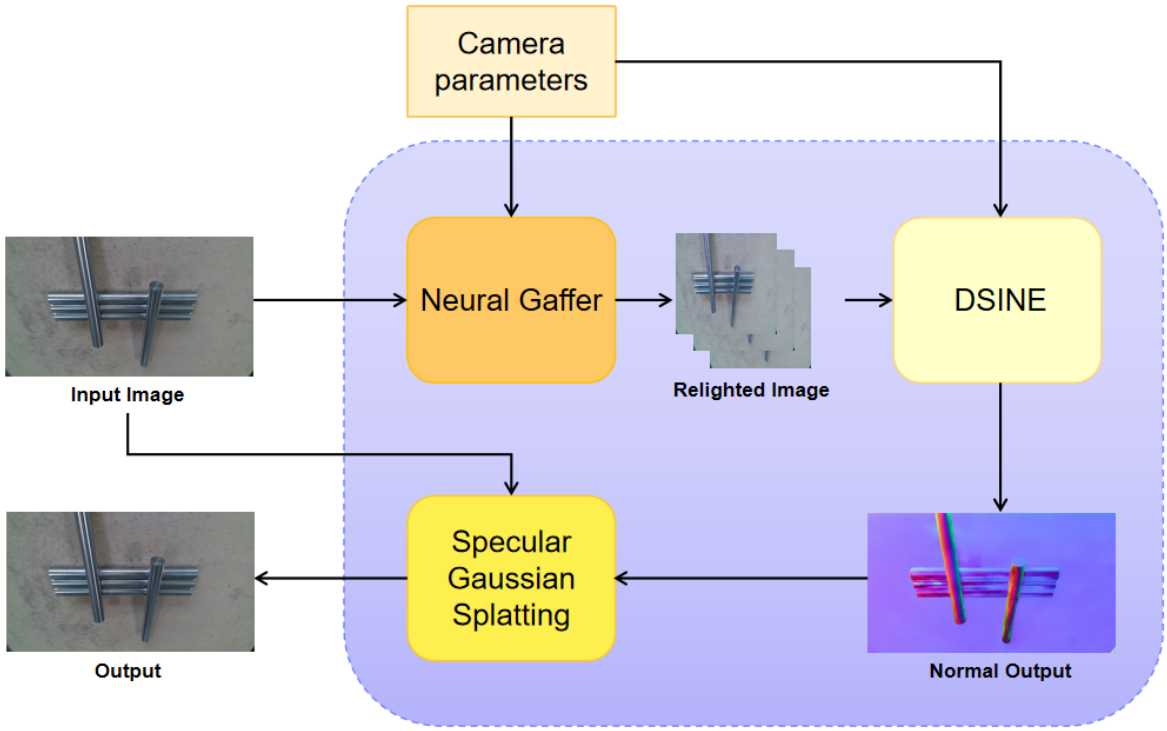


Figure 5.1: Overall pipeline of our approach. Given the input image and corresponding camera parameters, Neural Gaffer synthesizes relight images under various lighting conditions. These are used by DSINE to estimate surface normals, which in turn supervise the Specular Gaussian Splatting module.

The model is trained with a joint objective that combines photometric and normal supervision losses:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{photo}}(I_{\text{out}}, I_{\text{in}}) + \lambda_{\text{normal}} \cdot \mathcal{L}_{\text{normal}}(N_{\text{out}}, N_{\text{supervised}}) \tag{5.2}$$

where λ_{normal} is a weighting coefficient to balance the two terms.

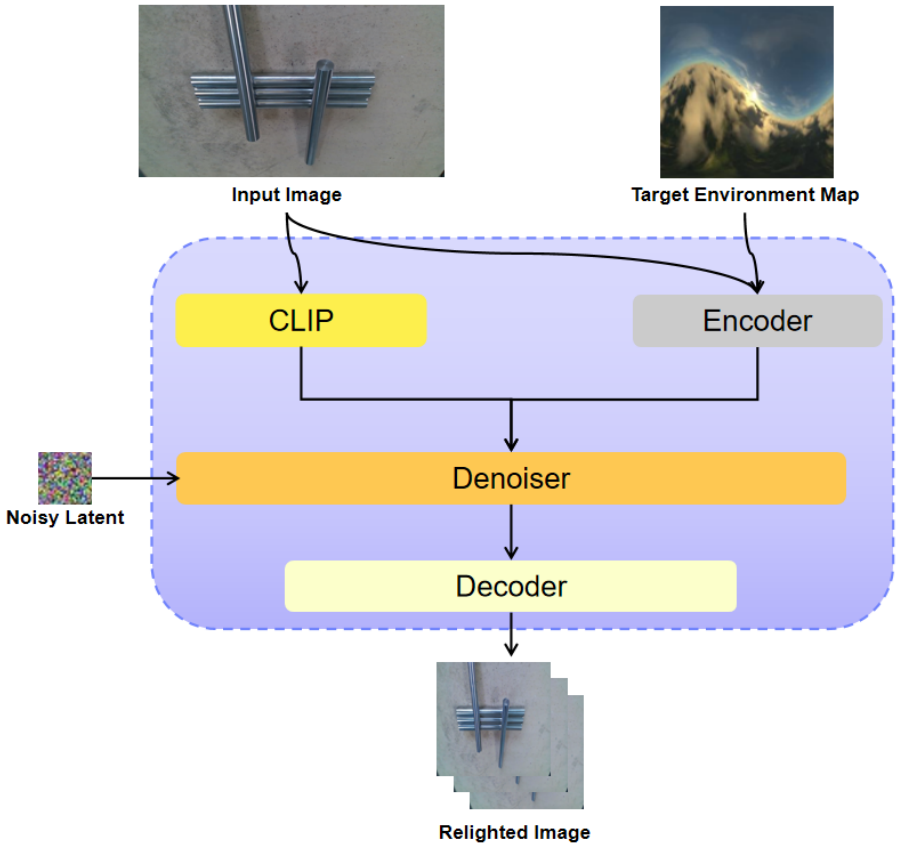


Figure 5.2: Architecture of the relighting module. Neural Gaffer is a latent diffusion model conditioned on the input image and rotated lighting maps.

5.2.2 Multi-illumination Data Generation

To enrich the geometric cues available for surface estimation, we first apply Neural Gaffer [107] as our relighting module $\mathcal{R}$ to synthesize images under various lighting conditions from the original input views. Given the input image I_{in}, it generates a set of relight images $\{I_l\}_{l=1}^{L}$, each corresponding to a different lighting condition. Figure 5.2 illustrates the overall pipeline. These synthetic images improve the photometric diversity without the need for physical lighting setups.

To maximize geometric information extraction, we use two different kinds of HDR environment maps. For each environment map, we perform eight evenly spaced horizontal rotations to simulate lighting from different angles. Specifically, we use 2 different HDR lighting environments and apply 8 rotations per lighting, resulting in 16 lighting conditions per input view.

By simulating realistic and physically consistent lighting effects, Neural Gaffer intro-

duces strong shading variations, which are essential for resolving ambiguities in normal estimation, especially on textureless or reflective surfaces.

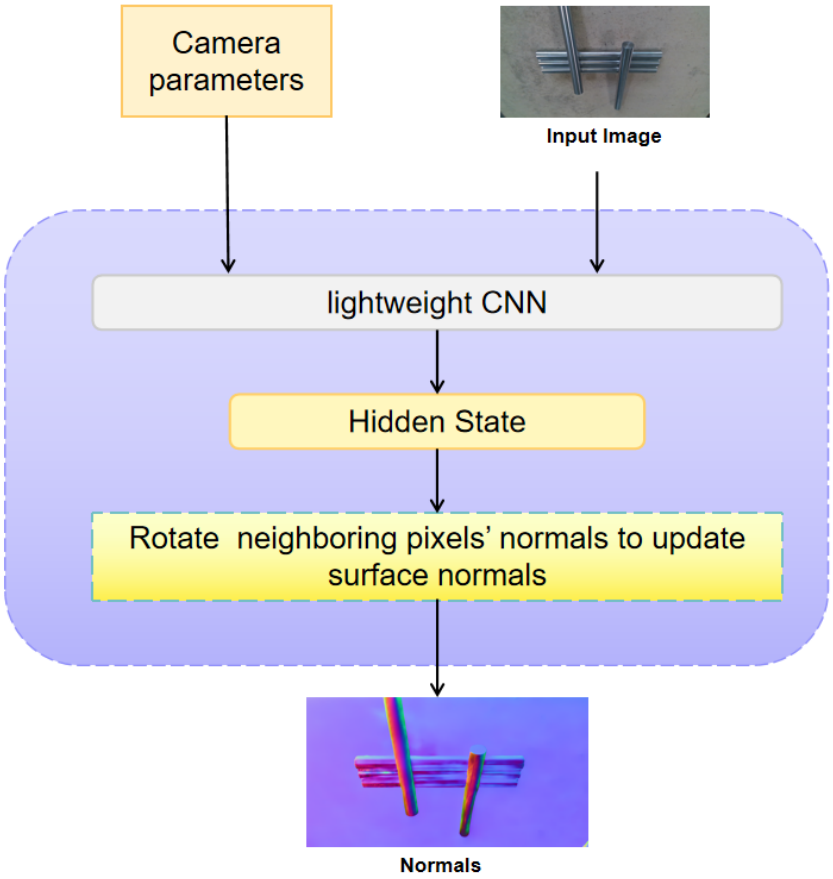


Figure 5.3: Overview of the DSINE normal estimation process. Given an input image and camera parameters, a lightweight CNN extracts features and generates an hidden state. This hidden state is used to iteratively update surface normals by rotating neighboring pixels' normals, resulting in high-quality normal estimation.

5.2.3 Normal Estimation and Supervision

The relighting module $\mathcal{R}$ generates a set of relight images that offer diverse shading information under different lighting conditions. These images are fed into our normal estimation module $\mathcal{N}$, which is implemented using DSINE [5], a learning-based approach designed especially for accurate normal prediction under spatially varying illumination. To better illustrate the architecture of the normal estimation module, we visualize the internal workflow of DSINE in Figure 5.3.

For each relight image $I_{i,l}$, DSINE predicts a normal map $N_{i,l}^{\text{cam}} \in \mathbb{R}^{H \times W \times 3}$ in the

camera coordinate system. The model leverages consistency across multiple lighting variations to disambiguate surface orientation, particularly in challenging regions with complex reflectance properties, such as areas with specular highlights.

In order to guarantee compatibility with the world-space 3D Gaussian representation, we transform the predicted normals into the world coordinate system using the corresponding camera extrinsics $\mathbf{T}_i$. The resulting world-space normals are denoted as $N_{i,l}^{\text{world}}$, and serve as the supervision signal $N_{\text{supervised}}$ for the subsequent Gaussian optimization process.

These multi-illumination normal maps provide geometrically meaningful guidance that complements the photometric loss, thereby enforcing surface-level consistency throughout the training procedure.

5.2.4 3D Reconstruction with Gaussian Splatting

The final stage of our pipeline optimizes a 3D Gaussian Splatting model to achieve photo-realistic rendering and accurate surface reconstruction. We build upon the Specular Gaussian Splatting framework $\mathcal{G}$ proposed by [322], extending it with normal supervision to enhance geometric fidelity, particularly in specular regions.

We follow the standard Gaussian rasterization pipeline to formulate image rendering. Each Gaussian is associated with a view-dependent appearance model, represented using the first three orders of spherical harmonics. The contribution of a Gaussian to the final image is determined by its opacity, which is modeled as a 2D Gaussian function projected onto the image plane. Specifically, the opacity term α_i is defined as:

$$\alpha_i = \sigma_i \exp\left(-\frac{1}{2}(\mathbf{p}-\boldsymbol{\mu}_i)^\top \Sigma_i'^{-1}(\mathbf{p}-\boldsymbol{\mu}_i)\right), \tag{5.3}$$

where $\boldsymbol{\mu}_i$ denotes the 2D projection of the i-th Gaussian onto the image plane, Σ_i' is the associated covariance matrix that defines the Gaussian's spatial extent, and σ_i is a scalar opacity value. This formulation facilitates a spatially smooth attenuation of influence for each Gaussian in screen space.

In addition, we design a specialized loss function based on the vanilla 3DGS for training. Since the relighting module expands the training set by generating 16× additional novel views, the synthesized images may introduce noise and geometric inconsistencies. To mitigate the effect of such diffusion-induced artifacts, we assign different supervision weights to real and generated images. Specifically, the overall training objective is formulated as:

$$\mathcal{L}_{\text{total}} = \lambda_{\text{gt}}\mathcal{L}_{\text{gt}} + \lambda_{\text{normal}}\mathcal{L}_{\text{normal}} + \lambda_{\text{novel}}\mathcal{L}_{\text{novel}}. \tag{5.4}$$

The weights for each component are set as $\lambda_{\text{gt}} = 0.7$, $\lambda_{\text{normal}} = 0.15$, and $\lambda_{\text{novel}} = 0.15$.

The loss for original images is defined as:

$$\mathcal{L}_{\text{gt}} = \frac{1}{|\mathcal{D}_{\text{gt}}|} \sum_{i \in \mathcal{D}_{\text{gt}}} \mathcal{L}_{\text{photo}}\big(I_i^{\text{render}}, I_i^{\text{gt}}\big), \tag{5.5}$$

The losses for normal supervision and relighted novel views are computed as follows:

$$\mathcal{L}_{\text{normal}} = \frac{1}{|\mathcal{D}_{\text{normal}}|} \sum_{k \in \mathcal{D}_{\text{normal}}} \mathcal{L}_{\text{normal}}\big(N_k^{\text{pred}}, N_k^{\text{gt}}\big), \tag{5.6}$$

$$\mathcal{L}_{\text{novel}} = \frac{1}{|\mathcal{D}_{\text{novel}}|} \sum_{j \in \mathcal{D}_{\text{novel}}} \mathcal{L}_{\text{photo}}\big(I_j^{\text{render}}, I_j^{\text{diff}}\big). \tag{5.7}$$

5.3 Experiments

5.3.1 Implementation Details

Our method is implemented in PyTorch and CUDA, and our 3D Gaussians are trained using the Spec-GS framework [322], extended with view-dependent shading and normal supervision. Experiments are conducted on a single NVIDIA Tesla A800 GPU. We evaluate our approach on several real-world scenes featuring transparent and specular surfaces, such as bottles, glass objects, and household items.

The input views are captured under fixed camera poses, with relight images synthesized by Neural Gaffer [107] using two HDR environment maps and eight horizontal rotations per lighting. To obtain high-quality normal supervision, we apply the DSINE model [5] to the relight images, followed by transformation to world coordinates using COLMAP-extracted camera poses.

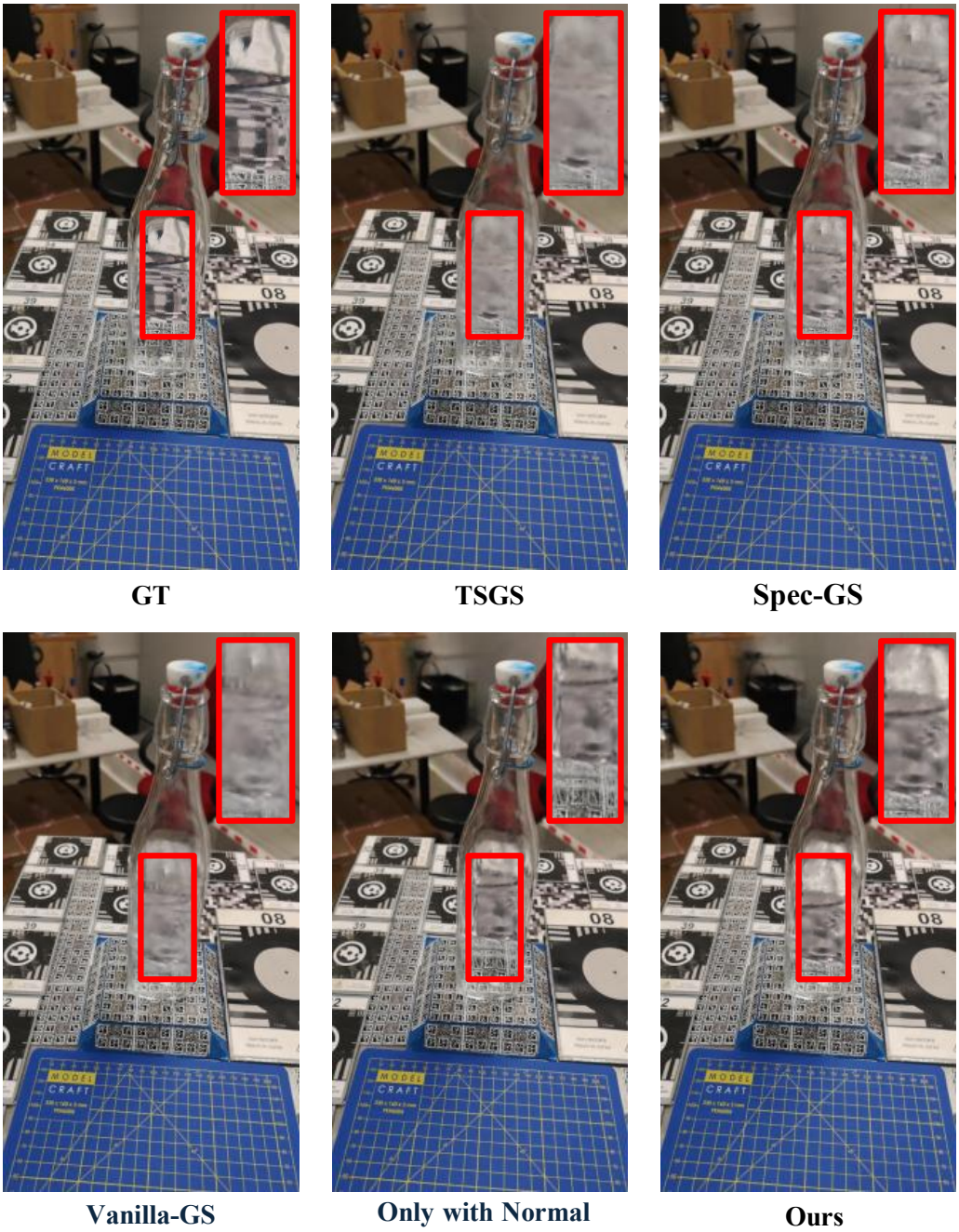


Figure 5.4: Qualitative comparison on the *Bottle* scene. The baselines fail to recover complete details due to transparency and refraction. Normal supervision slightly improves surface sharpness, but our full method with relighting achieves clearer contours and more complete structure.

We evaluate our method on five object categories drawn from different datasets: the *bottle* and *glass* objects are sourced from the NeRFBK dataset [314], the *household*

objects are from the ClearPose dataset [29], and the *metal objects* (referred to as *dimo*) are selected from the Industrial Metal Objects dataset [45]. These datasets cover a broad range of challenging material properties including transparency, specularity, and metallic reflectance, providing a comprehensive benchmark for evaluating reconstruction quality.

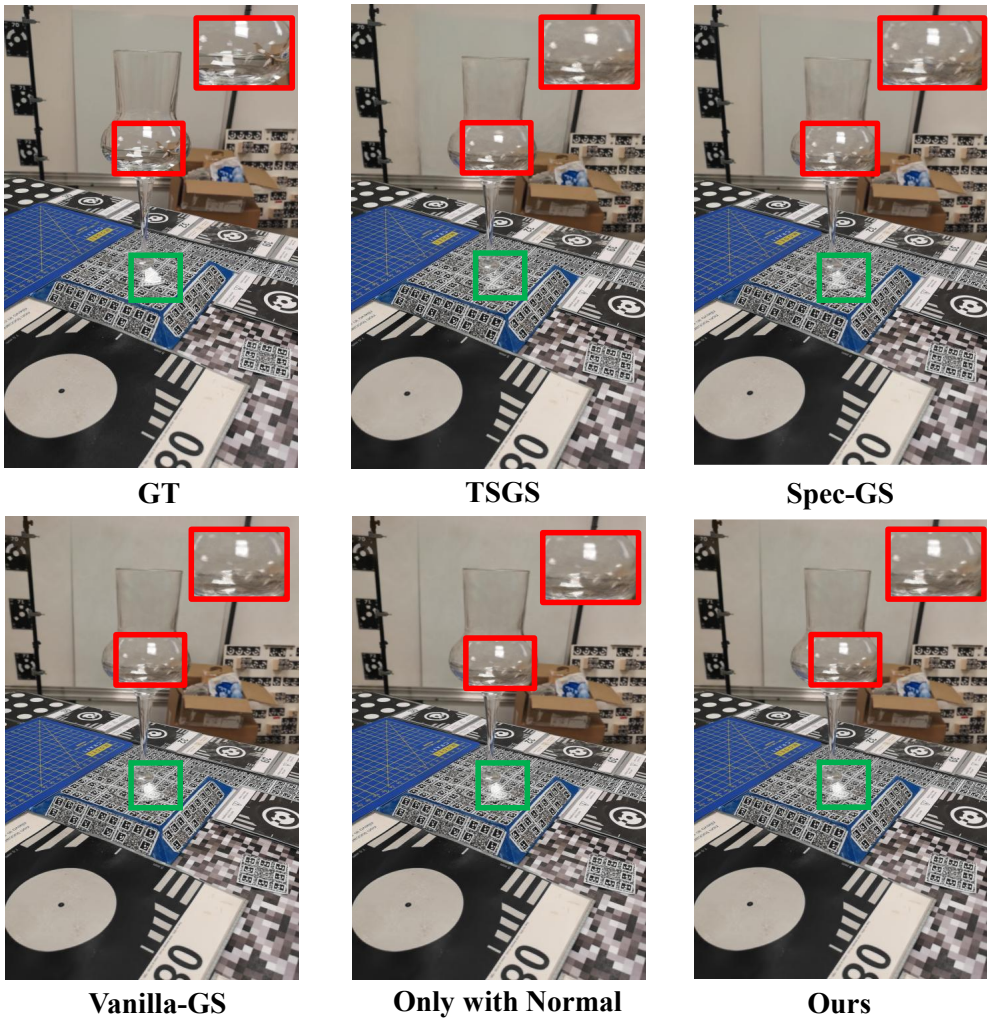
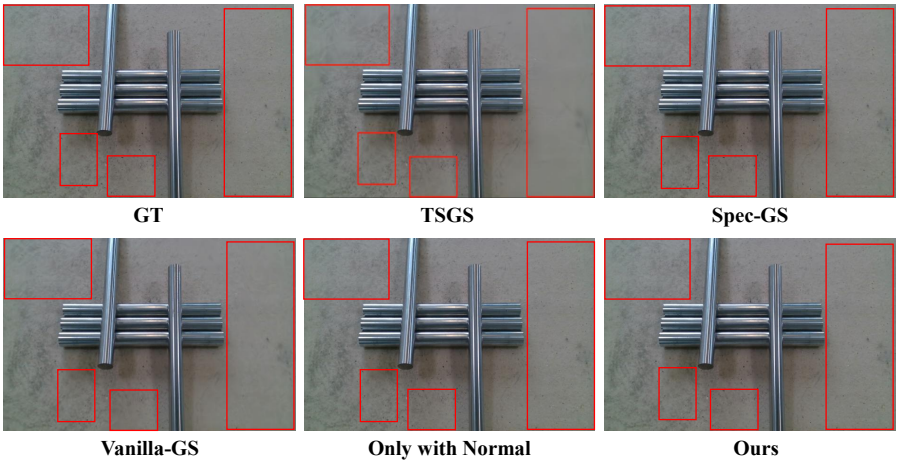


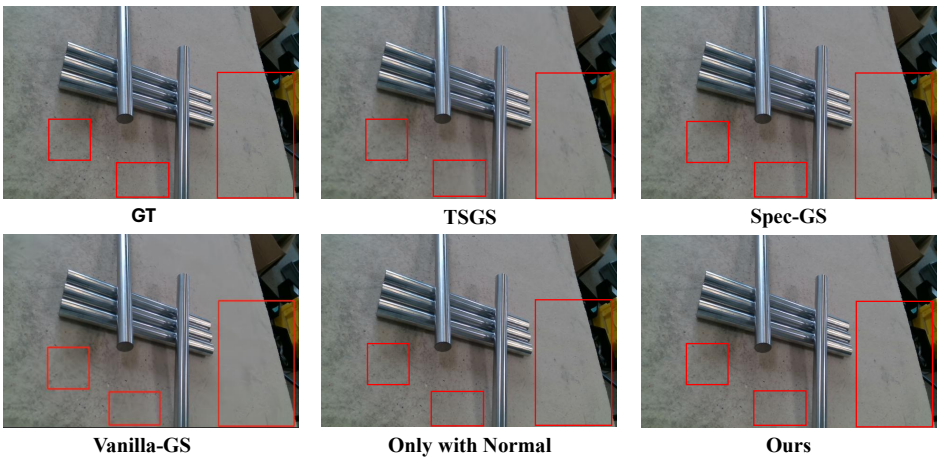
Figure 5.5: Qualitative comparison on the *Glass* scene. Similar to the *Bottle* case, the baseline methods suffer from inconsistent rendering with respect to the ground truth in refractive regions (highlighted by the red and green boxes). In contrast, our full model, incorporating normal estimation and relighting, produces clearer object boundaries and preserves finer structural details.

5.3.2 Qualitative Experiment

We provide qualitative comparisons to visually assess the effectiveness of our method. Figures 4–8 present rendered images and point cloud reconstructions for three representative scenes: *Bottle*, *Glass*, and *DIMO*. We compare our approach with four variants, including Vanilla-GS [110], TSGS [131], Spec-GS [322], and a normal-supervised variant (denoted as *Only with Normal*).



(a) DIMO View1

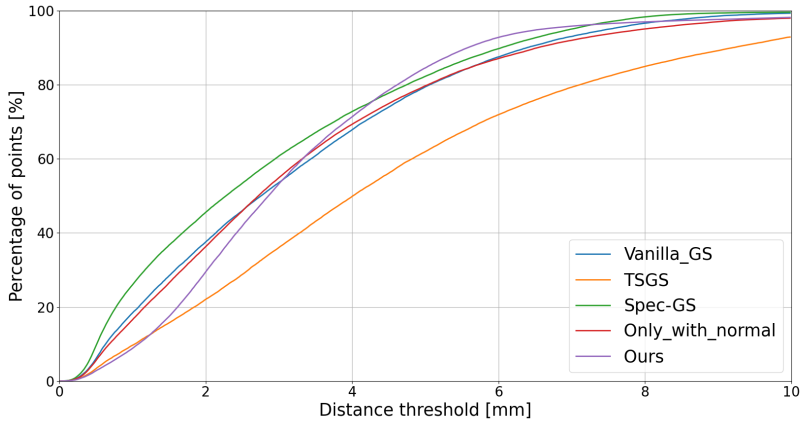


(b) DIMO View2

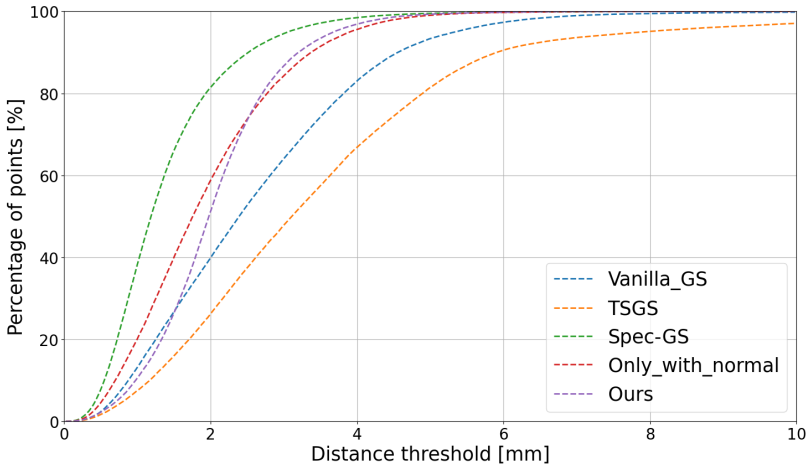
Figure 5.6: Qualitative comparison on *DIMO* scenes (containing only 14 images) containing industrial metal parts. In both views, our full model with normal and relighting supervision yields more accurate metal highlights, sharper specular reflections, and clearer background recovery.

On the Bottle and Glass scenes (Figures 5.4 and 5.5), which contain transparent and

refractive surfaces, Vanilla-GS, TSGS, Spec-GS suffer from incomplete geometry and blurry scene. Even though normal supervision alone can slightly increase sharpness, it cannot accurately represent surface curvature. However, our comprehensive approach recovers finer details and more complete contours, producing reconstructions that are structurally and structurally accurate.



(a) Accuracy of Bottle

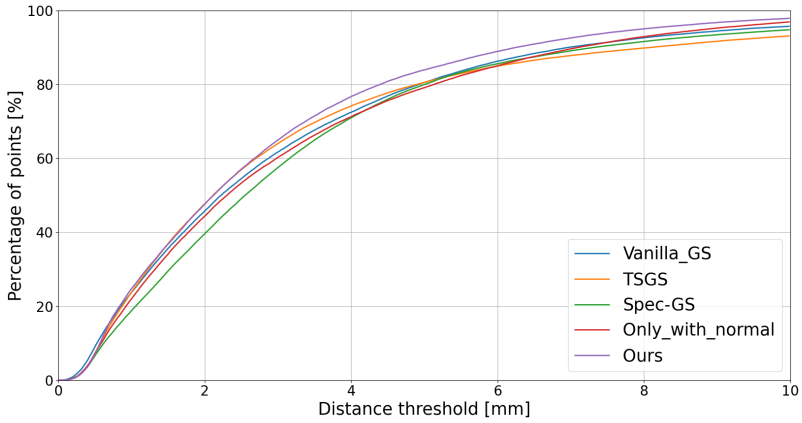


(b) Completeness of Bottle

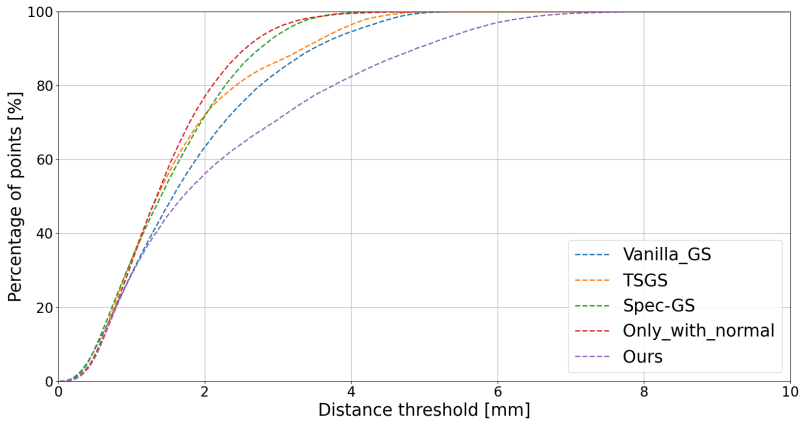
Figure 5.7: Accuracy and completeness evaluation on the *Bottle* scene. Solid lines represent accuracy (distance from reconstructed points to the GT), while dashed lines denote completeness (coverage of GT points by reconstructed points).

In the DIMO scene, which contains highly specular industrial metal objects, Vanilla-

GS and TSGS struggle with accurate background reconstruction. It fails to faithfully capture the sharp reflectance patterns of metallic parts and produces blurry background regions, as shown in Figure 5.6. In contrast, our method reconstructs cleaner backgrounds and significantly enhances specular highlights and geometric details on metal surfaces. These results demonstrate the effectiveness of our approach in modeling view-dependent reflections and mitigating artifacts caused by sparse observations, particularly for complex and highly reflective materials.



(a) Accuracy of Glass



(b) Completeness of Glass

Figure 5.8: Accuracy and completeness evaluation on the *Glass* scene. Solid lines represent accuracy (distance from reconstructed points to the GT), while dashed lines denote completeness (coverage of GT points by reconstructed points).

Overall, the qualitative results substantiate our claim that, particularly in challeng-

ing material scenarios where photometric cues alone are insufficient, the combination of relighting and normal supervision yields reconstructions that are both geometrically accurate and visually compelling.

5.3.3 Quantitative Evaluation

Quantitative Results of Tested Scenes

We demonstrate the geometric accuracy and completeness in Figure 5.7 and Figure 5.8, and report the quantitative cloud-to-cloud comparison results in Tables 5.1 and 5.2.

Figure 5.7 and Figure 5.8 presents the accuracy and completeness curves for the *Bottle* and *Glass* scenes. As shown in Figures 5.7(a) and 5.8(a), our method (purple curve) consistently achieves higher accuracy than the baseline approaches. For completeness on the *Bottle* scene, our method ranks second among all compared methods.

Figure 5.9 and Figure 5.10 with Tables 5.1 and 5.2 present the quantitative results of geometric error metrics (RMSE, MAE, STD, ME). Across both object categories, our approach achieves the lowest RMSE and MAE for the *Bottle* and *Glass* objects, indicating improved 3D geometry quality of the reconstructed point clouds. For instance, on the *Glass* object, our method outperforms all baselines, including the normal-only supervision variant, achieving an RMSE of 2.70, MAE of 2.04, and STD of 1.76. These results demonstrate that the combined supervision from relighted images and surface normals substantially enhances both the accuracy and density of the reconstructed point clouds, particularly for challenging transparent and specular surfaces. Moreover, the lower STD values indicate that our approach produces more reliable and consistent reconstructions.

Method	**RMSE**	**MAE**	**STD**	**ME**
Vanilla-GS	3.48	2.80	2.07	2.86
TSGS	5.08	4.07	3.04	-3.18
Spec-GS	3.29	2.57	2.05	**-1.54**
Only with Normal	3.42	2.73	2.06	2.88
Ours	**3.16**	**2.71**	**1.62**	2.87

Table 5.1: Quantitative results of cloud-to-cloud comparison on the *Bottle* object. Our approach achieves the best performance in RMSE (3.16), MAE (2.71), and STD (1.62), indicating superior reconstruction accuracy and stability compared to all baselines, including the normal-only supervision variant.

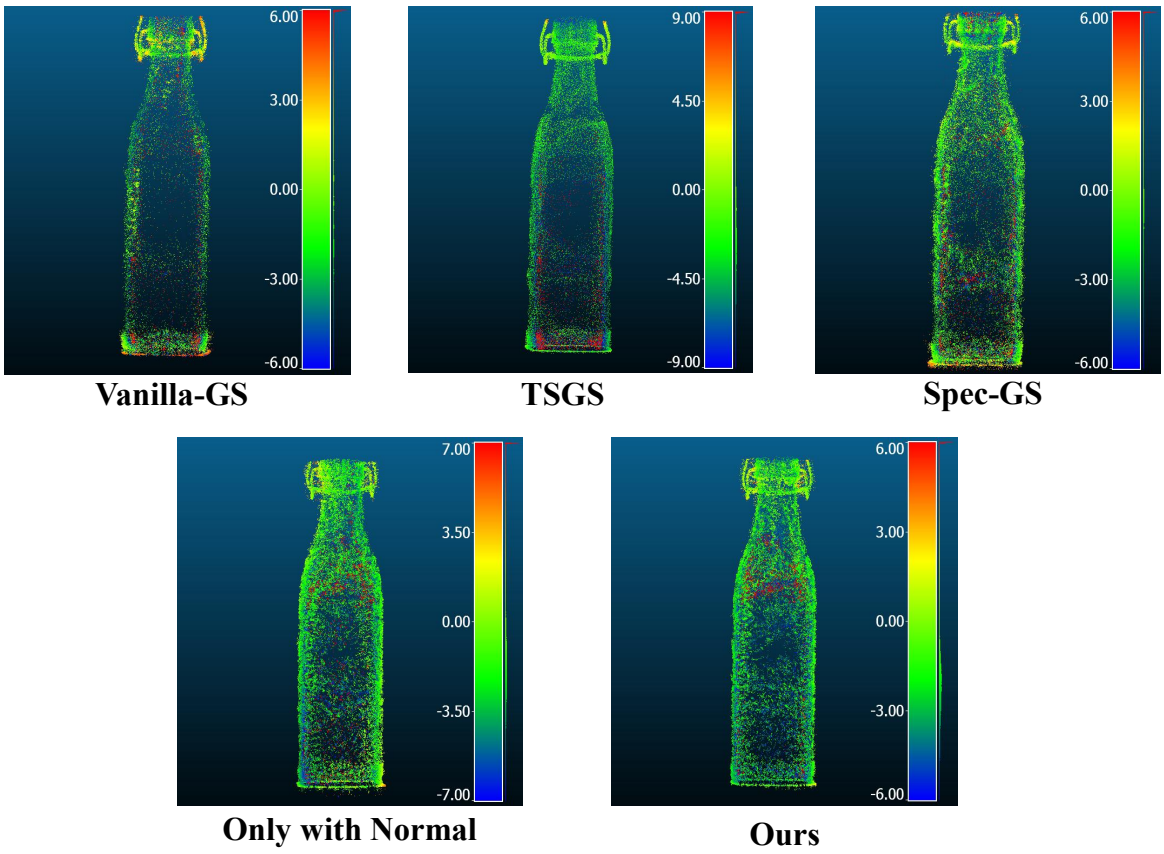


Figure 5.9: Visualization of Cloud-to-Cloud comparisons on the *bottle* object.

Method	RMSE	MAE	STD	ME
Vanilla-GS	2.85	2.11	1.91	2.35
TSGS	3.32	2.44	2.26	**-0.05**
Spec-GS	3.13	2.44	1.95	-0.11
Only with Normal	4.16	3.30	2.59	3.40
Ours	**2.70**	**2.06**	**1.76**	2.23

Table 5.2: Quantitative results of cloud-to-cloud comparison on the *Glass* object. Consistent with the results on the *Bottle* object, our method outperforms all competing approaches in terms of RMSE, MAE, and STD, demonstrating robust reconstruction quality and stable performance on transparent objects.

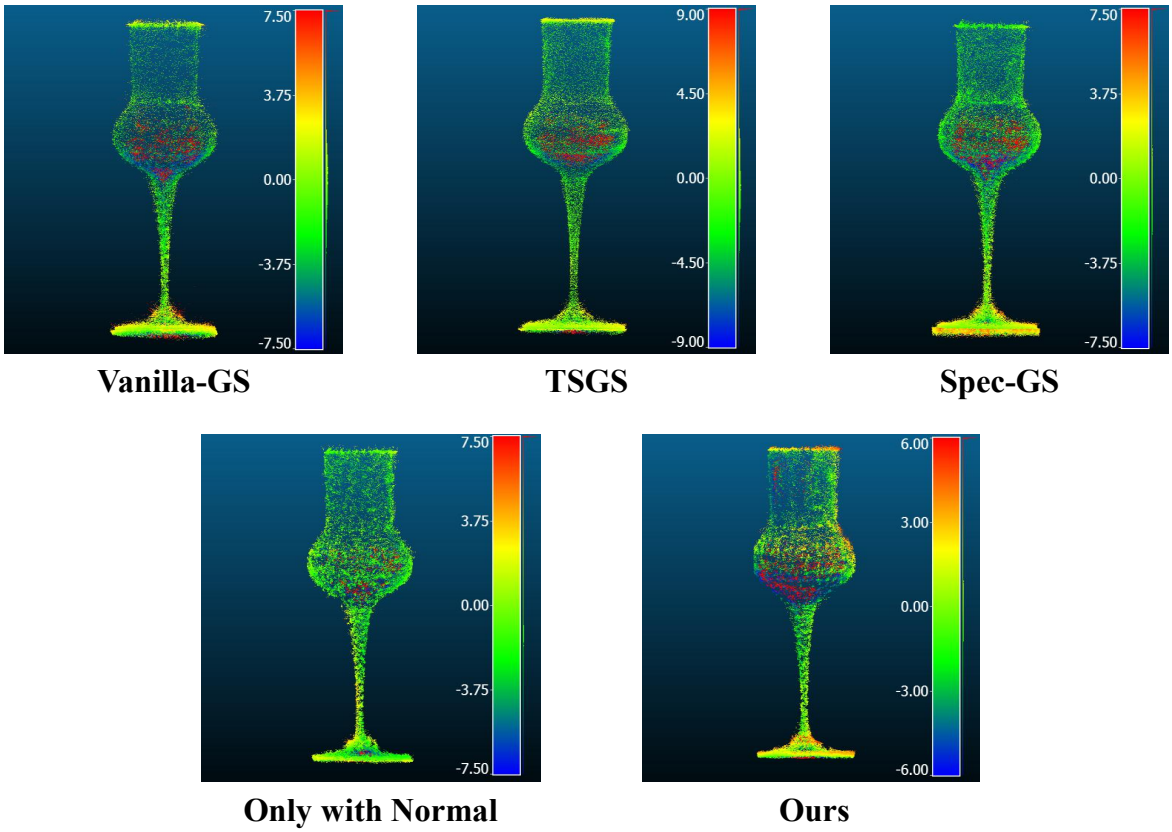


Figure 5.10: Visualization of Cloud-to-Cloud comparisons on the *glass* object.

We also evaluate our method using photometric metrics, as shown in Table 5.3, even though Colla-GS is primarily designed for geometric reconstruction rather than image rendering. Nonetheless, our approach achieves competitive SSIM, PSNR, and LPIPS scores across all datasets. For the DIMO object, which contains complex specular reflections and consists of only 14 images, our model attains the best performance across all metrics, with an SSIM of 0.9858, PSNR of 41.47, and LPIPS of 0.0597. This demonstrates the effectiveness of our data generation module, whereas other approaches suffer significant performance drops due to limited input and supervision.

The results in both geometric and photometric evaluations demonstrate that our method not only performs well in geometric reconstruction across challenging materials, but also maintains strong image rendering capabilities. The combination of normal supervision and relighting enables a more complete and robust representation than either approach alone.

Object	**Method**	**SSIM ↑**	**PSNR ↑**	**LPIPS ↓**
Bottle	Vanilla-GS	0.9108	27.19	0.156
	Spec-GS	0.9040	26.36	0.164
	TSGS	0.8757	25.40	0.164
	Normal	0.8429	22.26	0.224
	Normal+Relighting	0.8432	22.05	0.223
Glass	Vanilla-GS	0.9262	26.75	0.130
	Spec-GS	0.9309	27.24	0.127
	TSGS	0.9320	25.54	0.106
	Normal	0.9210	26.31	0.137
	Normal+Relighting	0.9217	26.52	0.136
DIMO	Vanilla-GS	0.9654	40.63	0.231
	Spec-GS	0.7121	19.95	0.485
	TSGS	0.9126	30.31	0.259
	Only with Normal	0.9857	41.69	0.062
	Ours	**0.9858**	**41.47**	**0.059**
Household	Vanilla-GS	0.7372	21.57	0.257
	Spec-GS	0.4148	15.17	0.542
	TSGS	0.6121	18.79	0.321
	Only with Normal	0.5001	18.07	0.498
	Ours	0.5034	18.03	0.497

Table 5.3: Photometric evaluation metrics on four representative scenes. Higher SSIM/PSNR and lower LPIPS indicate better quality.

5.3.4 Ablation Study

To better understand the contribution of each component in our pipeline, we conduct an ablation study using the household object from the ClearPose dataset [29], as shown in Figure 5.11. Although the overall reconstruction quality is limited by the low resolution of the input images (652×486), our proposed model still achieves superior performance on transparent surfaces. In particular, it preserves the geometry and transparency of the glass structures more effectively, while producing fewer rendering artifacts compared to both the baselines and the normal-only variant. These results highlight the benefit of integrating normal supervision with relighted image guidance when handling challenging materials such as glass.

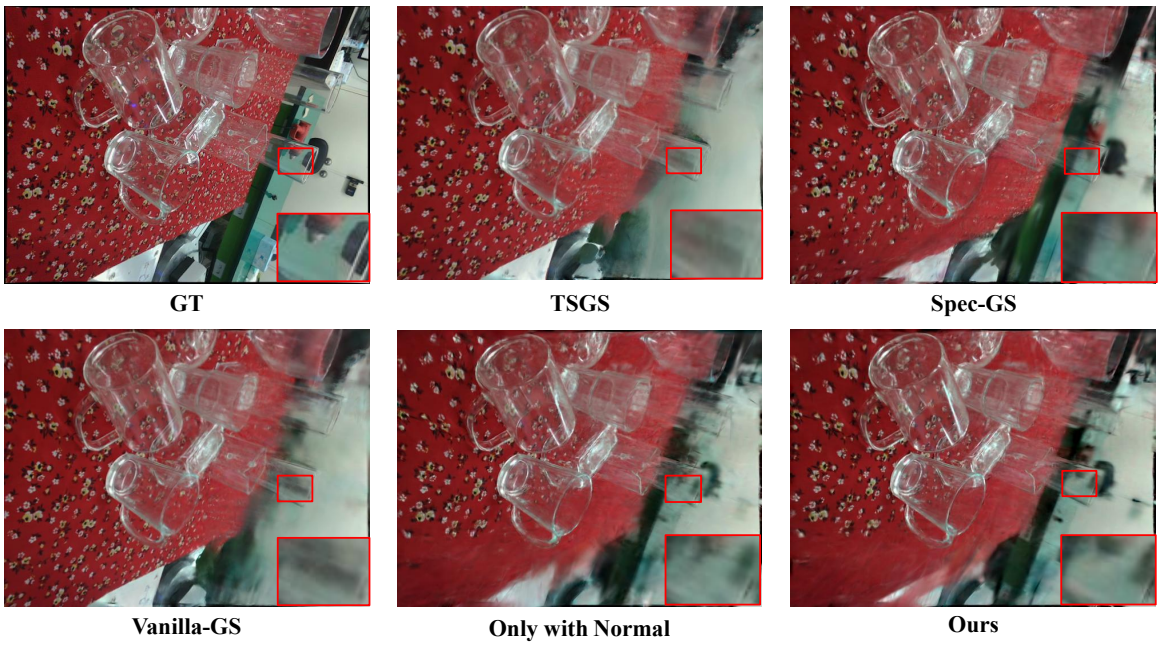


Figure 5.11: Qualitative ablation results on the *household* object. Despite the low resolution of input images, our method better reconstructs the transparent geometry compared to baselines and the normal-only variant.

5.4 Conclusions

In this chapter, we propose a novel pipeline that integrates relighting-based data augmentation with surface normal supervision to enhance 3D Gaussian Splatting. Our method leverages synthetic relighted images to extract richer geometric cues through a normal estimation network, resulting in improved rendering and reconstruction quality, particularly for challenging transparent and reflective objects. Extensive experiments on multiple datasets—including NeRFBK, ClearPose, and DIMO—demonstrate that our approach outperforms existing 3DGS baselines in both quantitative metrics and qualitative fidelity. Notably, our model exhibits clear advantages in dense point cloud recovery and in accurately modeling complex view-dependent reflectance on metallic surfaces.

5.5 Limitations and Future Works

Although the presented approach increases the geometric and photometric fidelity of the 3DGS reconstructions, it still has some limitations. First, the estimated normals obtained during training are often noisy and inconsistent, suggesting that more advanced normal estimation techniques are necessary for reliable geometric supervision.Second, although our model improves the reconstruction of transparent and specular objects, the current formulation of 3DGS relies on ordered alpha blending along straight viewing rays, which inherently assumes a single-ray path and does not explicitly model ray bending caused by refraction. As a result, background details observed through transparent objects cannot be faithfully recovered, limiting realism in such scenarios.

As possible future works, more robust and learning-adaptive normal estimation strategies designed for transparent and reflective materials are suggested.Regarding refraction, instead of explicitly modeling physical ray bending, which is not directly supported by the current 3DGS rendering pipeline, several approximation strategies could be considered. One possible direction is to incorporate view-dependent appearance models or learnable feature fields that implicitly encode refractive distortions. Another direction is to introduce multi-layer or volumetric Gaussian representations that approximate light transport through transparent media. Additionally, hybrid approaches that combine 3DGS with ray-based rendering techniques could provide a more physically grounded solution.

5.5.1 Summary and Relation to Research Question

This chapter directly addresses **Research Question 2** by demonstrating that the incorporation of physical priors, specifically surface normal supervision and multi-illumination cues, can effectively decouple geometry from appearance in 3DGS. By leveraging relighting-based data augmentation and normal-guided optimization, the proposed method significantly improves reconstruction fidelity for non-collaborative surfaces such as transparent and reflective objects. These results confirm that augmenting 3DGS with geometry-aware supervision is a viable direction for overcoming its inherent limitations in challenging real-world scenarios.

Chapter 6

Reconstruct and Understand Autonomous Driving Scenes with 3D Gaussian Splatting

6.1 Introduction

With the wide application of autonomous driving [140], [317], [87] [231], researchers gradually focus on better perception and forecasting methods [319], which are related to the decision-making ability and robustness of the system [256], [40]. Most current frameworks consist of perception [81], forecasting, and planning separately [84]. The most commonly used perception method is 3D target detection using vision and LIDAR fusion [87], [135], [140], allowing the model to better forecast future scenes and do motion planning. Since most 3D target detection methods [368, 224, 290] are unable to obtain fine-grained information in the environment, they are non-robust in planning [172] in the subsequent model, which affects the system safety. Current perception methods primarily utilize both LiDAR [356], [196] and cameras [165], but the high cost of LiDAR and the computational demands of multimodal fusion pose challenges to the real-time performance and robustness of autonomous driving systems.

While 3D occupancy has been widely used as an intermediate representation for world models, most existing approaches rely on discretized voxel grids or BEV-based representations, which are inherently limited by resolution and often suffer from information loss during the projection from 2D observations. In contrast, we leverage 3D Gaussian Splatting [110] as a continuous and explicit representation to bridge 2D perception and 3D occupancy generation. This enables more faithful reconstruction

of scene geometry from sparse multi-view inputs, while preserving fine-grained spatial structures that are typically lost in discretized pipelines.

In this chapter, we introduce **RenderWorld** [311], an autonomous driving framework for prediction and motion planning, which is trained on 3D occupancy labels generated by a Gaussian-based Img2Occ module. RenderWorld proposes an self-supervised Img2Occ module with Gaussian Splatting [110], trained on 2D multi-view depth and semantic images to generate 3D occupancy labels required for the world model. To enable the world model to better understand the scene represented by 3D occupancy, we propose the Air Mask Variational Autoencoder (AM-VAE) upon a vector-quantized variational autoencoder (VQ-VAE) [268]. This improves the inference capability of our world model by enhancing the granularity of the scene representation.

In order to verify the efficiency and reliability of RenderWorld, we evaluate the 3D occupancy generation and motion planing on NuScenes [17] separately. In summary, our contributions are:

1) To propose RenderWorld (see Figure 6.1), a pure 2D autonomous driving framework that uses labeled 2D images to train a Gaussian-based occupancy prediction module (Img2Occ) for generating the 3D labels required by the world model.

2) To improve spatial representation abilities, we introduce AM-VAE, which improves forecasting and planning in world models while reducing memory consumption by separately encoding air and non-air voxels.

6.2 Methodology

In this section, We describe the overall implementation of RenderWorld. We firstly propose an Img2Occ Module for occupancy prediction and 3D occupancy labels generation (Section 6.2.1). Subsequently, we introduce a module based on the Air Mask Variational Autoencoder (AM-VAE) to optimize occupancy representation and enhance data compression efficiency (Section 6.2.2). Finally, we elaborate on how to integrate the World Model for accurate prediction of 4D scene evolution (Section 6.2.3).

6.2.1 3D Occupancy Prediction with Multi-frame 2D Labels

To enable 3D semantic occupancy prediction and future 3D occupancy labels generation, we design an Img2Occ Module which is illustrated in Figure 6.2. Using images from multi-cameras $\{Img_i\}_{i=1}^{N}$ as inputs, we firstly extract 2D image features using a

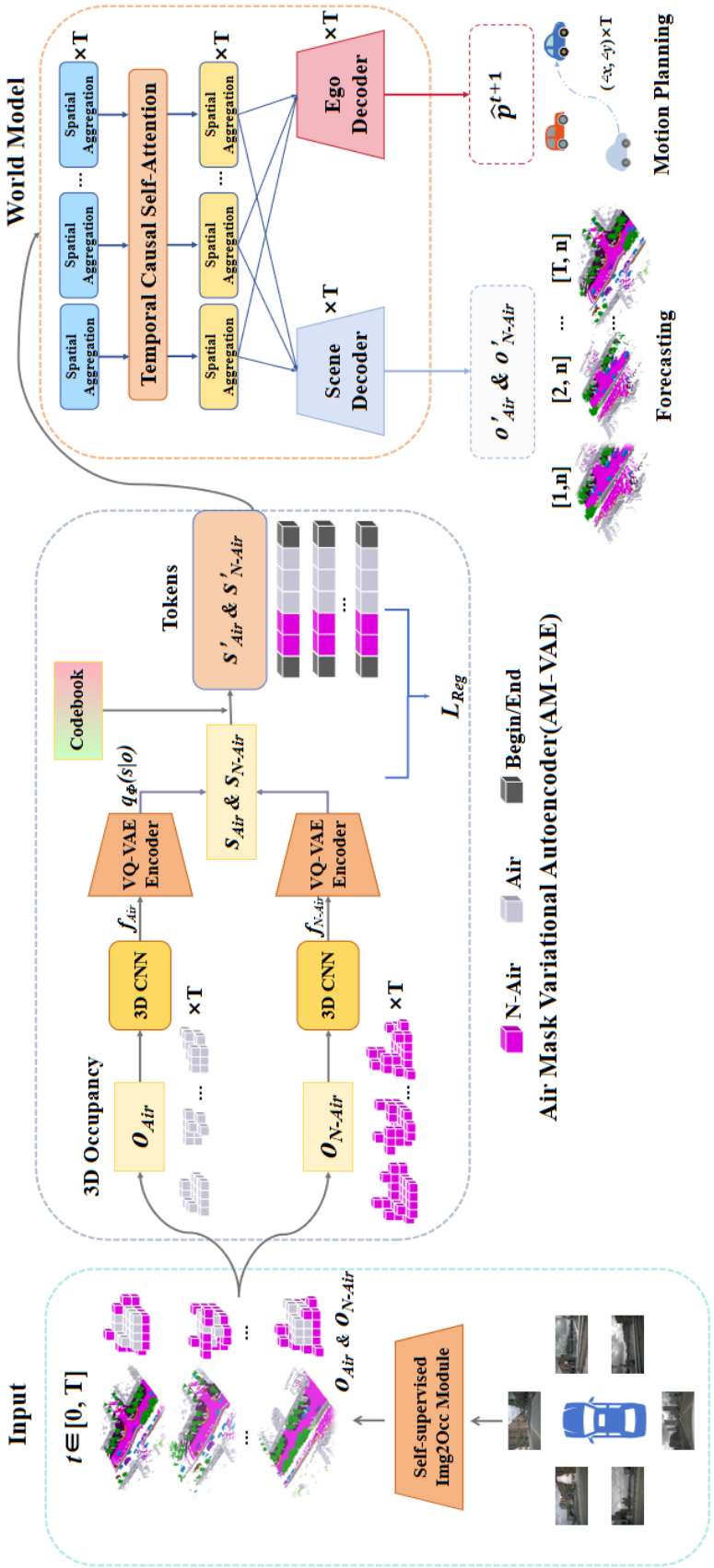


Figure 6.1: General pipeline of RenderWorld. We firstly generate the 3D occupancy labels through an Img2Occ Module (Figure 6.2). Then, using Air Mask Variational Autoencoder (AM-VAE) described in Section 6.2.2, the separated air and non-air voxels are independently encoded into latent representations (i.e., discrete tokens). Finally, these latent representations are processed according to the specifications in Section 6.2.3, and based on this, the voxels and trajectories are predicted, ultimately outputting the predicted occupancy and self-planning.

pretrained BEVStereo4D [86] backbone and Swin Transformer [164]. Then, these 2D messages are interpolated into 3D space to produce volume features by leveraging the known intrinsic parameters $\{I_i\}_{i=1}^{N}$ and extrinsic parameters $\{E_i\}_{i=1}^{N}$. To project the 3D occupancy voxels onto multi-camera semantic maps, we apply Gaussian Splatting [110], an advanced real-time rendering pipeline. Inspired by [169], we initialize anchor points with a learnable scale at the center of each voxel to approximate scene occupancy. The attributes of each anchor are determined based on the relative distance and viewing direction between the camera and the anchor. This anchor set is then used to initialize a Gaussian set with semantic labels $\{G_x\}_{x=1}^{N}$. Each Gaussian point x is then represented by a full 3D covariance matrix Σ in world space and its center position μ, and the color of each point is decided by the semantic label at that point.

$$G(x) = e^{-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)} \tag{6.1}$$

Directly optimizing Σ may lead to infeasible matrices as it must be positive semi-

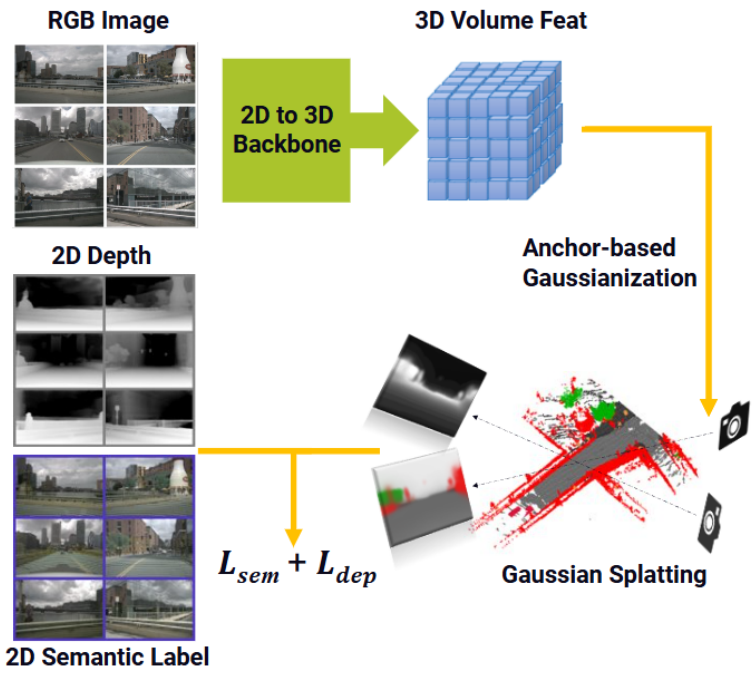


Figure 6.2: Training paradigm of 2D-to-3D occupancy prediction Module. Our proposed Img2Occ Module utilizes 2D labels to train the 3D occupancy network, allowing the model to take advantage of detailed 2D pixel-level semantics and depth supervision.

definite. To ensure the validity of Σ, it is decomposed into the scaling matrix S and

the rotation matrix R to characterize the geometry of a 3D Gaussian ellipsoid:

$$\Sigma = RSS^TR^T \tag{6.2}$$

Then the 3D Gaussians are projected to 2D for rendering by computing the camera space covariance matrix Σ' :

$$\Sigma' = JW\Sigma W^TJ^T, \tag{6.3}$$

where J is the Jacobian matrix of the affine approximation of the projection transformation and W is the viewing transformation. The semantic / depth of each pixel can then be calculated by applying alpha blending onto sorted Gaussians:

$$D = \sum_i^N (d_i)a_i \prod_j^{i-1}(1-a_j), \tag{6.4}$$

$$S = \sum_i^N (s_i)a_i \prod_j^{i-1}(1-a_j), \tag{6.5}$$

where s_i/d_i is the rendered semantic / depth of a 3D Gaussian, a_i is the product of an evaluated 2D Gaussian projection and its corresponding opacity.

To calculate the difference between ground truth depth and rendered depth, we utilize the Pearson correlation which can measure the distribution difference between 2D depth maps follows the following function:

$$L^i_{dep} = Corr(\bar{d}_i,\ d_i) = \frac{Cov(\bar{d}_i,\ d_i)}{Var(\bar{d}_i,\ d_i)}, \tag{6.6}$$

where $\bar{d}_i$ is the ground truth depth image and d_i is the rendered depth image.

Finally, we construct the loss function with a cross-entropy loss L_{sem} for supervising semantic segmentation and L_{dep} for depth supervision, the overall loss can be computed as follows:

$$L^i_{i2o} = L^i_{sem} + L^i_{dep} \tag{6.7}$$

Using the well-trained checkpoint, we generate 3D occupancy labels, which are then input into the subsequent AM-VAE module.

6.2.2 Air Mask Variational Autoencoder (AM-VAE)

Traditional Variational Autoencoders (VAEs) fail to encode the distinct features of non-air voxels, which hampers the model from representing scene elements at a fine-grained level. To address this issue, we introduce the Air Mask Variational Autoencoder (AM-VAE), a novel VAE that involves training two distinct Vector Quantized Variational Autoencoders (VQ-VAE) [268] to encode and decode air and non-air occupancy voxels separately.

Assuming o represents the input occupancy representation, and o_{Air} and o_{N-Air} represent the air and non-air voxels. We first utilize a 3D convolutional neural network to encode the occupancy data, with the output being a continuous latent space representation denoted as f. The encoder $q_\phi(s|o)$ maps the input f to the latent space s. Then, we use two latent variables s_{Air} and s_{N-Air} to represent the air and non-air voxels, respectively:

$$s_{Air} \sim q_\phi(s_{Air}|o_{Air}), \quad s_{N-Air} \sim q_\phi(s_{N-Air}|o_{N-Air}) \tag{6.8}$$

Each encoded latent variable s_{Air} or s_{N-Air} uses learnable codebook C_{Air} or C_{N-Air} to obtain discrete token, which is then replaced by the most similar codebook entry before being fed into the decoder. This process is represented as:

$$\begin{aligned} s'_{Air} &= \arg \min_{c_{Air} \in C_{Air}} \|s_{Air} - c_{Air}\|, \\ s'_{N-Air} &= \arg \min_{c_{N-Air} \in C_{N-Air}} \|s_{N-Air} - c_{N-Air}\| \end{aligned} \tag{6.9}$$

Then, the decoder $p_\theta(o|s)$ reconstructs the input occupancy from the quantized latent variables s'_{Air} and s'_{N-Air}:

$$\hat{o}_{Air} = p_\theta(o_{Air}|s'_{Air}), \quad \hat{o}_{N-Air} = p_\theta(o_{N-Air}|s'_{N-Air}) \tag{6.10}$$

To facilitate the separation of air and non-air elements within the occupancy representation, we denote M as the set of non-air categories. Then the indicator function for air and non-air in the modified occupancy can be defined as follows:

$$I_M(o) = \begin{cases} 1 & if\, o \in M, \\ 0 & otherwise. \end{cases} \tag{6.11}$$

The modified air occupancy o'_{Air} and non-air occupancy o'_{N-Air} are given by the

following equations:

$$\begin{aligned} o'_{Air} &= (1 - I_M(o)) \cdot o_{Air}, \\ o'_{N-Air} &= I_M(o) \cdot o_{N-Air} + (1 - I_M(o)) \cdot o_{Air} \end{aligned} \tag{6.12}$$

To reconstruct the original occupancy representation, we use a $mask = (\hat{o}_{Air} \neq 0)$ to distinguish areas filled only with air. Then the reconstructed occupancy $\hat{o}$ combines the air and non-air components as follows:

$$\hat{o} = \hat{o}_{Air} \cdot mask + \hat{o}_{N-Air} \cdot (1 - mask) \tag{6.13}$$

We then build the loss function L_{VAE} for training the AM-VAE with reconstruction loss L_{Recon} and commitment loss L_{Reg}:

$$\begin{aligned} L_{Recon} = &\mathbb{E}_{q_\phi(s_{Air}|o_{Air})}[\log p_\theta(o_{Air}|s'_{Air})] \\ &+ \mathbb{E}_{q_\phi(s_{N-Air}|o_{N-Air})}[\log p_\theta(o_{N-Air}|s'_{N-Air})], \end{aligned} \tag{6.14}$$

$$L_{Com} = \|s_{Air} - s'_{Air}\|^2 + \|s_{N-Air} - s'_{N-Air}\|^2, \tag{6.15}$$

$$L_{VAE} = L_{Recon} + \beta L_{Com} \tag{6.16}$$

AM-VAE utilizes separate codebooks for air and non-air voxels within a unified encoder-decoder setup. This method effectively captures the unique features of each voxel type, thereby improving both reconstruction accuracy and generalization potential.

6.2.3 World Model

By applying a world model in autonomous driving to encode 3D scenes into high-level tokens, our framework can effectively capture environmental complexity, enabling accurate autoregressive anticipation of future scenarios and vehicle decisions.

Inspired by OccWorld [211], we use a 3D occupancy to represent the scene and employ a self-supervised tokenizer to derive high-level scene tokens $\mathbf{T}$, and encode the spatial position of vehicles by aggregating the vehicle token $\mathbf{z}_0$. The world model is defined as w based on the current timestamp T and the number of historical frames t, then we establish the prediction with the following formula:

$$w(\mathbf{T}^T, \cdots, \mathbf{T}^{T-t}) = \mathbf{T}^{T+1}, \tag{6.17}$$

where $\mathbf{T}^{T+1}$ represents the scene tokens at the next time step.

At the same time, a temporal generative transformer architecture is adopted to effectively predict the future scene. It firstly processes scene tokens through spatial aggregation and downsampling, and then generates a hierarchical set of tokens $\{\mathbf{T}_0, \cdots, \mathbf{T}_K\}$. So as to predict the future at different spatial scales, we take multiple sub-world models $w = \{w_0, \cdots, w_K\}$ to achieve it and each sub-model w_i applies temporal attention to the tokens at each position j using the following formula:

$$\hat{\mathbf{z}}_{j,i}^{T+1} = \mathrm{TA}(\mathbf{z}_{j,i}^{T}, \cdots, \mathbf{z}_{j,i}^{T-t}), \tag{6.18}$$

where TA represents masked temporal attention, which predicts future tokens from influencing previous tokens. $\mathbf{z}_{j,i}^{t} \in \mathbf{T}_i^t$ denotes the j-th world token at scale i and timestamp t.

In the prediction module, we firstly utilize a self-supervised tokenizer e to convert the 3D scene into high-level scene tokens $\mathbf{T}$, and a vehicle token $\mathbf{z}_0$ to encode the spatial position of the vehicle. After predicting the future scene tokens, a scene decoder d is applied to decode the predicted 3D occupancy $\hat{\mathbf{y}}^{T+1} = d(\hat{\mathbf{z}}^{T+1})$, and learn a vehicle decoder d_{ego} which is for generating the vehicle displacement that relative to the current frame $\hat{p}^{T+1} = d_{ego}(\hat{z}_0^{T+1})$. The prediction module provides decision support for trajectory optimization of the autonomous driving system by generating continuous predictions of future vehicle displacements and scenario changes, ensuring safe and adaptive path planning.

We have implemented a two-stage training strategy to effectively train our prediction module. In the first phase, we train the scene tokenizer e and the decoder d using a 3D occupancy loss:

$$L_{e,d} = L_{soft}(d(e(\mathbf{y})), \mathbf{y}) + \lambda_1 \cdot L_{lovasz}(d(e(\mathbf{y})), \mathbf{y}), \tag{6.19}$$

where L_{soft} denotes the softmax loss and L_{lovasz} represents the Lovasz-softmax loss. The term λ_1 serves as a balancing factor between them.

Then we use the learned scene tokenizer e to obtain the scene tokens $\mathbf{z}$ for all frames and constrain the difference between the predicted tokens $\hat{\mathbf{z}}$ and $\mathbf{z}$. And a softmax loss is used to enforce the correct classification of $\hat{\mathbf{z}}$ to the correct code in the codebook $\mathbf{C}$. For the vehicle token, we simultaneously learn the vehicle decoder d_{ego} and apply an L2 loss on the predicted displacement $\hat{p} = d_{ego}(\hat{\mathbf{z}}_0)$ and the ground truth displacement

p. The overall loss in phase two can be formulated as follows:

$$L_{w,d_{ego}} = \sum_{t=1}^{T}(\sum_{j=1}^{M_0} L_{soft}(\hat{\mathbf{z}}_{j,0}^{t}, \mathbf{C}(\mathbf{z}_{j,0}^{t}) + \lambda_2 \; L_{L2}(d_{ego}(\hat{\mathbf{z}}_{0}^{t}), \mathbf{p}^{t})), \tag{6.20}$$

where T and M_0 are the number of frames and the number of spatial tokens at the original scale, respectively. $\mathbf{C}(\cdot)$ denotes the index of the corresponding code in the codebook $\mathbf{C}$. L_{L2} measures the L2 difference between the two trajectories.

6.3 Experiments

In this section we evaluate the performance of RenderWorld using NuScenes [17] dataset. We also performed extensive ablation experiments on the same dataset - as reported in Section 6.3.3 to deeper understand the proposed approach.

6.3.1 Experimental Setup

We adopt NuScenes as our evaluation dataset. NuScenes is a large-scale autonomous driving dataset that includes 700 scenes for training, 150 scenes for validation, and 150 scenes for testing, totaling approximately 40,000 frames across 17 classes. For self-supervised training, we generate ground truth depths and 2D segmentation ground truths by projecting LiDAR point clouds with their 3D segmentation labels onto corresponding 2D views. During the semantic occupancy prediction, each sample covers a range of [x:(-40 m, 40 m), y:(-40 m, 40 m), z:(-1.0 m, 5.4 m)] with a voxel size of 0.4 m. The evaluation experiments of our model are conducted on the 150 validation sets with one NVIDIA A30 GPU.

6.3.2 Main Result

3D semantic occupancy prediction: To demonstrate the performance of our model, we compare it against 10 occupancy prediction models, which are the existing common models evaluated on the NuScenes dataset. The results in Table 6.1 indicate that RenderWorld outperforms most state-of-the-art occupancy prediction methods in mIoU, ranking second overall, and only surpassed by CTF-OCC [257], which uses 3D occupancy GT as input. Furthermore, our method achieves outstanding performance in vehicle segmentation, including trailers, construction vehicles, trucks, etc and surpasses

Table 6.1: 3D Occupancy prediction performance on the Occ3D-nuScenes validation set.

Methods	GT	mIoU ↑	Others	barrier	bicycle	bus	car	cons. veh	motorcycle	pedestrian	traffic cone	trailer	truck	dri. sur	other flat	sidewalk	terrain	manmade	vegetation
TPVFormer [94]	3D	27.83	7.22	38.90	13.67	**40.78**	**45.90**	**17.23**	19.99	18.85	14.30	**26.69**	**34.17**	55.65	35.47	37.55	30.70	19.40	16.78
BEVFormer [140]	3D	26.88	5.03	38.79	9.98	34.41	41.09	13.24	16.50	18.15	17.83	18.66	27.70	48.95	27.73	29.08	25.38	15.41	14.46
OccFormer [355]	3D	21.93	5.94	30.29	12.32	34.40	39.17	14.44	16.45	17.22	9.27	13.90	26.36	50.99	30.96	34.66	22.73	6.76	6.97
CTF-Occ [257]	3D	**28.53**	**8.09**	**39.33**	**20.56**	38.29	42.24	16.93	**24.52**	**22.72**	**21.05**	22.98	31.11	53.33	33.84	37.98	33.23	20.79	18.0
RenderOcc [192]	2D	23.93	5.69	27.56	14.36	19.91	20.56	11.96	12.42	12.14	14.34	20.81	18.94	68.85	33.35	42.01	43.94	17.36	22.61
GaussianOcc [61]	2D	9.94	-	1.79	5.82	14.58	13.55	1.30	2.82	7.95	9.76	0.56	9.61	44.59	-	20.10	17.58	8.61	10.29
OccNeRF [349]	2D	9.53	-	0.83	0.82	5.13	12.49	3.50	0.23	3.10	1.84	0.52	3.90	52.62	-	20.81	24.75	18.45	13.19
SelfOcc [93]	2D	9.30	0.00	0.15	0.66	5.46	12.54	0.00	0.80	2.10	0.00	0.00	8.25	55.49	0.00	26.30	26.54	14.22	5.60
RenderWorld (Ours)	2D	27.87	6.83	32.54	7.44	21.15	29.92	16.68	11.43	17.45	16.48	24.02	27.86	**75.05**	**36.82**	**50.12**	**53.04**	**22.75**	**24.23**

Method	Input	Aux. Sup.	mIoU ↑				IoU ↑				Memory
			1s	2s	3s	Avg.	1s	2s	3s	Avg.	
Copy&Paste	3D-Occ	None	14.91	10.54	8.52	11.33	24.47	19.77	17.31	20.52	–
OccWorld (Original) [361]	3D-Occ	None	25.78	15.14	10.51	17.14	34.63	25.07	20.18	26.63	13500M
RenderWorld (Ours)	3D-Occ	None	**28.69**	**18.89**	**14.83**	**20.80**	**37.74**	**28.41**	**24.08**	**30.08**	**13000M**
TPVFormer [94]+LiDAR+OccWorld-T [361]	Camera	Semantic LiDAR	**4.68**	**3.36**	**2.63**	**3.56**	9.32	8.23	7.47	8.34	15000M
TPVFormer [94]+SelfOcc [93]+OccWorld-S [361]	Camera	None	0.28	0.26	0.24	0.26	5.05	5.01	4.95	5.00	15000M
RenderWorld (Ours)	Camera	None	2.83	2.55	2.37	2.58	**14.61**	**13.61**	**12.98**	**13.73**	**14400M**

Table 6.2: 4D occupancy forecasting performance. Aux. Sup. denotes auxiliary supervision beyond the ego trajectory. Avg. indicates the average performance over 1s, 2s, and 3s horizons.

Method	Input	Aux. Sup.	L2 (m) ↓				Collision Rate (%) ↓			
			1s	2s	3s	Avg.	1s	2s	3s	Avg.
IL [211]	LiDAR	None	0.44	1.15	2.47	1.35	0.08	0.27	1.95	0.77
NMP [348]	LiDAR	Box & Motion	0.53	1.25	2.67	1.48	**0.04**	0.12	0.87	0.34
FF [80]	LiDAR	Freespace	0.55	1.20	2.54	1.43	0.06	0.17	1.07	0.43
EO [113]	LiDAR	Freespace	0.67	1.36	2.78	1.60	**0.04**	**0.09**	0.88	0.33
ST-P3 [81]	Camera	Map & Box & Depth	1.33	2.11	2.90	2.11	0.23	0.62	1.27	0.71
UniAD [84]	Camera	Map & Box & Motion & Tracklets & Occ	0.48	0.96	**1.65**	**1.03**	0.05	0.17	**0.71**	**0.31**
VAD-Tiny [102]	Camera	Map & Box & Motion	0.60	1.23	2.06	1.30	0.31	0.53	1.33	0.72
VAD-Base [102]	Camera	Map & Box & Motion	0.54	1.15	1.98	1.22	**0.04**	0.39	1.17	0.53
OccNet [259]	Camera	3D-Occ & Map & Box	1.29	2.13	2.99	2.14	0.21	0.59	1.37	0.72
OccWorld-T [361]	Camera	Semantic LiDAR	0.54	1.36	2.66	1.52	0.12	0.40	1.59	0.70
OccWorld-S [361]	Camera	None	0.67	1.69	3.13	1.83	0.19	1.28	4.59	2.02
RenderWorld (Ours)	Camera	None	0.48	1.30	2.67	1.48	0.14	0.55	2.23	0.97
OccNet [259]	3D-Occ	Map & Box	1.29	2.31	2.98	2.25	0.20	0.56	1.30	0.69
OccWorld [361]	3D-Occ	None	0.43	1.08	1.99	1.17	0.07	0.38	1.35	0.60
RenderWorld (Ours)	3D-Occ	None	**0.35**	**0.91**	1.84	**1.03**	0.05	0.40	1.39	0.61

Table 6.3: Motion planning performance. Aux.Sup.denotes auxiliary supervision apart from the ego trajectory.

all other methods in segmenting various environmental terrains, such as vegetation, sidewalk etc. This is due to the 3D Gaussian representation, which effectively leverages the sparsity and object diversity in driving scenes, scaling with flexible location and covariance properties [95].

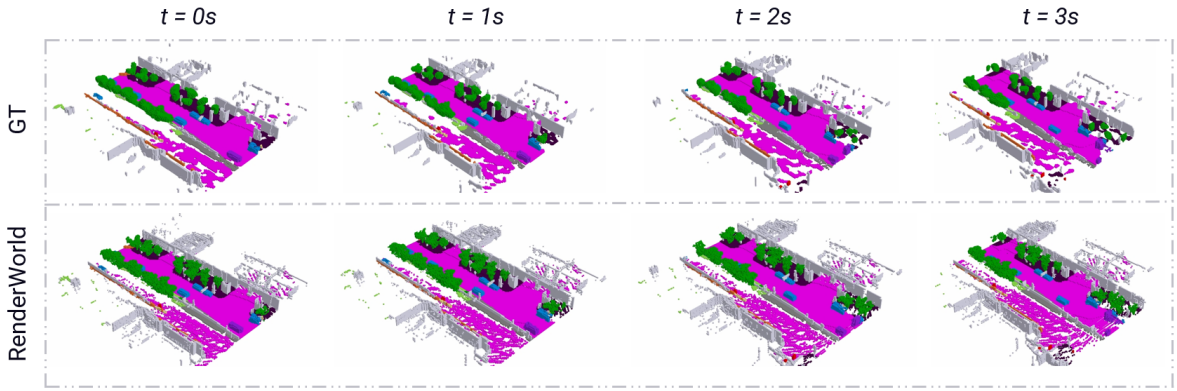


Figure 6.3: Visualization of the forecasting and planning results of RenderWorld.

4D occupancy forecasting: We evaluated the 4D occupancy forecasting performance under several settings as shown in Table 6.2

To capture finer-grained scene features and provide precise information for predictions, the air-separation technique is applied to prioritize crucial non-air components in the scene, boosting prediction accuracy and computational efficiency. The results

Setting	Forecasting mIoU (%) ↑				Planning L2 (m) ↓			
	1s	2s	3s	Avg.	1s	2s	3s	Avg.
(50^2, 128, 512)	**28.69**	**18.89**	**14.83**	**20.80**	**0.35**	0.91	1.84	1.03
(50^2, 128, 256)	27.16	18.09	14.45	19.90	**0.35**	**0.87**	**1.81**	**1.01**
(50^2, 128, 1024)	26.34	18.37	14.97	19.89	0.39	1.05	2.16	1.20
(25^2, 256, 512)	15.15	12.01	9.56	12.24	3.21	5.98	8.92	6.04
(100^2, 128, 512)	21.68	15.07	11.67	16.14	0.45	1.29	2.28	1.34

Table 6.4: Effect of different hyperparameters for the scene tokenizer. The setting denotes the latent spatial resolution, latent channel dimension, and codebook size, respectively.

show that RenderWorld can generate non-trivial future 3D occupancy, with results far superior to OccWorld and Copy&Paste, which indicates that our model learns the underlying scene evolution.

Motion planning: As shown in Table 6.3, we compare the motion planning performance between the proposed RenderWorld and state-of-the-art methods, and evaluate our model across various settings used in the 4D occupancy forecasting task. RenderWorld outperforms all compared methods in L2 metrics when taking 3D occupancy as input. Without any auxiliary support, our approach also achieves competitive results in collision rate and even outperforms OccWorld-S when it only uses 2D as input.

6.3.3 Ablation Study

To show the effectiveness of our innovative modules, we conduct three ablation studies and the results are shown in Tables 6.5, 6.4, and Table 6.6:

Efficiency comparisons among different representations: In Table 6.5, we present the efficiency comparisons of various representations, highlighting that 3D Gaussian surpasses all competitors with significantly reduced memory usage. Leveraging its explicit representation, this approach assigns specific semantic data to individual 3D Gaussians, facilitating the transition from scene depiction to occupancy forecasts. This method also circumvents the high memory usage linked to the ray initialization step in NeRF-based techniques. Although our method has higher GPU memory overhead compared to GaussianFormer, it avoids the trade-off of reducing the number of Gaussian points to save memory, but leading to a loss of semantic information.

Analysis of the scene tokenizer. Table 6.4 demonstrates the impact of different hyperparameter settings on the performance of the scene tokenizer; our parameters are

Method	Query Form	Query Resolution	Memory (MB)
BEVFormer [140]	2D BEV	200×200	25100
TPVFormer [94]	2D Tri-Plane	$200\times$ $(200 + 16 + 16)$	29000
PanoOcc [286]	3D Voxel	$100 \times 100 \times 16$	35000
Fb-Occ [141]	3D Voxel + 2D BEV	$200 \times 200 \times 16$ + 200×200	31000
OctreeOcc [170]	Octree Query	91200	26500
OccNeRF [349]	3D Voxel	$200 \times 200 \times 16$	79000
RenderOcc [192]	3D Voxel	$200 \times 200 \times 16$	23000
GaussianFormer [95]	3D Gaussian	144000	**6229**
RenderWorld (Ours)	3D Gaussian	640000	14400

Table 6.5: Efficiency comparison on the nuScenes dataset. 3D Gaussian–based representations significantly reduce memory usage compared to voxel- and BEV-based methods.

designed like OccWorld. Larger spatial resolutions can enhance reconstruction accuracy but hinder prediction and planning, because limited token capacity for learning high-level concepts complicates future predictions [361]. Codebook sizes exceeding 512 lead to overfitting, while smaller sizes or resolutions compromise scene representation accuracy.

Effects of mask module and VAE separation operation. Table 6.6 presents the ablation study about our AM-VAE module (Separate VAE refers to dividing the potential space of the VAE into air and non-air portions). The introduction of the Air-Mask module leads to a performance improvement, achieving an mIoU of 37.68%. When applying both the Mask module and VAE separation operation together, the performance can noticeably reach 40.25%. This indicates that our proposed Mask and VAE Separation operation offers substantial advantages in enhancing model reconstruction accuracy and reducing positional errors. Overall, the ablation study underscores the effectiveness of the proposed enhancements, especially the air separation strategy, in substantially boosting the performance of the RenderWorld framework.

6.4 Conclusions

In this chapter, we introduce RenderWorld, an end-to-end autonomous driving framework trained on 3D occupancy labels generated by a Gaussian-based Img2Occ module and use a world model for forecasting and motion planning. By leveraging Gaussian

Air-Mask	Separate VAE	mIoU
-	-	35.13
√	-	37.68
-	√	35.42
√	√	**40.25**

Table 6.6: Ablation studies of air mask and vae separation. Each value indicates the performance on the validation dataset.

Splatting and AM-VAE, we successfully reduce GPU memory usage by at least half in 3D occupancy label generation compared to NeRF-based approaches, while simultaneously attaining minimal memory requirements in 4D occupancy forecasting. Our work offers a valuable contribution to the autonomous driving community, enhancing real-time, efficient robot perception, forecasting, and motion planning.

6.5 Limitations and Future Works

While RenderWorld demonstrates the effectiveness of Gaussian-based 3D occupancy generation for real-time autonomous driving perception and planning, some limitations remain that warrant further investigation. The current system relies primarily on multi-view camera inputs, which can struggle in highly dynamic or occluded scenarios, and the absence of complementary sensors such as LiDAR or radar may reduce robustness under adverse weather or lighting conditions. Although Gaussian Splatting significantly reduces memory requirements compared to NeRF-based approaches, extremely large-scale urban scenes with high voxel resolutions can still incur substantial computational costs. Moreover, while our framework supports short- to mid-term motion forecasting, maintaining temporal consistency over longer horizons, particularly in complex traffic environments, remains challenging. In future work, we plan to evaluate the system under diverse weather and lighting conditions, extend it to long-term motion forecasting tasks, and address these limitations to advance RenderWorld toward more scalable, robust, and intelligent 3D world modeling suitable for real-world deployment.

6.6 Summary and Relation to Research Question

This chapter addresses **Research Question 3**, which investigates how 3D Gaussian Splatting can be extended into an efficient world representation for large-scale dynamic scene understanding and forecasting. The chapter demonstrates that 3DGS can serve as

an effective intermediate representation by bridging 2D perception and structured 3D occupancy through the proposed Img2Occ module. By leveraging the continuous and multi-view consistent nature of Gaussian Splatting, the proposed approach enables the generation of geometrically faithful and semantically enriched occupancy representations from sparse multi-view inputs which shows that 3DGS can be effectively extended beyond static reconstruction, serving as a scalable and efficient representation for dynamic scene understanding and world modeling, thereby providing a concrete answer to Research Question 3.

Chapter 7

Indoor Scene Understanding and Editing with 3D Gaussian Splatting

7.1 Introduction

The creation and editing of 3D scenes have been both costly and time-consuming. Designers have to manually work with various 3D tools, investing considerable time and effort in tasks like sketching, designing layouts, arranging objects, and selecting material textures [278]. However, the recent emergence of generative AI has revolutionized these processes, allowing the creation of high-quality 3D assets faster and more affordable. Using text-to-3D [200, 90, 129, 148, 30, 207, 143, 162, 293, 34, 126, 337, 333, 332] or image-to-3D [153, 152, 75, 151, 166, 326, 154, 252, 253] methods, users can now quickly generate or re-layout detailed 3D scenes using text prompts or images. As a result, AI-driven generation techniques have gradually gained widespread popularity across industries like advertising, animation, game development, and VR/AR.

Before the development of Gaussian Splatting [110], NeRF-based methods [241, 121, 52, 100, 236, 7, 342] dominate the field of 3D editing due to their powerful 3D scene representation capabilities [314, 213, 315]. These methods typically rely on pre-trained NeRF models to edit 3D scenes. A notable example is Instruct-NeRF2NeRF [73], which uses an image-conditional 2D diffusion model called InstructPix2Pix [16], for 3D scene editing. However, NeRF's dependence on high-dimensional multilayer perceptron (MLP) networks for encoding scene data limits its ability to directly modify specific scene elements and complicated tasks such as inpainting and scene composition [31]. Additionally, NeRF's implicit representation and resource-demanding issues pose a significant challenge for interactive-rate editing.

The emergence of 3D Gaussian Splatting has revolutionized both 3D reconstruction and image rendering, with significant impacts on 3D editing. 3D Gaussian Splatting [110] is a pioneering technique that achieves real-time rendering while maintaining high-quality outputs with fast training speed. Its explicit representation offers distinct advantages for editing, as each 3D Gaussian is individually manipulable, allowing for direct and efficient scene modifications. This innovation has inspired the creation of several 3D editing methods based on Gaussian Splatting, such as Instruct-GS2GS [267], GaussianEditor [31, 278], etc., which are built on InstructPix2Pix [16]. However, these editing approaches, fully based on diffusion models, often lack detailed control over scene modifications and are limited by input image resolution. For example, InstructPix2Pix, built on stable diffusion [219], primarily supports 512x512 or 768x768 px images, and deviations from these resolutions can significantly impact the quality of the output [182, 344, 343].

Since controllable scene editing in complex layouts using pre-trained generative models is highly challenging, current methods [328, 21, 232, 38, 82], largely based on Gaussian Splatting, rely on models like Grounding Dino [156] and SAM [115] to detect and segment objects in each 2D image before projecting features into 3D space. This 2D-based approach complicates the process, requiring users to render multiple 2D images from a 3D model, segment and ground each object, and project them frame-by-frame into 3D space. Therefore, achieving interactive-rate, user-friendly, and controllable editing would mark a significant breakthrough.

Boosted by these facts, we propose an editing paradigm shift with a 3D-only approach, named **3DSceneEditor**, for controllable editing of complex scenes based on Gaussian Splatting. Input only one prompt, 3DSceneEditor can achieve precise edits within seconds. The key to achieve interactive-rate editing is our fully 3D pipeline, which allows direct manipulation of Gaussians in a single step. We first use a pre-trained instance segmentation model from Mask3D [227] to assign semantic labels to each Gaussian. Next, we ground target instances using a zero-shot grounding algorithm and employ CLIP [206] to align target objects with the input prompt and the desired edits. Finally, the editing operations are applied directly to the Gaussians, and the entire process can be completed in just tens of seconds. To handle potential mis-segmentations from Mask3D, we use KNN to correct outliers through voting. Table 7.1 and experimental results show that our pipeline outperforms current SOTA approaches in editing quality and options, processing time and GPU usage.

In summary, the primary contributions of the proposed 3DSceneEditor [312] are:

1) A 3D-only editing approach, named **3DSceneEditor**, for complex scenarios: Unlike previous multi-step methods that always rely on 2D-3D semantic projection, our framework simplifies the process, enabling interactive-rate editing, higher quality results, and improved user interaction.

2) An innovative zero-shot instance grounding pipeline for precise grounding of target objects in complex 3D layouts, which is achieved through LLM-based keyword extraction, view-based relationships simplified with a 2D egocentric approach, and language-object correlation using a multimodal language model.

3) A controllable scene editing method enabling object addition, movement, recoloring, removal, and replacement through text-based instructions, followed with an optimization module for precise and efficient direct 3D manipulation.

Editing Capability	Removal	Addition	Recoloring	Stylization	Repositioning	Replacement	3D-only
GaussianEditor [31]	✓	✓	✓	✓			
GaussianEditor [278]	✓		✓	✓			
GaussianCtrl [296]			✓	✓			
DGE [27]			✓	✓			
3DEgo [111]	✓		✓	✓			
GaussianGrouping [328]	✓		✓		✓		
Ours	✓	✓	✓		✓	✓	✓

Table 7.1: Comparison of SOTA editing methods and our proposed approach: 3DSceneEditor offers more diverse editing capabilities and is currently the first pipeline that supports 3D-only editing.

7.2 Methodology

First, we provide an overview of our proposed 3D-only approach (Section 7.2.1), followed by an introduction to our Open-Vocabulary Object Grounding module, which uses a view-dependent module and multimodal alignment assistant (CLIP) (Section 7.2.2). Finally, Section 7.2.3 covers the 3D editing operations and optimization module.

7.2.1 Overall Framework

3D Gaussian Splatting [110] is an innovative approach that represents a 3D scene explicitly as a set of Gaussians $\{G_x\}_{x=1}^{N}$. Our editing pipeline (Fig. 7.1) starts from a set of 3D Gaussians G_{in} trained from a specific scene and a prompt τ. These Gaussians are processed through a pretrained instance segmentation model that assigns semantic labels to each Gaussian. Next, target objects and their references are identified based on the

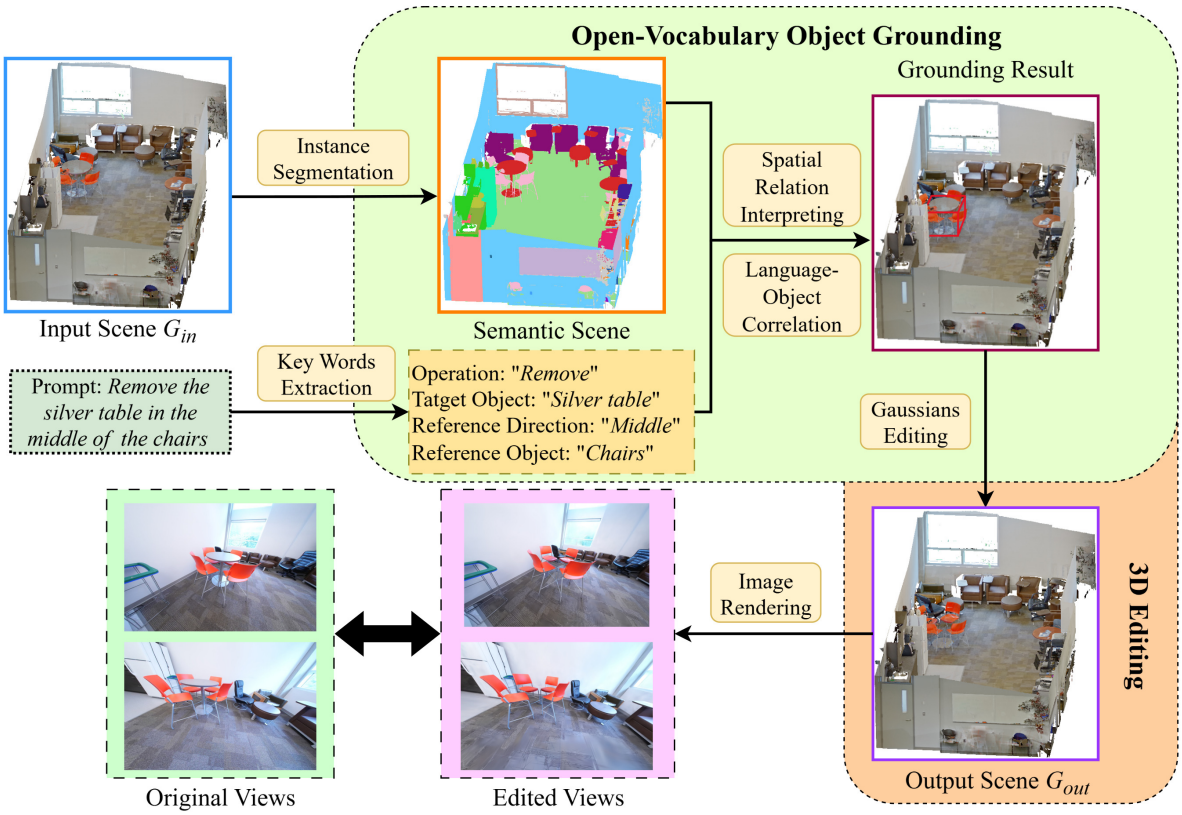


Figure 7.1: Our 3D-only paradigm, named **3DSceneEditor**, consists of three key steps. First, a pre-trained instance segmentation model is applied to understand the input scene and assign a semantic label to each Gaussian. Followed by an Open Vocabulary Object Grounding module to ground the target objects from the input semantic Gaussians and generate the ROI for target objects. Finally, we edit the specified scene in ROI based on the prompt and render the edited views.

keywords extracted from the prompt, and their Region of Interest (ROI) is determined using an Open-Vocabulary Object Grounding module (Section 7.2.2). With this information and editing directives from the prompt, our pipeline enables interactive-rate scene editing through direct manipulations of Gaussians within the ROI, supporting operations such as object addition, movement, removal, replacement, and colorization (Section 7.2.3). Thus, our editing task can be defined as:

$$G_{out} = Edit(G_{in}, \tau). \tag{7.1}$$

7.2.2 Open-Vocabulary Object Grounding



Figure 7.2: Visualization of our Object Grounding. We first extract the key words from the prompt (**bold fonts** in the figure). Since the positional relationships between objects in 3D space change in different viewpoints, we need to project them onto a static 2D plane to better understand the scene.

3D Objects Grounding is an essential part of our 3D-only editing pipeline. Prior approaches, like Gaussian grouping [328] and FlashSplat [232], extract frame-level features using pre-trained 2D detection and segmentation models, then project semantic labels from 2D images into 3D space, dividing Gaussians into instance-based groups. However, these methods neglect the inherent spatial relationships within 3D Gaussians, posing major limitations for complex editing tasks. For instance, similar or identical objects are common in 3D scenes, especially indoors. Even though groups of Gaussians allow for object removal or recoloring by directly operating on Gaussian clusters, performing more complex interactions, such as adding objects or swapping their positions, remains extremely challenging. Additionally, these pipelines require users to know in advance which instances each Gaussian group represents, rather than enabling direct localization of target objects through interactive prompts. To address these challenges, we introduce an open-vocabulary 3D grounding module as shown in Fig. 7.2.

Key words extraction. For editing a given 3D scene based on a specific prompt,

we first use GPT-4V to extract and classify key elements, including object queries (e.g., *coffee table*, *monitor*), scene-editing actions (e.g., *remove*, *recolor*), view-dependent descriptors (e.g., *left*, *between*), and color mappings. Leveraging the semantic information of the Gaussians, we then filter and identify candidate objects that match the specified categories.

Spatial relation interpreting. In complex 3D scenes, distinguishing objects of the same category poses a significant challenge. Our 3D grounding module tackles this by interpreting spatial relationships for each candidate object. We leverage the 2D egocentric, view-related module proposed by [346] to simulate a virtual camera at the scene center, projecting complex geometric relationships between target and reference objects from 3D space onto a 2D plane. This enables pixel-level filtering of candidate objects based on view-dependent relations (e.g., *identifying which trash can is to the right of the refrigerator*). Additionally, the 3D bbox of selected objects are projected to mitigate occlusion issues. For more details, please refer to supplementary material.

Language-Object correlation. Finally, an Image-Text Alignment module is applied [206] to evaluate the cosine similarity between the prompt and the tokenized image query, finding the optimal candidate target objects and returning their 3D bounding boxes as ROI, which is curial for the following 3D Gaussian editing.

7.2.3 3D Gaussian Editing

The proposed pipeline initiates the 3D editing operation based on prompt instructions, applying editing only to the Gaussians located within the specified ROI, as reported in Section 7.2.3. It can support totally 5 types operations: object removal, recoloring, object addition, object movement and object replacement.

Object removal and recoloring. Our approach can easily achieve object removal and recoloring by either removing Gaussians or directly changing their color feature with the target semantic labels. To facilitate prompt-based recoloring, we first construct a color mapping table with common colors and map the color keyword from prompt to color table directly.

Object addition and replacement. Other image-based methods [296, 351, 31] primarily rely on 2D diffusion priors and novel view synthesis to add or replace objects in scenes. In contrast, our pipeline achieves object integration directly in 3D space by generating new objects from prompts or images using a Gaussian-based generative model [251]. We then incorporate these objects by adding the new Gaussians or substituting them within ROI, as shown in Fig. 7.3.

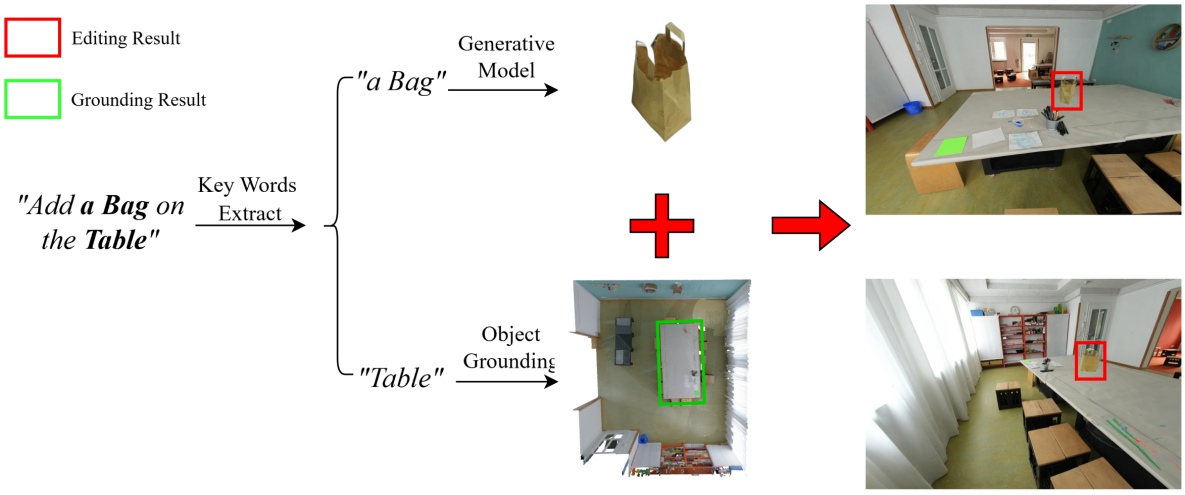


Figure 7.3: Visualization of our object addition pipeline. We generate new objects using a Gaussian-based generative model, guided by keywords extracted from the prompt. With the assistance of the Object Grounding module, these new Gaussians are then integrated into the ROI within the input scene.

Since the size of AI-generated objects can be unpredictable, we first apply an adjustable scaling parameter to match their size to reference objects. We then position the Gaussians of new objects into target regions or replace existing objects with the newly generated Gaussians. For object addition, our method aligns the central axes of the external bounding boxes of both the new and reference objects, ensuring their corresponding bounding box sides overlap based on the target view-dependent relationship.

This geometry-based stitching technique effectively minimizes prediction errors, commonly encountered in diffusion-prior-based methods, making it highly applicable to a wide range of 3D scenes with complex layouts.

Object movement. To achieve object movement, we select the valid Gaussians from their semantic labels and lightly adjust their coordinates (x_in, y_in, z_in) in the world coordinate system based on their reference objects and text instruction (***close***, ***far away***). Moving 3D Gaussians are inherently complicated. Thus, each Gaussian not only represents one object attribute [232]. Since each 3D Gaussian is projected onto a 2D plane via orthographic projection, the Gaussian's covariance in ray space is derived by applying a series of transformations to the Gaussian's covariance matrix Σ and its center (x_0, y_0, z_0) in world coordinate. Moving a 3D Gaussian can affect other objects along the same ray in ray space, resulting in extra noise and artifacts in the projected 2D image, which impact different objects across various viewpoints. As a result, our pipeline currently only supports moving small objects within a limited range

to avoid displacing too many Gaussians at once, which could disrupt scene rendering significantly.

Optimization of editing. Pre-trained instance segmentation models may struggle in specialized scenarios (e.g., objects near wall-floor junctions). To address this, we refine Gaussian semantics by applying K-Nearest Neighbors (KNN) clustering to all Gaussians within the ROI based on their positional features prior to editing. Artifacts or "black holes" emerging post-removal are mitigated by reapplying KNN to inpaint background regions using Gaussian features from neighboring points (Section 7.3.4).

For 3D generation, we employ LGM [251] as the base model, enhanced by a adaptive optimization framework. As shown in Fig. 7.4, our pipeline first generates multiple 3D models in parallel from text prompts. A pre-trained aesthetic predictor[1] (threshold: 4.8) selects the optimal candidate, with subthreshold cases refined through GPT-4V-redescribed prompts and seed-guided regeneration. Moreover, an optional object-relighting [197] module is designed to align object-scene lighting (15–20 min runtime; more details are provided in the supplementary material.).

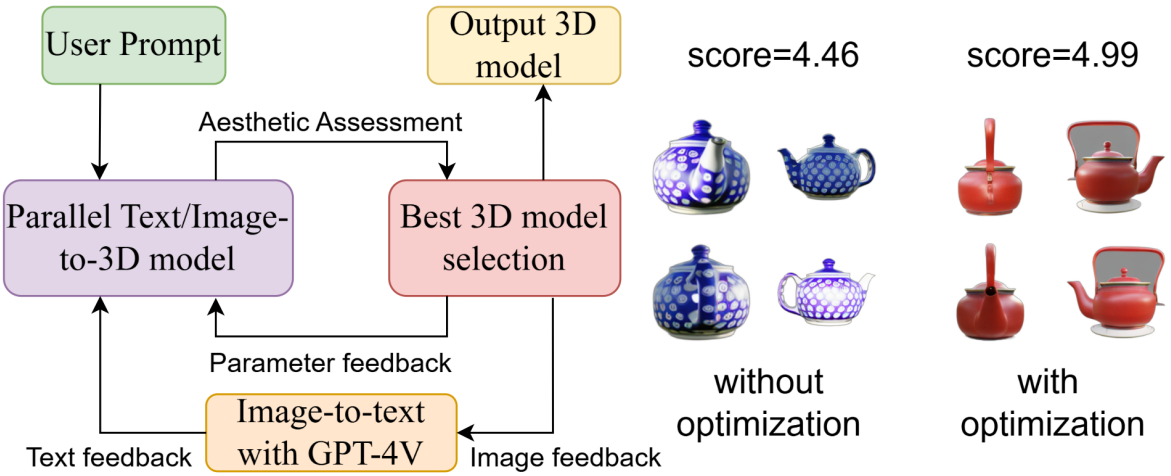


Figure 7.4: Optimization mechanism of object addition and replacement.

7.3 Experiments

7.3.1 Implementation Details

Our method is implemented in PyTorch and CUDA, with all 3D Gaussians trained using the original 3D Gaussian Splatting [110] and LGM [251]. Experiments were conducted

[1]https://github.com/christophschuhmann/improved-aesthetic-predictor

on a single NVIDIA Tesla A100 GPU from ScanNet++ [330] and Mip-NeRF 360 [8] datasets, including a variety of indoor and outdoor scenes, with prompts customized for each scene layout. Since ScanNet++ images are captured by a fisheye digital SLR camera, which is incompatible with 3D Gaussian Splatting, we utilize the ScanNet++ Toolkit [330] to undistort the fisheye images and convert the camera model to a pinhole model with COLMAP [226]. To support a variety of editing applications, we use a pre-trained instance segmentation model from ScanNet200 [221] and OpenIns3D [96] for indoor and outdoor scenes respectively to obtain Gaussian semantics. For more implementation details, please refer to supplementary material.



Figure 7.5: Results of 3DSceneEditor. This figure presents additional results across diverse scenes, demonstrating that our method enables precise and varied scene editing for layouts and objects of different scales.

7.3.2 Qualitative Evaluation

Visualization results of different scenes. Fig. 7.5 present visual results from 3DSceneEditor, demonstrating its capability for precise, controllable, and 3D-consistent editing. In Fig. 1.3, our pipeline represents various editing operations on individual objects within a 3D scene. The left side of the figure shows original scenes and the object grounding results, where the stool is anchored to the left of the table. The right side illustrates different edits applied to the stool using varied text instructions. By

leveraging Gaussian 3D spatial information and memorizing ROI, our method minimizes the resources consumed in repeated semantic reasoning and grounding of the same scene, significantly reducing secondary edit time to just seconds. Fig. 7.5 further showcases these capabilities across diverse 3D scenes. In the first two columns, we demonstrate object removal (a black chair) and color modification (an office chair). Despite multiple identical chairs in these scenes, our pipeline accurately grounds and edits objects based on prompts. While the middle column showcases our method's precision in adding objects via interacting with generative models. The fourth and fifth columns highlight movement and replacement of small objects within complex scenes; even with numerous small items, our approach maintains precise object grounding and editing, delivering robust editing capabilities across complex scenes and varied object types.

Comparisons with other Gaussian-based controllable editing approaches. Fig. 7.6 reports a comparison of the performances of the proposed method across the five supported editing capabilities, benchmarked against GaussCtrl [296], GaussianEditor [31] and DGE [27]. Since all of them are based on Instruct-Pix2Pix, which performs best at a resolution of 512x512 px. Thus, we resized their input images to that resolution and apply the same edit operations supported by our pipeline to all methods. The results demonstrate that:

(i) Our 3D-only pipeline uniquely provides stable, high-quality, and controllable edits in complex scenes, highlighting its distinct advantages over compared approaches. In contrast, the images generated by GaussCtrl and DGE show significant degradation in tasks involving object addition, recoloration movement, and replacement. This occurs because their pipelines process each frame individually with Instruct-Pix2Pix, which fails to maintain the original scene style and produces disrupted Gaussian features. While GaussianEditor uses Gaussian semantic tracking to define editable areas within the ROI, limitations in the diffusion model and semantic projection precision hinder its ability to deliver high-quality editing in complicated scenes. To further investigate this issue, we additionally process the same images directly with Instruct-Pix2Pix and show them also in Fig. 7.6 with other approaches together. These outcomes further validate our previous observations.

(ii) Our approach uniquely supports object movement and replacement, as none of the compared methods effectively utilize the 3D spatial information of Gaussians. This demonstrates our pipeline's ability to offer a wider range of editing operations.

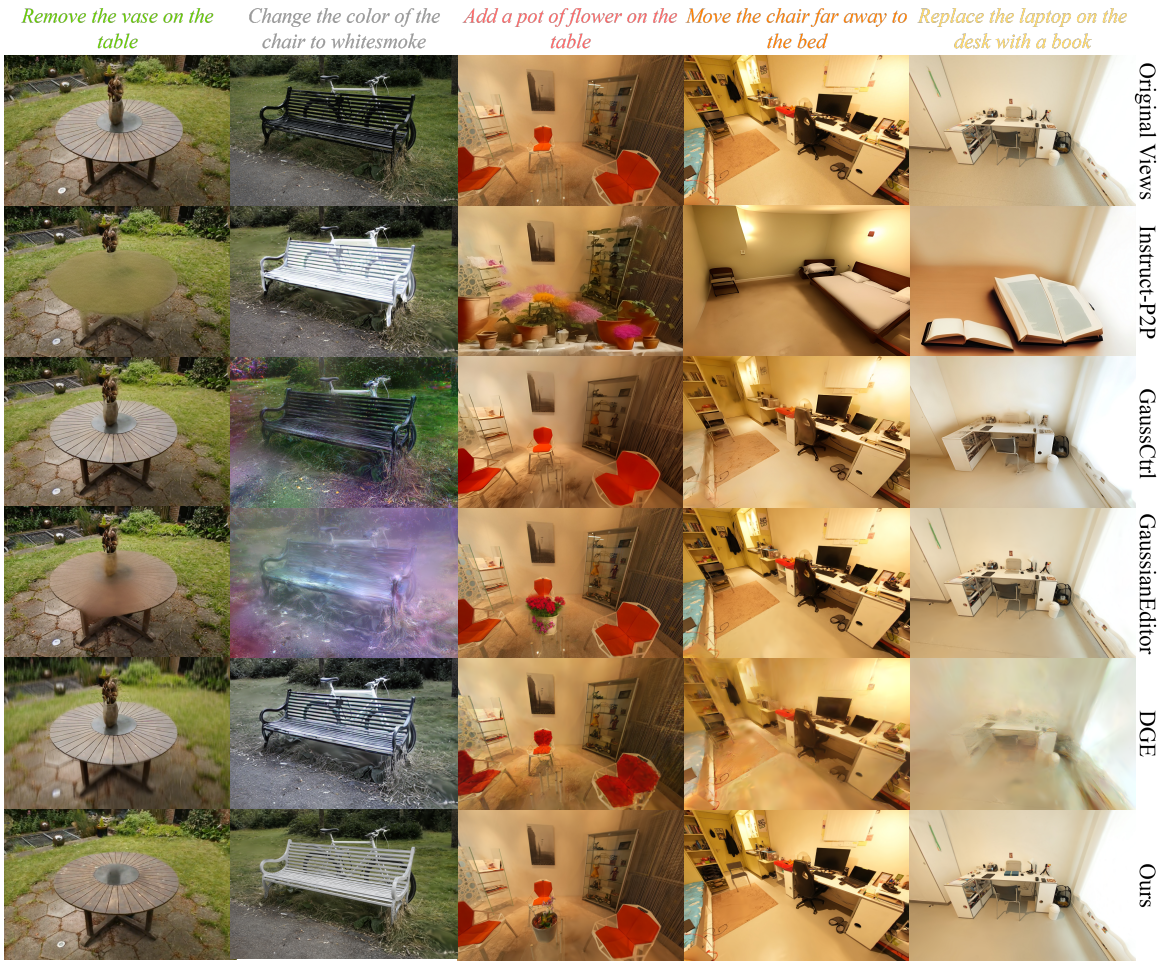


Figure 7.6: Qualitative comparison with respect to other controllable scene editing approaches. The experimental results in ScanNet++ and Mip-NeRF 360 datasets show that in both indoor/outdoor scene layouts, our 3D-only approach effectively utilizes 3D Gaussian spatial information, maintaining high editing quality independent of input image pixels and consistently preserving scene style, enabling more reliable and controllable editing in intricate settings.

7.3.3 Quantitative Evaluation

Table 7.2 presents quantitative comparisons among all evaluated approaches, including ours. We apply two CLIP metrics (incldue Text-Image Similarity (TIS) and Image-Image Similarity (IIS)), two computational metrics (incldue Running Time and video RAM (VRAM) usage), and user stduy (incldue Text Alignment and Editing Quality, more details please refer to the supplementary material.) as measured based on the

Method	CLIP Metrics ↑		Computational Consuming ↓		User Study ↑	
	TIS	IIS	Running Time	VRAM	Prompt Alignment	Editing Quality
GaussCtrl[296]	22.16%	91.93%	8 min	24000MB	11.90%	2.38%
GaussianEditor[31]	23.26%	94.35%	8 min	12000MB	42.86%	7.14%
DGE[27]	23.12%	86.49%	4 min	10000MB	9.52%	0
Ours	**23.28%**	**96.17%**	**2–5 min**[2] **1 min**[3]	**9400MB**[2] **9100MB**[3]	**92.86%**	**90.48%**

Table 7.2: Quantitative Comparison. 3DSceneEditor outperforms existing methods in both CLIP metrics while requiring less time and GPU memory.

results in Fig. 7.6. 3DSceneEditor achieves the highest performance in both TIS and IIS, indicating that our 3D-only design can better preserve scene style consistency while effectively responding to text instructions. 3DSceneEditor consumes less GPU memory (VRAM) than other methods and completes the initial edit[2] in just 2-5 minutes, while secondary edits[3] take less than 1 minute, significantly outperforming other approaches and enabling interactive-rate scene editing. Furthermore, our method receives the highest user preference in both Text Alignment and Editing Quality, demonstrating that the edits are not only precise but also visually appealing to human evaluators.

7.3.4 Ablation Study and Analysis

Ablation of language-object correlation. Table 7.3 and Fig. 7.7 present the results and evaluation pipeline for assessing the scalability of our Language-Object Correlation module with different Vision-Language Models (VLMs). Our ablation study is conducted on 105 object pairs from 89 scenes, utilizing GroundingDino [155], with all pairs labeled by GPT-4V. Across various encoder combinations, our module consistently achieves high classification accuracy, demonstrating its flexibility and adaptability, enabling developers to interchange text and image encoders as needed.

[2]Initial edit: Initializing scene semantics with instance segmentation for object grounding and editing.

[3]Secondary edit: Utilizing saved instance semantics for object grounding and editing.

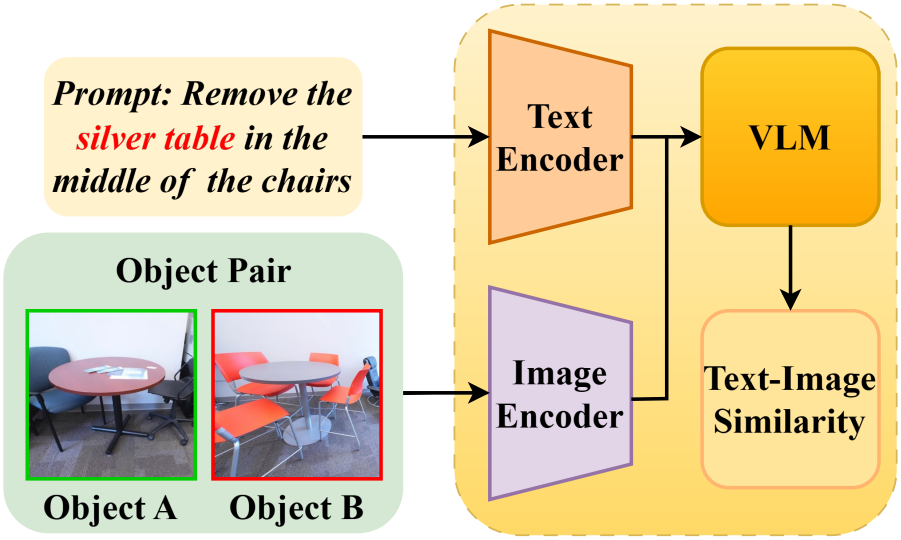


Figure 7.7: Visualization of Language-Object Correlation. The keyword in the prompt (red font) is extracted and tokenized by a text encoder, then input into the vision-language model (VLM) alongside the image token. The Text-Image Similarity (TIS) is then calculated from the VLM's output.

Encoder (Image-Text)	Distractor (%)	Target (%)	Accuracy (%)
CLIP-CLIP[206]	26.16	**28.99**	91.43
CLIP-Llama[260]	18.28	**18.51**	88.57
Blip2-Blip2[127]	30.14	**35.68**	94.29
CLIP-Qwen[280]	18.10	**18.87**	89.52
CLIP-BERT[48]	22.76	**23.67**	93.33
CLIP- Llama 2[261]	18.91	**19.22**	92.38

[1] Llama 2 is compressed to FP16/INT4 by GWQ [229]

Table 7.3: Ablation study of Language-Object Correlation. This table presents the average TIS for tested object pairs and the accuracy of different image-text encoder combinations. The object with the higher TIS is selected as the final choice.

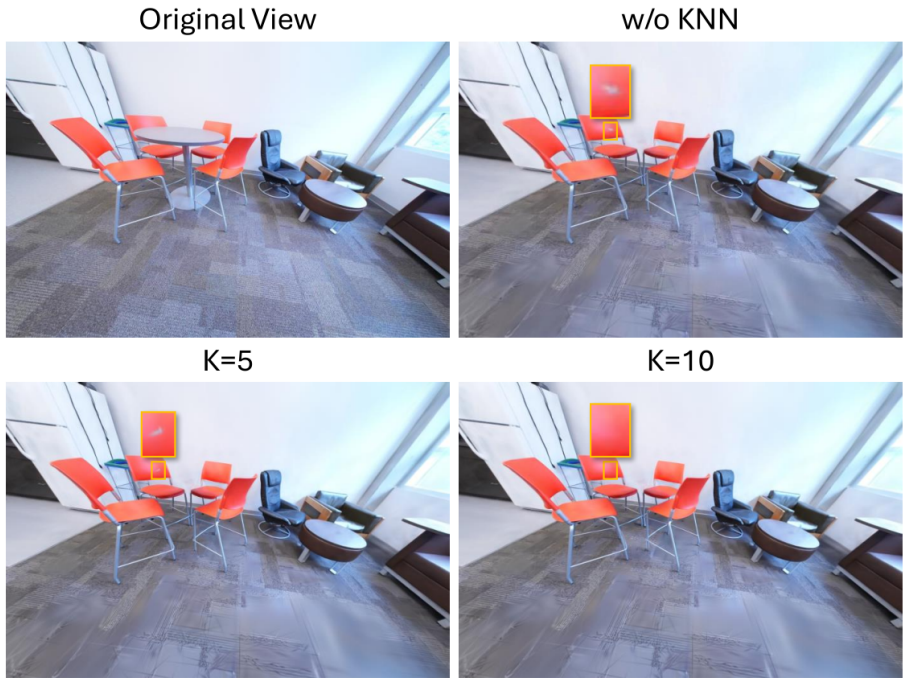


Figure 7.8: Visualization of KNN in our optimizion module. By using the KNN to refine the segmentation labels within ROI regions, we effectively reduces noise, and yielding more accurate editing.

Ablation of editing optimization. Fig. 7.8 reports ablation experiments on the Editing Optimization module. Without this module, edited images exhibit noise within the ROI, primarily due to segmentation errors near the junction between the chair and table. By adjusting the parameter K, our optimization module effectively reduces segmentation errors and enhances editing precision.

7.4 Conclusions

This paper introduced 3DSceneEditor, an innovative 3D-only paradigm for text-guided, precise scene editing. To our knowledge, 3DSceneEditor is the first fully 3D-based approach for editing 3D Gaussians, fully leveraging 3D spatial information in Gaussians to enhance both efficiency and accuracy. Key techniques include applying instance segmentation to 3D Gaussians, extracting key instructions from the prompt, grounding the ROI to 3D Gaussians with a zero-shot object grounding module, and editing the scene within the Gaussian ROI. Our experiments demonstrated that 3DSceneEditor outperformed GaussCtrl [296], GaussianEditor [31] and DGE [27] with higher TIS and IIS scores, greater user preference, reduced running time, and lower GPU memory

usage, validating its ability to achieve accurate, controllable, and interactive-rate scene editing.

7.5 Limitations and Future Works

Achieving physical plausibility after scene editing remains a key challenge in the 3D field. Currently, our method does not explicitly enforce physical constraints. This limitation is largely due to the limited reasoning capabilities of existing 3D-LLMs [354, 367], which are difficult to reliably evaluate the physical rationality of edits in a purely 3D manner.

In addition, our limitation of stylization is also linked to the current status of 3D foundation models: due to the limited scale of high-quality 3D data, there are not yet general-purpose models for direct 3D object editing. Instead, we focus on controllability and reproducibility, which are often lacking in 2D instruction-driven methods. Nevertheless, our pipeline is modular and can seamlessly integrate future 3D base models once they become available, ensuring adaptability and extensibility.

7.6 Summary and Relation to Research Question

This chapter addresses **Research Question 4**, which explores how 3DGS can serve as a unified and efficient representation for advanced scene understanding and downstream 3D scene manipulation. The chapter demonstrates that the explicit and point-based nature of 3DGS allows each Gaussian to carry semantic and instance-level information, enabling direct interaction with the scene at a fine-grained level. Building upon this property, the proposed 3DSceneEditor framework eliminates the need for intermediate 2D projections and multi-view consistency constraints, instead performing editing operations directly in 3D space. By integrating instance segmentation, zero-shot grounding, and language-guided editing within a unified 3D pipeline, our approach enables precise, controllable, and interactive-rate scene manipulation that bridges scene understanding and manipulation, supporting both semantic reasoning and direct geometry editing within a single framework, thereby providing a concrete answer to Research Question 4.

Chapter 8

Conclusions

8.1 Core Contributions

This thesis has made significant contributions across multiple facets of neural scene representation. A concise overview is as follows:

1) Development of a multi-scale benchmark dataset, named NeRFBK (https://github.com/3DOM-FBK/NeRFBK), for systematic evaluation of NeRF- and 3DGS-based 3D reconstruction methods, enabling fair comparisons, quantitative performance analysis, and future method development across diverse real and synthetic scenarios (Research Question 1, Chapter 3).

2) Systematic evaluation and analysis of learning-based 3D reconstruction methods across industrial and urban-scale settings, revealing their strengths, limitations, and practical applicability on non-collaborative surfaces and large-scale environments (Research Question 1, Chapter 4).

3) A GS-based surface reconstruction framework integrating relighting and surface normal supervision, significantly improving the reconstruction of reflective, glossy, and transparent non-collaborative objects (Research Question 2, Chapter 5).

4) RenderWorld, a self-supervised Gaussian-based 3D occupancy generation framework that produces 3D occupancy labels from 2D multi-view images, enhancing scene understanding, forecasting, and motion planning for autonomous driving applications (Research Question 3, Chapter 6).

5) 3DSceneEditor, a fully 3DGS-based scene editing framework that leverages the explicit nature of 3D Gaussians to assign semantic labels and perform direct,

precise, and real-time modifications of complex indoor scenes (Research Question 4, Chapter 7).

In detail, in Chapter 3, we introduced NeRFBK, a comprehensive benchmark dataset designed for systematic evaluation of NeRF- and 3D GS-based 3D reconstruction methods. NeRFBK integrates both real-world and synthetic scenes spanning multiple scales, surface properties, and application domains, including industrial inspection, transparent and reflective objects, cultural heritage, and aerial reconstruction. It provides high-resolution images and videos, accurate camera intrinsics and extrinsics, and reliable ground-truth geometry in the form of point clouds and mesh models. By addressing long-standing limitations in dataset completeness, surface diversity, and ground-truth accuracy, NeRFBK enables fair comparisons between view-synthesis-based and traditional image-based reconstruction pipelines. Its effectiveness and utility are demonstrated through extensive experimental evaluations, establishing it as a standardized benchmark for advancing high-fidelity 3D reconstruction research.

In Chapter 4, we conducted a systematic evaluation of learning-based 3D reconstruction methods across both industrial object inspection and urban-scale scene modeling. We analyzed NeRF-based methods, 3DGS, learning-based Multi-View Stereo, monocular depth estimation, and generative AI techniques under challenging conditions, such as texture-less, reflective, shiny, and transparent surfaces. For industrial scenarios, the evaluation focuses on non-collaborative objects, some drawn from NeRFBK, enabling controlled comparisons with accurate ground truth. For urban environments, we extended the analysis to large-scale outdoor reconstructions, considering challenges including sparse observations, occlusions, memory consumption, and computational overhead. Using standard photogrammetric metrics alongside large-scale evaluation criteria, we quantitatively assessed reconstruction accuracy, completeness, and precision across methods and scales. The results highlight the respective strengths and limitations of different learning-based paradigms and provide practical guidance for industrial 3D metrology, inspection, and city-scale reconstruction.

In Chapter 5, we proposed a novel surface reconstruction framework built upon 3D Gaussian Splatting to address limitations in handling non-collaborative surfaces, including reflective, glossy, and transparent materials. The framework introduces surface normal supervision by leveraging multi-illumination relight images. A pre-trained relighting network generates images under diverse lighting conditions, enhancing geometric and appearance cues, which are then processed by a surface normal estimation network to produce high-quality normal maps for geometric supervision during Gaus-

sian optimization. By jointly integrating illumination-consistent relighting and normal-guided geometry learning, the method produces denser and more accurate Gaussian distributions, improves surface coverage, and captures complex view-dependent effects. Experiments demonstrate significantly improved reconstruction fidelity and rendering quality compared to existing 3DGS baselines.

In Chapter 6, we introduced RenderWorld, a self-supervised autonomous driving framework that generates 3D occupancy labels using Gaussian-based Img2Occ from 2D multi-view depth and semantic images. The differentiable 3DGS representation enables efficient 3D occupancy reconstruction, which serves as input for downstream world models in scene understanding, forecasting, and motion planning. To improve spatial granularity and inference capability, we proposed the Air Mask Variational Autoencoder (AM-VAE), which separately encodes air and non-air voxels to reduce memory consumption while enhancing scene representation. Extensive evaluations on the NuScenes dataset demonstrate high-quality occupancy generation and reliable motion planning, highlighting RenderWorld's potential for real-time, memory-efficient, and robust autonomous driving applications.

In Chapter 7, we proposed 3DSceneEditor, a fully 3DGS-based framework for interactive and controllable editing of complex indoor scenes. Unlike previous multi-step approaches relying on 2D-to-3D semantic projection, 3DSceneEditor enables direct manipulation of individual Gaussians for object addition, movement, recoloring, removal, and replacement. The pipeline integrates semantic labeling through a pre-trained Mask3D instance segmentation model, zero-shot instance grounding using CLIP for language-object alignment, and outlier correction via KNN voting to ensure robust editing. By eliminating multi-view 2D projection steps, the framework achieves interactive-rate editing, improving computational efficiency, output fidelity, and user experience. Experiments demonstrate that 3DSceneEditor outperforms existing state-of-the-art methods in editing precision, speed, and GPU usage, showing its potential for efficient indoor scene modeling, VR/AR content creation, and AI-assisted design workflows.

Overall, while further research remains to be done (see Section 8.2), the contributions presented in this thesis provide a strong foundation for future advances in learning-based 3D reconstruction, scene understanding, and controllable 3D editing.

8.2 Future Works

The rapid progress over the last years and recent breakthroughs in 3D Gaussian Splatting have demonstrated a promising future in 3D reconstruction and scene understand-

ing. In this section, we provide some interesting future directions in this domain.

8.2.1 3DGS Future in 3D

Based on the findings in this study and identified issues of performance of the investigated methods, potential future research directions in the related fields include the following:

Real-time High-fidelity Rendering

Since Gaussian Splatting was proposed in 2023, it quickly became a widely popular topic for 3D reconstruction purposes due to the model's light weight, showing the potential to replace NeRF in rendering scenarios. Its explicit representational approach enables it to bypass sampling from the entire space, unlike NeRF-based methods, thereby requiring few computational resources. However, the effectiveness of Gaussian Splatting is closely tied to the quality of its initial Gaussian points. Particularly when dealing with reflective and refractive surfaces, the quality of these initialized Gaussians emerges as a pivotal factor in achieving high-fidelity ren-dering. Currently, although a limited number of NeRF/Gaussian Splatting studies [269, 105, 139, 163] have looked into enhancing performance in reflective scenes through methods like the mathematical modeling of reflections or normal estimation, there is still a need for further research in real-time, high-quality reconstruction for industrial inspections.

Few-shot 3D Reconstruction

High-quality 3D reconstruction through sparse viewpoints is also a prevailing research focus in the computer vision community. Previously published learning-based reconstruction methods usually relied on dozens to hundreds of images to reconstruct a scene, which led to massive GPU memory con-sumption or TPUs for training purposes, particularly when higher resolution images were used. Recent studies primarily integrated techniques such as depth prior [273, 369], diffusion prior [46, 65], or geometric regularization [237, 118] to achieve novel view synthesis (followed by 3D reconstruction) from a few-shot input images. However, the challenge remains unresolved and awaits further exploration. Removing dependence on camera priors: During the training of the models, the knowledge of the camera interior and exterior parameters, as well as the redundant time loss in format transformation and coordinate system harmonization, hamper model generalization performance [193]. Previous research has utilized photo-

metric reconstruction [289] or has incorporated undistorted monocular depth priors [11] to estimate camera parameters in typical scenarios. However, adapting and generalizing these approaches to non-collaborative surfaces remains a challenge. The recent introduction of DUSt3R [282] represents a significant achievement that enables an MVS network to eliminate its dependence on camera poses by utilizing the cross-attention mechanism of vision Transformer (ViT) [208] to perform image pair joint pose inference. Nevertheless, the experimental results presented in this paper indicate that this method temporarily falls short in surpassing other learning-based methods [308, 281, 173] and traditional handcrafted methods [209, 60] in terms of accuracy or complete-ness. Therefore, the 3D reconstruction of non-collaborative surfaces without a camera prior requires further investigations.

Non-collaborative Surface-related Open Datasets

As widely acknowledged, having sufficient data is a fundamental component in training a high-quality, learning-based model. Currently, we already have datasets such as Clearpose (Transparent objects) [29], TRansPose (Transparent objects) [114], NeRFBK (both real and synthetic transparent and shiny objects) [314], Nex (Shiny objects) [292], Trosd (Transparent objects) [245], Tom-net (Synthetic transparent objects) [22], and Industrial Metal Objects [44] at our disposal. Nevertheless, improving model generalization performance for non-collaborative surfaces still requires more than having common objects with simple structures. Challenging and complex data, captured under various lighting conditions, is necessary to reach further improvements in deep learning-based model performance.

8.2.2 3DGS future in 4D

Embodied AI Simulation

The application of 4D reconstruction extends to embedded AI simulations, where dynamic and temporally coherent models are crucial for training and evaluating intelligent systems. By capturing not only geometry and appearance but also motion over time, advanced neural rendering techniques of NeRF and Gaussian Splatting enable the creation of realistic environments that reflect complex spatiotemporal dynamics.

In embedded AI, these contributions [244, 167, 28] are particularly valuable for robotic perception and interaction in confined or structured environments. Realistic dynamic models of objects—including their deformations, articulations, and motion

patterns—allow robots to be trained in simulation for tasks such as object recognition, fine-grained manipulation , and coordinated multi-agent interactions. By leveraging temporally consistent reconstructions, researchers can construct training datasets that better capture real-world variability, improving the adaptability, robustness, and safety of embedded AI systems.

Dynamic Scene Understanding and Editing

4D reconstruction also plays a crucial role in dynamic scene understanding and editing [133], focusing on perception [311], navigation [279], and content generation [365] in larger-scale, complex environments. Temporally coherent reconstructions provide AI systems with the ability to analyze and predict changes in dynamic scenes, such as urban traffic or crowded pedestrian areas, which is essential for autonomous driving, indoor navigation, and robotic path planning.

Moreover, 4D reconstructions enable scene editing and asset generation, allowing manipulation of objects, agents, and environmental conditions in a temporally consistent manner. This facilitates the creation of diverse virtual worlds, augmentation of training datasets, and production of realistic digital assets for simulation, games, and immersive applications. Tasks such as trajectory prediction, grasp simulation, and environment adaptation directly benefit from these capabilities.

Human-Centered Applications

Another promising direction lies in human-centered applications, where accurate capture of human appearance, motion, and interaction is essential. 4D reconstruction supports temporally consistent representations of people [55], gestures [360], and activities [42], enabling applications in AR/VR [134], telepresence [91], sports training [106], medical [338], and digital twins [69]. By preserving fine-grained spatiotemporal details, these methods enhance realism in immersive experiences and facilitate more natural human–computer interaction. Beyond entertainment and education, dynamic reconstructions of human behaviors and performances also contribute to healthcare and cultural heritage preservation, creating new opportunities for analysis, simulation, and dissemination.

8.2.3 3DGS Future in Metrology and Geomatics

High-Precision Measurement and Inspection

The integration of 3D Gaussian Splatting into metrology introduces new possibilities for high-precision measurement and inspection in industrial and scientific settings [243]. By leveraging its efficient and continuous volumetric representation, 3DGS enables accurate reconstruction of fine geometric details, even for challenging surfaces such as reflective or textureless objects [315]. This is particularly valuable for quality control, reverse engineering, and dimensional analysis, where sub-millimeter accuracy is often required.

Large-Scale Mapping and Geospatial Reconstruction

In geomatics, 3DGS offers a scalable solution for large-scale mapping and geospatial reconstruction [201]. By efficiently handling massive multi-view datasets, it enables the reconstruction of urban environments, infrastructure, and natural landscapes with high fidelity. The continuous nature of Gaussian primitives allows for smooth interpolation across views, improving completeness in areas with sparse observations [316].

Data Fusion and Intelligent Geospatial Analysis

The combination of 3DGS with multi-modal data sources (LiDAR, GNSS, satellite imagery, and other sensing modalities) for intelligent geospatial analysis shall be investigated. 3DGS can serve as a unified representation that captures both geometric and semantic information [137]. This facilitates more comprehensive analysis of spatial data, enabling tasks such as object classification, semantic mapping, and infrastructure management.

Bibliography

[1] Ali Hosseininaveh Ahmadabadian, Ali Karami, and Rouhallah Yazdan. An automatic 3d reconstruction system for texture-less objects. *Robotics and Autonomous Systems*, 117:29–39, 2019.

[2] Muhammad Salman Ali, Chaoning Zhang, Marco Cagnazzo, Giuseppe Valenzise, Enzo Tartaglione, and Sung-Ho Bae. Compression in 3d gaussian splatting: A survey of methods, trends, and future directions. *arXiv preprint arXiv:2502.19457*, 2025.

[3] Titas Anciukevičius, Zexiang Xu, Matthew Fisher, Paul Henderson, Hakan Bilen, Niloy J Mitra, and Paul Guerrero. Renderdiffusion: Image diffusion for 3d reconstruction, inpainting and generation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12608–12618, 2023.

[4] Abdalla Arafa and Didier Stricker. Beyond averages: Open-vocabulary 3d scene understanding with gaussian splatting and bag of embeddings. *arXiv preprint arXiv:2509.12938*, 2025.

[5] Gwangbin Bae and Andrew J. Davison. Rethinking inductive biases for surface normal estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[6] Shuanghao Bai, Wenxuan Song, Jiayi Chen, Yuheng Ji, Zhide Zhong, Jin Yang, Han Zhao, Wanqi Zhou, Wei Zhao, Zhe Li, et al. Towards a unified understanding of robot manipulation: A comprehensive survey. *arXiv preprint arXiv:2510.10903*, 2025.

[7] Chong Bao, Yinda Zhang, Bangbang Yang, Tianxing Fan, Zesong Yang, Hujun Bao, Guofeng Zhang, and Zhaopeng Cui. Sine: Semantic-driven image-based nerf editing with prior-guided editing field. In *Proceedings of the IEEE/CVF*

Conference on Computer Vision and Pattern Recognition, pages 20919–20929, 2023.

[8] Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5470–5479, 2022.

[9] Paul J Besl and Neil D McKay. Method for registration of 3-d shapes. In *Sensor fusion IV: control paradigms and data structures*, volume 1611, pages 586–606. Spie, 1992.

[10] Shariq Farooq Bhat, Reiner Birkl, Diana Wofk, Peter Wonka, and Matthias Müller. Zoedepth: Zero-shot transfer by combining relative and metric depth. *arXiv preprint arXiv:2302.12288*, 2023.

[11] Wenjing Bian, Zirui Wang, Kejie Li, Jia-Wang Bian, and Victor Adrian Prisacariu. Nope-nerf: Optimising neural radiance field with no pose prior. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4160–4169, 2023.

[12] Reiner Birkl, Diana Wofk, and Matthias Müller. Midas v3. 1–a model zoo for robust monocular relative depth estimation. *arXiv preprint arXiv:2307.14460*, 2023.

[13] Simon Boeder, Fabian Gigengack, and Benjamin Risse. Occflownet: Towards self-supervised occupancy estimation via differentiable rendering and occupancy flow. *arXiv preprint arXiv:2402.12792*, 2024.

[14] Mariusz Bojarski. End to end learning for self-driving cars. *arXiv preprint arXiv:1604.07316*, 2016.

[15] Mario Botsch, Alexander Hornung, Matthias Zwicker, and Leif Kobbelt. High-quality surface splatting on today's gpus. In *Proceedings Eurographics/IEEE VGTC Symposium on Point-Based Graphics*, pages 17–141. IEEE, 2005.

[16] Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18392–18402, 2023.

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

[18] Wenxiao Cai, Iaroslav Ponomarenko, Jianhao Yuan, Xiaoqi Li, Wankou Yang, Hao Dong, and Bo Zhao. Spatialbot: Precise spatial understanding with vision language models. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9490–9498. IEEE, 2025.

[19] Anh-Quan Cao and Raoul De Charette. Monoscene: Monocular 3d semantic scene completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3991–4001, 2022.

[20] Chenjie Cao, Xinlin Ren, and Yanwei Fu. Mvsformer: Multi-view stereo by learning robust image features and temperature-based depth. *arXiv preprint arXiv:2208.02541*, 2022.

[21] Jiazhong Cen, Jiemin Fang, Chen Yang, Lingxi Xie, Xiaopeng Zhang, Wei Shen, and Qi Tian. Segment any 3d gaussians. *arXiv preprint arXiv:2312.00860*, 2023.

[22] Guanying Chen, Kai Han, and Kwan-Yee K Wong. Tom-net: Learning transparent object matting from a single image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9233–9241, 2018.

[23] Guikun Chen and Wenguan Wang. A survey on 3d gaussian splatting. *arXiv preprint arXiv:2401.03890*, 2024.

[24] Hansheng Chen, Jiatao Gu, Anpei Chen, Wei Tian, Zhuowen Tu, Lingjie Liu, and Hao Su. Single-stage diffusion nerf: A unified approach to 3d generation and reconstruction. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2416–2425, 2023.

[25] Jun-Kun Chen and Yu-Xiong Wang. Proedit: Simple progression is all you need for high-quality 3D scene editing. In *NeurIPS*, 2024.

[26] Li Chen, Penghao Wu, Kashyap Chitta, Bernhard Jaeger, Andreas Geiger, and Hongyang Li. End-to-end autonomous driving: Challenges and frontiers. *IEEE*

Transactions on Pattern Analysis and Machine Intelligence, 46(12):10164–10183, 2024.

[27] Minghao Chen, Iro Laina, and Andrea Vedaldi. Dge: Direct gaussian 3d editing by consistent multi-view editing. *arXiv preprint arXiv:2404.18929*, 2024.

[28] Timothy Chen, Ola Shorinwa, Joseph Bruno, Aiden Swann, Javier Yu, Weijia Zeng, Keiko Nagami, Philip Dames, and Mac Schwager. Splat-nav: Safe real-time robot navigation in gaussian splatting maps. *IEEE Transactions on Robotics*, 2025.

[29] Xinyu Chen, Hao Zhang, Zhixin Yu, Anthony Opipari, and Chadwicke O. Jenkins. Clearpose: Large-scale transparent object dataset and benchmark. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 381–396, Tel Aviv-Yafo, Israel, October 2022.

[30] Yang Chen, Yingwei Pan, Yehao Li, Ting Yao, and Tao Mei. Control3d: Towards controllable text-to-3d generation. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 1148–1156, 2023.

[31] Yiwen Chen, Zilong Chen, Chi Zhang, Feng Wang, Xiaofeng Yang, Yikai Wang, Zhongang Cai, Lei Yang, Huaping Liu, and Guosheng Lin. Gaussianeditor: Swift and controllable 3d editing with gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21476–21485, 2024.

[32] Yufan Chen, Lizhen Wang, Qijing Li, Hongjiang Xiao, Shengping Zhang, Hongxun Yao, and Yebin Liu. Monogaussianavatar: Monocular gaussian point-based head avatar. In *ACM SIGGRAPH 2024 conference papers*, pages 1–9, 2024.

[33] Zheng Chen, Chen Wang, Yuan-Chen Guo, and Song-Hai Zhang. Structnerf: Neural radiance fields for indoor scenes with structural hints. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

[34] Zilong Chen, Feng Wang, Yikai Wang, and Huaping Liu. Text-to-3d using gaussian splatting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21401–21412, 2024.

[35] Kai Cheng, Xiaoxiao Long, Wei Yin, Jin Wang, Zhiqiang Wu, Yuexin Ma, Kaixuan Wang, Xiaozhi Chen, and Xuejin Chen. Uc-nerf: Neural radiance field

for under-calibrated multi-view cameras in autonomous driving. *arXiv preprint arXiv:2311.16945*, 2023.

[36] Changwoon Choi, Juhyeon Kim, and Young Min Kim. Ibl-nerf: Image-based lighting formulation of neural radiance fields. In *Computer graphics forum*, volume 42, page e14929. Wiley Online Library, 2023.

[37] Jun Myeong Choi, Annie Wang, Pieter Peers, Anand Bhattad, and Roni Sengupta. Scribblelight: Single image indoor relighting with scribbles. In *CVPR*, 2025. arXiv:2411.17696.

[38] Seokhun Choi, Hyeonseop Song, Jaechul Kim, Taehyeong Kim, and Hoseok Do. Click-gaussian: Interactive segmentation to any 3d gaussians. *arXiv preprint arXiv:2407.11793*, 2024.

[39] Jaeyoung Chung, Jeongtaek Oh, and Kyoung Mu Lee. Depth-regularized optimization for 3d gaussian splatting in few-shot images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 811–820, 2024.

[40] Yaodong Cui, Shucheng Huang, Jiaming Zhong, Zhenan Liu, Yutong Wang, Chen Sun, Bai Li, Xiao Wang, and Amir Khajepour. Drivellm: Charting the path toward full autonomous driving with large language models. *IEEE Transactions on Intelligent Vehicles*, 2023.

[41] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5828–5839, 2017.

[42] Pinxuan Dai, Peiquan Zhang, Zheng Dong, Ke Xu, Yifan Peng, Dandan Ding, Yujun Shen, Yin Yang, Xinguo Liu, Rynson WH Lau, et al. 4d gaussian videos with motion layering. *ACM Transactions on Graphics (TOG)*, 44(4):1–14, 2025.

[43] Erik Daxberger, Nina Wenzel, David Griffiths, Haiming Gang, Justin Lazarow, Gefen Kohavi, Kai Kang, Marcin Eichner, Yinfei Yang, Afshin Dehghan, et al. Mm-spatial: Exploring 3d spatial understanding in multimodal llms. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7395–7408, 2025.

[44] Peter De Roovere, Steven Moonen, Nick Michiels, and Francis Wyffels. Dataset of industrial metal objects. *arXiv preprint arXiv:2208.04052*, 2022.

[45] Pieter De Roovere, Stef Moonen, Nick Michiels, and Francis Wyffels. Dataset of industrial metal objects. *arXiv preprint arXiv:2208.04052*, 2022.

[46] Congyue Deng, Chiyu Jiang, Charles R Qi, Xinchen Yan, Yin Zhou, Leonidas Guibas, Dragomir Anguelov, et al. Nerdi: Single-view nerf synthesis with language-guided diffusion as general image priors. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 20637–20647, 2023.

[47] Kangle Deng, Andrew Liu, Jun-Yan Zhu, and Deva Ramanan. Depth-supervised nerf: Fewer views and faster training for free. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12882–12891, 2022.

[48] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018.

[49] Helisa Dhamo, Yinyu Nie, Arthur Moreau, Jifei Song, Richard Shaw, Yiren Zhou, and Eduardo Pérez-Pellitero. Headgas: Real-time animatable head avatars via 3d gaussian splatting. In *European Conference on Computer Vision*, pages 459–476. Springer, 2024.

[50] Yikang Ding, Wentao Yuan, Qingtian Zhu, Haotian Zhang, Xiangyue Liu, Yuanjiang Wang, and Xiao Liu. Transmvsnet: Global context-aware multi-view stereo network with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8585–8594, 2022.

[51] Yikang Ding, Qingtian Zhu, Xiangyue Liu, Wentao Yuan, Haotian Zhang, and Chi Zhang. Kd-mvs: Knowledge distillation based self-supervised learning for multi-view stereo. In *European conference on computer vision*, pages 630–646. Springer, 2022.

[52] Jiahua Dong and Yu-Xiong Wang. Vica-nerf: View-consistency-aware 3d editing of neural radiance fields. *Advances in Neural Information Processing Systems*, 36, 2024.

[53] Jiahua Dong and Yu-Xiong Wang. 3dgs-drag: Dragging gaussians for intuitive point-based 3d editing. *arXiv preprint arXiv:2601.07963*, 2026.

[54] Cheng-De Fan et al. Spectromotion: Dynamic 3d reconstruction of specular scenes. *arXiv preprint arXiv:2410.17249*, 2024.

[55] Zhe Fan, Shi-Sheng Huang, Yichi Zhang, Dachao Shang, Juyong Zhang, Yudong Guo, and Hua Huang. Rgavatar: Relightable 4d gaussian avatar from monocular videos. *IEEE Transactions on Visualization and Computer Graphics*, 2025.

[56] Yalda Foroutan, Daniel Rebain, Kwang Moo Yi, and Andrea Tagliasacchi. Does gaussian splatting need sfm initialization? *arXiv preprint arXiv:2404.12547*, 2024.

[57] Frédéric Fortier-Chouinard, Zitian Zhang, Louis-Etienne Messier, Mathieu Garon, Anand Bhattad, and Jean-François Lalonde. Spotlight: Shadow-guided object relighting via diffusion. *arXiv preprint*, arXiv:2411.18665, 2024.

[58] Xiao Fu, Shangzhan Zhang, Tianrun Chen, Yichong Lu, Lanyun Zhu, Xiaowei Zhou, Andreas Geiger, and Yiyi Liao. Panoptic nerf: 3d-to-2d label transfer for panoptic urban scene segmentation. In *2022 International Conference on 3D Vision (3DV)*, pages 1–11. IEEE, 2022.

[59] Yang Fu, Sifei Liu, Amey Kulkarni, Jan Kautz, Alexei A Efros, and Xiaolong Wang. Colmap-free 3d gaussian splatting. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 20796–20805, 2024.

[60] Yasutaka Furukawa and Jean Ponce. Accurate, dense, and robust multiview stereopsis. *IEEE transactions on pattern analysis and machine intelligence*, 32(8):1362–1376, 2009.

[61] Wanshui Gan, Fang Liu, Hongbin Xu, Ningkai Mo, and Naoto Yokoya. Gaussianocc: Fully self-supervised and efficient 3d occupancy estimation with gaussian splatting. *arXiv preprint arXiv:2408.11447*, 2024.

[62] Kyle Gao, Yina Gao, Hongjie He, Dening Lu, Linlin Xu, and Jonathan Li. Nerf: Neural radiance field in 3d vision, a comprehensive review. *arXiv preprint arXiv:2210.00379*, 2022.

[63] Lin Gao, Jie Yang, Bo-Tao Zhang, Jia-Mu Sun, Yu-Jie Yuan, Hongbo Fu, and Yu-Kun Lai. Mesh-based gaussian splatting for real-time large-scale deformation. *arXiv preprint arXiv:2402.04796*, 2024.

[64] Sorin Grigorescu, Bogdan Trasnea, Tiberiu Cocias, and Gigel Macesanu. A survey of deep learning techniques for autonomous driving. *Journal of field robotics*, 37(3):362–386, 2020.

[65] Jiatao Gu, Alex Trevithick, Kai-En Lin, Joshua M Susskind, Christian Theobalt, Lingjie Liu, and Ravi Ramamoorthi. Nerfdiff: Single-image view synthesis with nerf-guided distillation from 3d-aware diffusion. In *International Conference on Machine Learning*, pages 11808–11826. PMLR, 2023.

[66] Junru Gu, Chenxu Hu, Tianyuan Zhang, Xuanyao Chen, Yilun Wang, Yue Wang, and Hang Zhao. Vip3d: End-to-end visual trajectory prediction via 3d agent queries. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5496–5506, 2023.

[67] Antoine Guédon and Vincent Lepetit. Sugar: Surface-aligned gaussian splatting for efficient 3d mesh reconstruction and high-quality mesh rendering. In *CVPR*, pages 5354–5363, 2024.

[68] Gabriele Guidi, Fabio Remondino, Michele Russo, Fabio Menna, Alessandro Rizzi, et al. 3d modeling of large and complex site using multi-sensor integration and multi-resolution data. In *VAST*, pages 85–92, 2008.

[69] Junfu Guo, Yu Xin, Gaoyi Liu, Kai Xu, Ligang Liu, and Ruizhen Hu. Articulatedgs: Self-supervised digital twin modeling of articulated objects using 3d gaussian splatting. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 27144–27153, 2025.

[70] Yuan-Chen Guo, Di Kang, Linchao Bao, Yu He, and Song-Hai Zhang. Nerfren: Neural radiance fields with reflections. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18409–18418, 2022.

[71] David Ha and Jürgen Schmidhuber. World models. *arXiv preprint arXiv:1803.10122*, 2018.

[72] Jahanzeb Hafeez, Jaehyun Lee, Soonchul Kwon, Sungjae Ha, Gitaek Hur, and Seunghyun Lee. Evaluating feature extraction methods with synthetic noise patterns

for image-based modelling of texture-less objects. *Remote Sensing*, 12(23):3886, 2020.

[73] Ayaan Haque, Matthew Tancik, Alexei A Efros, Aleksander Holynski, and Angjoo Kanazawa. Instruct-nerf2nerf: Editing 3d scenes with instructions. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19740–19750, 2023.

[74] Runze He, Shaofei Huang, Xuecheng Nie, Tianrui Hui, Luoqi Liu, Jiao Dai, Jizhong Han, Guanbin Li, and Si Liu. Customize your nerf: Adaptive source driven 3d scene editing via local-global iterative training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6966–6975, 2024.

[75] Yicong Hong, Kai Zhang, Jiuxiang Gu, Sai Bi, Yang Zhou, Difan Liu, Feng Liu, Kalyan Sunkavalli, Trung Bui, and Hao Tan. Lrm: Large reconstruction model for single image to 3d. *arXiv preprint arXiv:2311.04400*, 2023.

[76] A Hosseininaveh, R Yazdan, A Karami, M Moradi, and F Ghorbani. A low-cost and portable system for 3d reconstruction of texture-less objects. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 40:327–332, 2015.

[77] Hao-Yu Hsu, Zhi-Hao Lin, Albert Zhai, Hongchi Xia, and Shenlong Wang. Autovfx: Physically realistic video editing from natural language instructions. *arXiv preprint arXiv:2411.02394*, 2024.

[78] Anthony Hu, Zak Murez, Nikhil Mohan, Sofía Dudas, Jeffrey Hawke, Vijay Badrinarayanan, Roberto Cipolla, and Alex Kendall. Fiery: Future instance prediction in bird's-eye view from surround monocular cameras. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15273–15282, 2021.

[79] Anthony Hu, Lloyd Russell, Hudson Yeo, Zak Murez, George Fedoseev, Alex Kendall, Jamie Shotton, and Gianluca Corrado. Gaia-1: A generative world model for autonomous driving. *arXiv preprint arXiv:2309.17080*, 2023.

[80] Peiyun Hu, Aaron Huang, John Dolan, David Held, and Deva Ramanan. Safe local motion planning with self-supervised freespace forecasting. In *Proceedings of*

the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 12732–12741, 2021.

[81] Shengchao Hu, Li Chen, Penghao Wu, Hongyang Li, Junchi Yan, and Dacheng Tao. St-p3: End-to-end vision-based autonomous driving via spatial-temporal feature learning. In *European Conference on Computer Vision*, pages 533–549. Springer, 2022.

[82] Xu Hu, Yuxi Wang, Lue Fan, Junsong Fan, Junran Peng, Zhen Lei, Qing Li, and Zhaoxiang Zhang. Semantic anything in 3d gaussians. *arXiv preprint arXiv:2401.17857*, 2024.

[83] Yao Hu, Shaopu Wang, Xuemin Cheng, Chengqiang Xu, and Qun Hao. Dynamic deformation measurement of specular surface with deflectometry and speckle digital image correlation. *Sensors*, 20(5):1278, 2020.

[84] Yihan Hu, Jiazhi Yang, Li Chen, Keyu Li, Chonghao Sima, Xizhou Zhu, Siqi Chai, Senyao Du, Tianwei Lin, Wenhai Wang, et al. Planning-oriented autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17853–17862, 2023.

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

[86] Junjie Huang and Guan Huang. Bevdet4d: Exploit temporal cues in multi-camera 3d object detection. *arXiv preprint arXiv:2203.17054*, 2022.

[87] Junjie Huang, Guan Huang, Zheng Zhu, Yun Ye, and Dalong Du. Bevdet: High-performance multi-camera 3d object detection in bird-eye-view. *arXiv preprint arXiv:2112.11790*, 2021.

[88] Letian Huang, Jiayang Bai, Jie Guo, and Yanwen Guo. Gs++: Error analyzing and optimal gaussian splatting. *arXiv preprint arXiv:2402.00752*, 2(3), 2024.

[89] Sanao Huang, Ke Xu, Ming Li, and Mingren Wu. Improved visual inspection through 3d image reconstruction of defects based on the photometric stereo technique. *Sensors*, 19(22):4970, 2019.

[90] Tianyu Huang, Yihan Zeng, Zhilu Zhang, Wan Xu, Hang Xu, Songcen Xu, Rynson WH Lau, and Wangmeng Zuo. Dreamcontrol: Control-based text-to-3d generation with 3d self-prior. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5364–5373, 2024.

[91] Xincheng Huang, Dieter Frehlich, Ziyi Xia, Peyman Gholami, and Robert Xiao. Gaussiannexus: Room-scale real-time ar/vr telepresence with gaussian splatting. In *Proceedings of the 38th Annual ACM Symposium on User Interface Software and Technology*, pages 1–18, 2025.

[92] Yiming Huang, Beilei Cui, Long Bai, Ziqi Guo, Mengya Xu, Mobarakol Islam, and Hongliang Ren. Endo-4dgs: Endoscopic monocular scene reconstruction with 4d gaussian splatting. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 197–207. Springer, 2024.

[93] Yuanhui Huang, Wenzhao Zheng, Borui Zhang, Jie Zhou, and Jiwen Lu. Selfocc: Self-supervised vision-based 3d occupancy prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19946–19956, 2024.

[94] Yuanhui Huang, Wenzhao Zheng, Yunpeng Zhang, Jie Zhou, and Jiwen Lu. Tri-perspective view for vision-based 3d semantic occupancy prediction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9223–9232, 2023.

[95] Yuanhui Huang, Wenzhao Zheng, Yunpeng Zhang, Jie Zhou, and Jiwen Lu. Gaussianformer: Scene as gaussians for vision-based 3d semantic occupancy prediction. *arXiv preprint arXiv:2405.17429*, 2024.

[96] Zhening Huang, Xiaoyang Wu, Xi Chen, Hengshuang Zhao, Lei Zhu, and Joan Lasenby. Openins3d: Snap and lookup for 3d open-vocabulary instance segmentation. In *European Conference on Computer Vision*, pages 169–185. Springer, 2024.

[97] Zhiyu Huang, Haochen Liu, and Chen Lv. Gameformer: Game-theoretic modeling and learning of transformer-based interactive prediction and planning for autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3903–3913, 2023.

[98] Nail Ibrahimli, Hugo Ledoux, Julian FP Kooij, and Liangliang Nan. Ddl-mvs: Depth discontinuity learning for multi-view stereo networks. *Remote Sensing*, 15(12):2970, 2023.

[99] Jeffrey Ichnowski, Yahav Avigal, Justin Kerr, and Ken Goldberg. Dex-nerf: Using a neural radiance field to grasp transparent objects. *arXiv preprint arXiv:2110.14217*, 2021.

[100] Clément Jambon, Bernhard Kerbl, Georgios Kopanas, Stavros Diolatzis, Thomas Leimkühler, and George Drettakis. Nerfshop: Interactive editing of neural radiance fields. *Proceedings of the ACM on Computer Graphics and Interactive Techniques*, 6(1), 2023.

[101] Rasmus Jensen, Anders Dahl, George Vogiatzis, Engin Tola, and Henrik Aanæs. Large scale multi-view stereopsis evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 406–413, 2014.

[102] Bo Jiang, Shaoyu Chen, Qing Xu, Bencheng Liao, Jiajie Chen, Helong Zhou, Qian Zhang, Wenyu Liu, Chang Huang, and Xinggang Wang. Vad: Vectorized scene representation for efficient autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8340–8350, 2023.

[103] Chongcong Jiang, Tianxingjian Ding, Chuhan Song, Jiachen Tu, Ziyang Yan, Yihua Shao, Zhenyi Wang, Yuzhang Shang, Tianyu Han, and Yu Tian. Medical sam3: A foundation model for universal prompt-driven medical image segmentation. *arXiv preprint arXiv:2601.10880*, 2026.

[104] Jianfei Jiang, Mingwei Cao, Jun Yi, and Chenglong Li. Di-mvs: learning efficient multi-view stereo with depth-aware iterations. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3180–3184. IEEE, 2024.

[105] Yingwenqi Jiang et al. Gaussianshader: 3d gaussian splatting with shading functions for reflective surfaces. *arXiv preprint arXiv:2311.17977*, 2023.

[106] Yuqi Jiang, Jing Li, Haidong Qin, Yanran Dai, Jing Liu, Guodong Zhang, Canbin Zhang, and Tao Yang. Gs-sfs: Joint gaussian splatting and shape-from-silhouette for multiple human reconstruction in large-scale sports scenes. *IEEE Transactions on Multimedia*, 26:11095–11110, 2024.

[107] Haian Jin, Zexiang Xu, and Nima Khademi Kalantari. Neural gaffer: Relighting scenes for view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[108] Gyeongjin Kang, Jisang Yoo, Jihyeon Park, Seungtae Nam, Hyeonsoo Im, Sangheon Shin, Sangpil Kim, and Eunbyung Park. Selfsplat: Pose-free and 3d prior-free generalizable 3d gaussian splatting. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 22012–22022, 2025.

[109] A Karami, F Menna, and F Remondino. Investigating 3d reconstruction of non-collaborative surfaces through photogrammetry and photometric stereo. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 43:519–526, 2021.

[110] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics (TOG)*, 42(4):1–14, 2023.

[111] Umar Khalid, Hasan Iqbal, Azib Farooq, Jing Hua, and Chen Chen. 3dego: 3d editing on the go! In *European Conference on Computer Vision*, pages 73–89. Springer, 2025.

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

[113] Tarasha Khurana, Peiyun Hu, Achal Dave, Jason Ziglar, David Held, and Deva Ramanan. Differentiable raycasting for self-supervised occupancy forecasting. In *European Conference on Computer Vision*, pages 353–369. Springer, 2022.

[114] Gwanghyun Kim and Se Young Chun. Datid-3d: Diversity-preserved domain adaptation using text-to-image diffusion for 3d generative model. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14203–14213, 2023.

[115] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4015–4026, 2023.

[116] Arno Knapitsch, Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Tanks and temples: Benchmarking large-scale scene reconstruction. *ACM Transactions on Graphics (ToG)*, 36(4):1–13, 2017.

[117] Pou-Chun Kung, Skanda Harisha, Ram Vasudevan, Aline Eid, and Katherine A Skinner. Radarsplat: Radar gaussian splatting for high-fidelity data synthesis and 3d reconstruction of autonomous driving scenes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 27596–27606, 2025.

[118] Min-Seop Kwak, Jiuhn Song, and Seungryong Kim. Geconerf: Few-shot neural radiance fields via geometric consistency. *arXiv preprint arXiv:2301.10941*, 2023.

[119] Joohyun Kwon, Hanbyel Cho, and Junmo Kim. Instruct-4dgs: Efficient dynamic scene editing via 4d gaussian-based static-dynamic separation. *arXiv preprint arXiv:2502.02091*, 2025.

[120] Shuichang Lai, Letian Huang, Jie Guo, Kai Cheng, Bowen Pan, Xiaoxiao Long, Jiangjing Lyu, Chengfei Lv, and Yanwen Guo. Glossygs: Inverse rendering of glossy objects with 3d gaussian splatting. *IEEE Transactions on Visualization and Computer Graphics*, 2025.

[121] Verica Lazova, Vladimir Guzov, Kyle Olszewski, Sergey Tulyakov, and Gerard Pons-Moll. Control-nerf: Editable feature volumes for scene rendering and manipulation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 4340–4350, 2023.

[122] Byeonghyeon Lee, Howoong Lee, Xiangyu Sun, Usman Ali, and Eunbyung Park. Deblurring 3d gaussian splatting. In *European Conference on Computer Vision*, pages 127–143. Springer, 2024.

[123] Dong In Lee, Hyeongcheol Park, Jiyoung Seo, Eunbyung Park, Hyunje Park, Ha Dam Baek, Sangheon Shin, Sangmin Kim, and Sangpil Kim. Editsplat: Multi-view fusion and attention-guided optimization for view-consistent 3d scene editing with 3d gaussian splatting. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 11135–11145, 2025.

[124] Di Li, Jie Feng, Jiahao Chen, Weisheng Dong, Guanbin Li, Guangming Shi, and Licheng Jiao. Egosplat: Open-vocabulary egocentric scene understanding with language embedded 3d gaussian splatting. *arXiv preprint arXiv:2503.11345*, 2025.

[125] Hong Li, Chongjie Ye, Houyuan Chen, Weiqing Xiao, Ziyang Yan, Lixing Xiao, Zhaoxi Chen, Jianfeng Xiang, Shaocong Xu, Xuhui Liu, et al. Near: Coupled neural asset-renderer stack. *arXiv preprint arXiv:2511.18600*, 2025.

[126] Jiahao Li, Hao Tan, Kai Zhang, Zexiang Xu, Fujun Luan, Yinghao Xu, Yicong Hong, Kalyan Sunkavalli, Greg Shakhnarovich, and Sai Bi. Instant3d: Fast text-to-3d with sparse-view generation and large reconstruction model. *arXiv preprint arXiv:2311.06214*, 2023.

[127] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pages 19730–19742. PMLR, 2023.

[128] Kunyang Li, Jeffrey A. Chan Santiago, Sarinda Dhanesh Samarasinghe, Gaowen Liu, and Mubarak Shah. Gvd: Guiding video diffusion model for scalable video distillation. *arXiv preprint arXiv:2507.22360*, 2025.

[129] Ming Li, Pan Zhou, Jia-Wei Liu, Jussi Keppo, Min Lin, Shuicheng Yan, and Xiangyu Xu. Instant3d: Instant text-to-3d generation. *International Journal of Computer Vision*, pages 1–17, 2024.

[130] Mingrui Li, Shuhong Liu, Heng Zhou, Guohao Zhu, Na Cheng, Tianchen Deng, and Hongyu Wang. Sgs-slam: Semantic gaussian splatting for neural dense slam. In *European Conference on Computer Vision*, pages 163–179. Springer, 2024.

[131] Mingwei Li, Pu Pang, Hehe Fan, Hua Huang, and Yi Yang. Tsgs: Improving gaussian splatting for transparent surface reconstruction via normal and de-lighting priors. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pages 7220–7229, 2025.

[132] Qingquan Li, Hui Huang, Wenshuai Yu, and San Jiang. Optimized views photogrammetry: Precision analysis and a large-scale case study in qingdao. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 16:1144–1159, 2023.

[133] Wanhua Li, Renping Zhou, Jiawei Zhou, Yingwei Song, Johannes Herter, Minghan Qin, Gao Huang, and Hanspeter Pfister. 4d langsplat: 4d language gaussian splatting via multimodal large language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 22001–22011, 2025.

[134] Yijie Li, Ce Zheng, Yijie He, Joel Julin, Ryosuke Ichikari, Satoki Ogiso, Satoshi Nakae, Akihiro Sato, Takeshi Kurata, and Laszlo A Jeni. Training-free multi-view 4d human motion reconstruction virtual reality system. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 318–327, 2026.

[135] Yinhao Li, Zheng Ge, Guanyi Yu, Jinrong Yang, Zengran Wang, Yukang Shi, Jianjian Sun, and Zeming Li. Bevdepth: Acquisition of reliable depth for multi-view 3d object detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 1477–1485, 2023.

[136] Yixuan Li, Lihan Jiang, Linning Xu, Yuanbo Xiangli, Zhenzhi Wang, Dahua Lin, and Bo Dai. Matrixcity: A large-scale city dataset for city-scale neural rendering and beyond. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3205–3215, 2023.

[137] Yuchen Li, Jing Li, Kai Qin, Haitao Wu, Yin Gao, and Banghui Yang. Cross-modal 3-d gaussian reconstruction: Fusing image, geometry, and open-vocabulary semantics for large-scale real-world scenes. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 19:12166–12186, 2026.

[138] Zhaoshuo Li, Thomas Müller, Alex Evans, Russell H Taylor, Mathias Unberath, Ming-Yu Liu, and Chen-Hsuan Lin. Neuralangelo: High-fidelity neural surface reconstruction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8456–8465, 2023.

[139] Zhengqin Li, Yu-Ying Yeh, and Manmohan Chandraker. Through the looking glass: Neural 3d reconstruction of transparent shapes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1262–1271, 2020.

[140] Zhiqi Li, Wenhai Wang, Hongyang Li, Enze Xie, Chonghao Sima, Tong Lu, Yu Qiao, and Jifeng Dai. Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers. In *European conference on computer vision*, pages 1–18. Springer, 2022.

[141] Zhiqi Li, Zhiding Yu, David Austin, Mingsheng Fang, Shiyi Lan, Jan Kautz, and Jose M Alvarez. Fb-occ: 3d occupancy prediction based on forward-backward view transformation. *arXiv preprint arXiv:2307.01492*, 2023.

[142] Minwen Liao, Haobo Dong, Xinyi Wang, Kurban Ubul, Yihua Shao, and Ziyang Yan. Gm-moe: Low-light enhancement with gated-mechanism mixture-of-experts. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8766–8776, 2025.

[143] Chen-Hsuan Lin, Jun Gao, Luming Tang, Towaki Takikawa, Xiaohui Zeng, Xun Huang, Karsten Kreis, Sanja Fidler, Ming-Yu Liu, and Tsung-Yi Lin. Magic3d: High-resolution text-to-3d content creation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 300–309, 2023.

[144] Jiaqi Lin, Zhihao Li, Xiao Tang, Jianzhuang Liu, Shiyong Liu, Jiayue Liu, Yangdi Lu, Xiaofei Wu, Songcen Xu, Youliang Yan, et al. Vastgaussian: Vast 3d gaussians for large scene reconstruction. In *CVPR*, pages 5166–5175, 2024.

[145] Liqiang Lin, Yilin Liu, Yue Hu, Xingguang Yan, Ke Xie, and Hui Huang. Capturing, reconstructing, and simulating: the urbanscene3d dataset. In *European Conference on Computer Vision*, pages 93–109. Springer, 2022.

[146] Peng Ling, Tiao Tan, Jiaqi Lin, and Wenming Yang. Sovgaussian: Sparse-view 3d gaussian splatting for open-vocabulary scene understanding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 5343–5351, 2025.

[147] Dengzhi Liu, Yu Zhang, Lin Luo, Jinlong Li, and Xiaorong Gao. Pdc-net: Robust point cloud registration using deep cyclic neural network combined with pca. *Applied optics*, 60(11):2990–2997, 2021.

[148] Fangfu Liu, Diankun Wu, Yi Wei, Yongming Rao, and Yueqi Duan. Sherpa3d: Boosting high-fidelity text-to-3d generation via coarse 3d prior. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20763–20774, 2024.

[149] Jianheng Liu, Chunran Zheng, Yunfei Wan, Bowen Wang, Yixi Cai, and Fu Zhang. Neural surface reconstruction and rendering for lidar-visual systems, 2024.

[150] Ming Liu, Yuxuan Liang, Siwei Chen, Junjie Wang, and Yang Na. Sv-2dgs: Optimization of sparse view 3d reconstruction based on 2dgs models. *Expert Systems with Applications*, 297:129334, 2026.

[151] Minghua Liu, Ruoxi Shi, Linghao Chen, Zhuoyang Zhang, Chao Xu, Xinyue Wei, Hansheng Chen, Chong Zeng, Jiayuan Gu, and Hao Su. One-2-3-45++: Fast single image to 3d objects with consistent multi-view generation and 3d diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10072–10083, 2024.

[152] Minghua Liu, Chao Xu, Haian Jin, Linghao Chen, Mukund Varma T, Zexiang Xu, and Hao Su. One-2-3-45: Any single image to 3d mesh in 45 seconds without per-shape optimization. *Advances in Neural Information Processing Systems*, 36, 2024.

[153] Pengkun Liu, Yikai Wang, Fuchun Sun, Jiafang Li, Hang Xiao, Hongxiang Xue, and Xinzhou Wang. Isotropic3d: Image-to-3d generation based on a single clip embedding. *arXiv preprint arXiv:2403.10395*, 2024.

[154] Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. Zero-1-to-3: Zero-shot one image to 3d object. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9298–9309, 2023.

[155] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *European Conference on Computer Vision*, pages 38–55. Springer, 2024.

[156] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499*, 2023.

[157] Shuai Liu, Mengmeng Yang, Tingyan Xing, and Ran Yang. A survey of 3d reconstruction: the evolution from multi-view geometry to nerf and 3dgs. *Sensors*, 25(18):5748, 2025.

[158] Tianqi Liu, Xinyi Ye, Weiyue Zhao, Zhiyu Pan, Min Shi, and Zhiguo Cao. When epipolar constraint meets non-local operators in multi-view stereo. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 18088–18097, 2023.

[159] Yang Liu, Chuanchen Luo, Lue Fan, Naiyan Wang, Junran Peng, and Zhaoxiang Zhang. Citygaussian: Real-time high-quality large-scale scene rendering with gaussians. In *ECCV*, pages 265–282. Springer, 2024.

[160] Yifan Liu, Chenxin Li, Chen Yang, and Yixuan Yuan. Endogaussian: Gaussian splatting for deformable surgical scene reconstruction. *arXiv preprint arXiv:2401.12561*, 3(7):8, 2024.

[161] Yili Liu, Linzhan Mou, Xuan Yu, Chenrui Han, Sitong Mao, Rong Xiong, and Yue Wang. Let occ flow: Self-supervised 3d occupancy flow prediction. *arXiv preprint arXiv:2407.07587*, 2024.

[162] Ying-Tian Liu, Yuan-Chen Guo, Guan Luo, Heyi Sun, Wei Yin, and Song-Hai Zhang. Pi3d: Efficient text-to-3d generation with pseudo-image diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19915–19924, 2024.

[163] Yuan Liu, Peng Wang, Cheng Lin, Xiaoxiao Long, Jiepeng Wang, Lingjie Liu, Taku Komura, and Wenping Wang. Nero: Neural geometry and brdf reconstruction of reflective objects from multiview images. *ACM Transactions on Graphics (ToG)*, 42(4):1–22, 2023.

[164] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

[165] Zhijian Liu, Haotian Tang, Alexander Amini, Xinyu Yang, Huizi Mao, Daniela L Rus, and Song Han. Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation. In *2023 IEEE international conference on robotics and automation (ICRA)*, pages 2774–2781. IEEE, 2023.

[166] Xiaoxiao Long, Yuan-Chen Guo, Cheng Lin, Yuan Liu, Zhiyang Dou, Lingjie Liu, Yuexin Ma, Song-Hai Zhang, Marc Habermann, Christian Theobalt, et al. Wonder3d: Single image to 3d using cross-domain diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9970–9980, 2024.

[167] Haozhe Lou, Yurong Liu, Yike Pan, Yiran Geng, Jianteng Chen, Wenlong Ma, Chenglong Li, Lin Wang, Hengzhen Feng, Lu Shi, et al. Robo-gs: A physics

consistent spatial-temporal model for robotic arm with hybrid representation. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 15379–15386. IEEE, 2025.

[168] Chongshan Lu, Fukun Yin, Xin Chen, Wen Liu, Tao Chen, Gang Yu, and Jiayuan Fan. A large-scale outdoor multi-modal dataset and benchmark for novel view synthesis and implicit scene reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7557–7567, 2023.

[169] Tao Lu, Mulin Yu, Linning Xu, Yuanbo Xiangli, Limin Wang, Dahua Lin, and Bo Dai. Scaffold-gs: Structured 3d gaussians for view-adaptive rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[170] Yuhang Lu, Xinge Zhu, Tai Wang, and Yuexin Ma. Octreeocc: Efficient and multi-granularity occupancy prediction using octree queries. *arXiv preprint arXiv:2312.03774*, 2023.

[171] Yueen Ma, Zixing Song, Yuzheng Zhuang, Jianye Hao, and Irwin King. A survey on vision-language-action models for embodied ai. *arXiv preprint arXiv:2405.14093*, 2024.

[172] Yuexin Ma, Xinge Zhu, Sibo Zhang, Ruigang Yang, Wenping Wang, and Dinesh Manocha. Trafficpredict: Trajectory prediction for heterogeneous traffic-agents. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 6120–6127, 2019.

[173] Zeyu Ma, Zachary Teed, and Jia Deng. Multiview stereo with cascaded epipolar raft. In *European Conference on Computer Vision*, pages 734–750. Springer, 2022.

[174] Francesca Menna, E Nocerino, Daniele Morabito, EM Farella, M Perini, and F Remondino. An open source low-cost automatic system for image-based 3d digitization. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 42:155–162, 2017.

[175] Gal Metzer, Elad Richardson, Or Patashnik, Raja Giryes, and Daniel Cohen-Or. Latent-nerf for shape-guided generation of 3d shapes and textures. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12663–12673, 2023.

[176] Zhenxing Mi, Chang Di, and Dan Xu. Generalized binary search network for highly-efficient multi-view stereo. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12991–13000, 2022.

[177] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 405–421. Springer, 2020.

[178] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021.

[179] Niluthpol Chowdhury Mithun, Tuan Pham, Qiao Wang, Ben Southall, Kshitij Minhas, Bogdan Matei, Stephan Mandt, Supun Samarasekera, and Rakesh Kumar. Diffusion-guided gaussian splatting for large-scale unconstrained 3d reconstruction and novel view synthesis. *arXiv preprint arXiv:2504.01960*, 2025.

[180] Arthur Moreau, Jifei Song, Helisa Dhamo, Richard Shaw, Yiren Zhou, and Eduardo Pérez-Pellitero. Human gaussian splatting: Real-time rendering of animatable avatars. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 788–798, 2024.

[181] L Morelli, A Karami, F Menna, and F Remondino. Orientation of images with low contrast textures and transparent objects. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 48:77–84, 2022.

[182] Hugo Moreno, Adrià Gómez, Sergio Altares-López, Angela Ribeiro, and Dionisio Andújar. Analysis of stable diffusion-derived fake weeds performance for training convolutional neural networks. *Computers and Electronics in Agriculture*, 214:108324, 2023.

[183] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM transactions on graphics (TOG)*, 41(4):1–15, 2022.

[184] Jacob Munkberg, Jon Hasselgren, Tianchang Shen, Jun Gao, Wenzheng Chen, Alex Evans, Thomas Müller, and Sanja Fidler. Extracting triangular 3d models,

materials, and lighting from images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8280–8290, 2022.

[185] Liangliang Nan. Mvstudio. `https://github.com/LiangliangNan/MVStudio`, 2018. Accessed: November 1, 2023.

[186] Slavcho Neshev, Krasimir Tonchev, Agata Manolova, and Vladimir Poulkov. 3d scene segmentation. a comprehensive survey and open problems. *IEEE Access*, 2025.

[187] F Nex, M Gerke, Fabio Remondino, H-J Przybilla, M Bäumker, and A Zurhorst. Isprs benchmark for multi-platform photogrammetry. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 2:135–142, 2015.

[188] Junfeng Ni, Yixin Chen, Bohan Jing, Nan Jiang, Bin Wang, Bo Dai, Yixin Zhu, Song-Chun Zhu, and Siyuan Huang. Phyrecon: Physically plausible neural scene reconstruction. *arXiv preprint arXiv:2404.16666*, 2024.

[189] Michael Niemeyer, Jonathan T Barron, Ben Mildenhall, Mehdi SM Sajjadi, Andreas Geiger, and Noha Radwan. Regnerf: Regularizing neural radiance fields for view synthesis from sparse inputs. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5480–5490, 2022.

[190] Michael Oechsle, Songyou Peng, and Andreas Geiger. Unisurf: Unifying neural implicit surfaces and radiance fields for multi-view reconstruction. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5589–5599, 2021.

[191] Nazanin Padkan, Ziyan Yan, and Fabio Remondino. Evaluating monocular depth estimation methods on industrial objects. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 48:175–181, 2025.

[192] Mingjie Pan, Jiaming Liu, Renrui Zhang, Peixiang Huang, Xiaoqi Li, Hongwei Xie, Bing Wang, Li Liu, and Shanghang Zhang. Renderocc: Vision-centric 3d occupancy prediction with 2d rendering supervision. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 12404–12411. IEEE, 2024.

[193] Chethan M Parameshwara, Gokul Hari, Cornelia Fermüller, Nitin J Sanket, and Yiannis Aloimonos. Diffposenet: Direct differentiable camera pose estimation.

In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6845–6854, 2022.

[194] Jongsu Park, Shokhikha Amalana Murdivien, Kyungwan Choi, Duhwan Mun, and Jumyung Um. Generative 3d appearance design: A survey of generation, segmentation and editing by artificial intelligence. *Journal of Computational Design and Engineering*, 13(1):1–23, 2026.

[195] Dolores Parras-Burgos, Daniel G Fernández-Pacheco, Francisco Cavas-Martínez, José Nieto, and Francisco JF Cañavate. Initiation to reverse engineering by using activities based on photogrammetry as new teaching method in university technical studies. In *International Conference on Human-Computer Interaction*, pages 159–176. Springer, 2019.

[196] Xidong Peng, Xinge Zhu, and Yuexin Ma. Cl3d: Unsupervised domain adaptation for cross-lidar 3d detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 2047–2055, 2023.

[197] Pakkapon Phongthawee, Worameth Chinchuthakun, Nontaphat Sinsunthithet, Varun Jampani, Amit Raj, Pramook Khungurn, and Supasorn Suwajanakorn. Diffusionlight: Light probes for free by painting a chrome ball. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 98–108, 2024.

[198] Jens Piekenbrinck, Christian Schmidt, Alexander Hermans, Narunas Vaskevicius, Timm Linder, and Bastian Leibe. Opensplat3d: Open-vocabulary 3d instance segmentation using gaussian splatting. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 5246–5255, 2025.

[199] Dean A Pomerleau. Alvinn: An autonomous land vehicle in a neural network. *Advances in neural information processing systems*, 1, 1988.

[200] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988*, 2022.

[201] Jiating Qian, Yiming Yan, Fengjiao Gao, Baoyu Ge, Maosheng Wei, Boyi Shangguan, and Guangjun He. C3dgs: Compressing 3d gaussian model for surface reconstruction of large-scale scenes based on multiview uav images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 18:4396–4409, 2025.

[202] Shengyi Qian, Kaichun Mo, Valts Blukis, David F Fouhey, Dieter Fox, and Ankit Goyal. 3d-mvp: 3d multiview pretraining for manipulation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 22530–22539, 2025.

[203] Zhiyin Qian, Shaofei Wang, Marko Mihajlovic, Andreas Geiger, and Siyu Tang. 3dgs-avatar: Animatable avatars via deformable 3d gaussian splatting. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5020–5030, 2024.

[204] Kai Qin, Jing Li, Sisi Zlatanova, Haitao Wu, Yin Gao, Yuchen Li, Sizhe Shen, Xiangjun Qu, Zhiyuan Yang, Zhenxin Zhang, et al. Novel uav-based 3d reconstruction using dense lidar point cloud and imagery: A geometry-aware 3d gaussian splatting approach. *International Journal of Applied Earth Observation and Geoinformation*, 140:104590, 2025.

[205] Yansong Qu, Dian Chen, Xinyang Li, Xiaofan Li, Shengchuan Zhang, Liujuan Cao, and Rongrong Ji. Drag your gaussian: Effective drag-based editing with score distillation for 3d gaussian splatting. In *Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers*, pages 1–12, 2025.

[206] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

[207] Amit Raj, Srinivas Kaza, Ben Poole, Michael Niemeyer, Nataniel Ruiz, Ben Mildenhall, Shiran Zada, Kfir Aberman, Michael Rubinstein, Jonathan Barron, et al. Dreambooth3d: Subject-driven text-to-3d generation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2349–2359, 2023.

[208] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 12179–12188, 2021.

[209] René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE transactions on pattern analysis and machine intelligence*, 44(3):1623–1637, 2020.

[210] Dimitar Rangelov, Sierd Waanders, Kars Waanders, Maurice van Keulen, and Radoslav Miltchev. 3d reconstruction techniques and the impact of lighting conditions on reconstruction quality: A comprehensive review. 2025.

[211] Nathan D Ratliff, J Andrew Bagnell, and Martin A Zinkevich. Maximum margin planning. In *Proceedings of the 23rd international conference on Machine learning*, pages 729–736, 2006.

[212] Fabio Remondino. Heritage recording and 3d modeling with photogrammetry and 3d scanning. *Remote sensing*, 3(6):1104–1138, 2011.

[213] Fabio Remondino, Ali Karami, Ziyang Yan, Gabriele Mazzacca, Simone Rigon, and Rongjun Qin. A critical analysis of nerf-based 3d reconstruction. *Remote Sensing*, 15(14):3585, 2023.

[214] Fabio Remondino, Maria Grazia Spera, Erica Nocerino, Fabio Menna, and Francesco Nex. State of the art in high density image matching. *The photogrammetric record*, 29(146):144–166, 2014.

[215] Kerui Ren, Lihan Jiang, Tao Lu, Mulin Yu, Linning Xu, Zhangkai Ni, and Bo Dai. Octree-gs: Towards consistent real-time rendering with lod-structured 3d gaussians. *arXiv preprint arXiv:2403.17898*, 2024.

[216] Tianhe Ren, Shilong Liu, Ailing Zeng, Jing Lin, Kunchang Li, He Cao, Jiayu Chen, Xinyu Huang, Yukang Chen, Feng Yan, et al. Grounded sam: Assembling open-world models for diverse visual tasks. *arXiv preprint arXiv:2401.14159*, 2024.

[217] Barbara Roessle, Jonathan T Barron, Ben Mildenhall, Pratul P Srinivasan, and Matthias Nießner. Dense depth priors for neural radiance fields from sparse input views. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12892–12901, 2022.

[218] Barbara Roessle, Norman Müller, Lorenzo Porzi, Samuel Rota Bulo, Peter Kontschieder, and Matthias Nießner. Ganerf: Leveraging discriminators to optimize neural radiance fields. *ACM Transactions on Graphics (TOG)*, 42(6):1–14, 2023.

[219] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Pro-*

ceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 10684–10695, 2022.

[220] Dennis Rotondi, Fabio Scaparro, Hermann Blum, and Kai O Arras. Fungraph: Functionality aware 3d scene graphs for language-prompted scene interaction. In *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4083–4090. IEEE, 2025.

[221] David Rozenberszki, Or Litany, and Angela Dai. Language-grounded indoor 3d semantic segmentation in the wild. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022.

[222] Cong Ruan, Yuesong Wang, Tao Guan, Bin Zhang, and Lili Ju. Indoorgs: Geometric cues guided gaussian splatting for indoor scene reconstruction. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 844–853, 2025.

[223] Viktor Rudnev, Mohamed Elgharib, William Smith, Lingjie Liu, Vladislav Golyanik, and Christian Theobalt. Nerf for outdoor scene relighting. In *European Conference on Computer Vision*, pages 615–631. Springer, 2022.

[224] Abbas Sadat, Sergio Casas, Mengye Ren, Xinyu Wu, Pranaab Dhawan, and Raquel Urtasun. Perceive, predict, and plan: Safe motion planning through interpretable semantic representations. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIII 16*, pages 414–430. Springer, 2020.

[225] Chandan K Sahu, Crystal Young, and Rahul Rai. Artificial intelligence (ai) in augmented reality (ar)-assisted manufacturing applications: a review. *International journal of production research*, 59(16):4903–4959, 2021.

[226] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4104–4113, 2016.

[227] Jonas Schult, Francis Engelmann, Alexander Hermans, Or Litany, Siyu Tang, and Bastian Leibe. Mask3d: Mask transformer for 3d semantic instance segmentation. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 8216–8223. IEEE, 2023.

[228] Yihua Shao, Haojin He, Sijie Li, Siyu Chen, Xinwei Long, Fanhu Zeng, Yuxuan Fan, Muyang Zhang, Ziyang Yan, Ao Ma, et al. Eventvad: Training-free event-aware video anomaly detection. *arXiv preprint arXiv:2504.13092*, 2025.

[229] Yihua Shao, Siyu Liang, Zijian Ling, Minxi Yan, Haiyang Liu, Siyu Chen, Ziyang Yan, Chenyu Zhang, Haotong Qin, Michele Magno, et al. Gwq: Gradient-aware weight quantization for large language models. *arXiv preprint arXiv:2411.00850*, 2024.

[230] Yihua Shao, Deyang Lin, Fanhu Zeng, Minxi Yan, Muyang Zhang, Siyu Chen, Yuxuan Fan, Ziyang Yan, Haozhe Wang, Jingcai Guo, et al. Tr-dq: Time-rotation diffusion quantization. *arXiv preprint arXiv:2503.06564*, 2025.

[231] Yihua Shao, Yeling Xu, Xinwei Long, Siyu Chen, Ziyang Yan, Yang Yang, Haoting Liu, Yan Wang, Hao Tang, and Zhen Lei. Accidentblip: Agent of accident warning based on ma-former. *arXiv preprint arXiv:2404.12149*, 2024.

[232] Qiuhong Shen, Xingyi Yang, and Xinchao Wang. Flashsplat: 2d to 3d gaussian splatting segmentation solved optimally. In *European Conference on Computer Vision*, pages 456–472. Springer, 2025.

[233] Vinita Shukla, Amit Shukla, Surya Prakash SK, and Shraddha Shukla. A systematic survey: role of deep learning-based image anomaly detection in industrial inspection contexts. *Frontiers in Robotics and AI*, 12:1554196, 2025.

[234] Nagabhushan Somraj, Adithyan Karanayil, and Rajiv Soundararajan. Simplenerf: Regularizing sparse input neural radiance fields with simpler solutions. In *SIGGRAPH Asia 2023 Conference Papers*, pages 1–11, 2023.

[235] Chan Hee Song, Valts Blukis, Jonathan Tremblay, Stephen Tyree, Yu Su, and Stan Birchfield. Robospatial: Teaching spatial understanding to 2d and 3d vision-language models for robotics. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 15768–15780, 2025.

[236] Hyeonseop Song, Seokhun Choi, Hoseok Do, Chul Lee, and Taehyeong Kim. Blending-nerf: Text-driven localized editing in neural radiance fields. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14383–14393, 2023.

[237] Liangchen Song, Zhong Li, Xuan Gong, Lele Chen, Zhang Chen, Yi Xu, and Junsong Yuan. Harnessing low-frequency neural fields for few-shot view synthesis. *arXiv preprint arXiv:2303.08370*, 2023.

[238] Ziying Song, Lin Liu, Feiyang Jia, Yadan Luo, Caiyan Jia, Guoxin Zhang, Lei Yang, and Li Wang. Robustness-aware 3d object detection in autonomous driving: A review and outlook. *IEEE Transactions on Intelligent Transportation Systems*, 25(11):15407–15436, 2024.

[239] Elisavet Konstantina Stathopoulou and Fabio Remondino. A survey on conventional and learning-based methods for multi-view stereo. *The Photogrammetric Record*, 38(183):374–407, 2023.

[240] Kewei Sui, Anindita Ghosh, Inwoo Hwang, Bing Zhou, Jian Wang, and Chuan Guo. A survey on human interaction motion generation. *International Journal of Computer Vision*, 134(3):113, 2026.

[241] Chunyi Sun, Yanbin Liu, Junlin Han, and Stephen Gould. Nerfeditor: Differentiable style decomposition for 3d scene editing. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 7306–7315, 2024.

[242] Ciliang Sun, Yuqi Li, Jiabao Li, Chong Wang, and Xinmiao Dai. Case-nerf: Camera settings editing of neural radiance fields. In *Computer Graphics International Conference*, pages 95–107. Springer, 2023.

[243] Guodong Sun, Dingjie Liu, Zeyu Yang, Shaoran An, and Yang Zhang. 3d reconstruction and precision evaluation of industrial components via gaussian splatting. *Computers & Graphics*, page 104422, 2025.

[244] Jiakai Sun, Han Jiao, Guangyuan Li, Zhanjie Zhang, Lei Zhao, and Wei Xing. 3dgstream: On-the-fly training of 3d gaussians for efficient streaming of photo-realistic free-viewpoint videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20675–20685, 2024.

[245] Tianyu Sun, Guodong Zhang, Wenming Yang, Jing-Hao Xue, and Guijin Wang. Trosd: A new rgb-d dataset for transparent and reflective object segmentation in practice. *IEEE Transactions on Circuits and Systems for Video Technology*, 33(10):5721–5733, 2023.

[246] Wei Sun, Yuan Li, and Jianbin Jiao. Cags: Open-vocabulary 3d scene understanding with context-aware gaussian splatting. *Image and Vision Computing*, 165:105830, 2026.

[247] Richard S Sutton. Integrated architectures for learning, planning, and reacting based on approximating dynamic programming. In *Machine learning proceedings 1990*, pages 216–224. Elsevier, 1990.

[248] Matthew Tancik, Vincent Casser, Xinchen Yan, Sabeek Pradhan, Ben Mildenhall, Pratul P Srinivasan, Jonathan T Barron, and Henrik Kretzschmar. Block-nerf: Scalable large scene neural view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8248–8258, 2022.

[249] Matthew Tancik, Ethan Weber, Evonne Ng, Ruilong Li, Brent Yi, Terrance Wang, Alexander Kristoffersen, Jake Austin, Kamyar Salahi, Abhik Ahuja, et al. Nerfstudio: A modular framework for neural radiance field development. In *ACM SIGGRAPH 2023 conference proceedings*, pages 1–12, 2023.

[250] Jiadong Tang, Yu Gao, Dianyi Yang, Liqi Yan, Yufeng Yue, and Yi Yang. Dronesplat: 3d gaussian splatting for robust 3d reconstruction from in-the-wild drone imagery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 833–843, 2025.

[251] Jiaxiang Tang, Zhaoxi Chen, Xiaokang Chen, Tengfei Wang, Gang Zeng, and Ziwei Liu. Lgm: Large multi-view gaussian model for high-resolution 3d content creation. In *European Conference on Computer Vision*, pages 1–18. Springer, 2024.

[252] Jiaxiang Tang, Jiawei Ren, Hang Zhou, Ziwei Liu, and Gang Zeng. Dreamgaussian: Generative gaussian splatting for efficient 3d content creation. *arXiv preprint arXiv:2309.16653*, 2023.

[253] Junshu Tang, Tengfei Wang, Bo Zhang, Ting Zhang, Ran Yi, Lizhuang Ma, and Dong Chen. Make-it-3d: High-fidelity 3d creation from a single image with diffusion prior. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 22819–22829, 2023.

[254] Kaiyuan Tang, Kuangshi Ai, Jun Han, and Chaoli Wang. Texgs-volvis: Expressive scene editing for volume visualization via textured gaussian splatting. *IEEE Transactions on Visualization and Computer Graphics*, 2025.

[255] Siyu Teng, Xuemin Hu, Peng Deng, Bai Li, Yuchen Li, Yunfeng Ai, Dongsheng Yang, Lingxi Li, Zhe Xuanyuan, Fenghua Zhu, et al. Motion planning for autonomous driving: The state of the art and future perspectives. *IEEE Transactions on Intelligent Vehicles*, 8(6):3692–3711, 2023.

[256] Xiaoyu Tian, Junru Gu, Bailin Li, Yicheng Liu, Chenxu Hu, Yang Wang, Kun Zhan, Peng Jia, Xianpeng Lang, and Hang Zhao. Drivevlm: The convergence of autonomous driving and large vision-language models. *arXiv preprint arXiv:2402.12289*, 2024.

[257] Xiaoyu Tian, Tao Jiang, Longfei Yun, Yucheng Mao, Huitong Yang, Yue Wang, Yilun Wang, and Hang Zhao. Occ3d: A large-scale 3d occupancy prediction benchmark for autonomous driving. *Advances in Neural Information Processing Systems*, 36, 2024.

[258] Jinguang Tong, Xuesong Li, Fahira Afzal Maken, Sundaram Muthu, Lars Petersson, Chuong Nguyen, and Hongdong Li. Gs-2dgs: Geometrically supervised 2dgs for reflective object reconstruction. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 21547–21557, 2025.

[259] Wenwen Tong, Chonghao Sima, Tai Wang, Li Chen, Silei Wu, Hanming Deng, Yi Gu, Lewei Lu, Ping Luo, Dahua Lin, et al. Scene as occupancy. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8406–8415, 2023.

[260] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

[261] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

[262] Prune Truong, Martin Danelljan, Luc V Gool, and Radu Timofte. Gocor: Bringing globally optimized correspondence volumes into your neural network. *Advances in Neural Information Processing Systems*, 33:14278–14290, 2020.

[263] Prune Truong, Martin Danelljan, and Radu Timofte. Glu-net: Global-local universal network for dense flow and correspondences. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6258–6268, 2020.

[264] Paschalis Tsirtsakis, Georgios Zacharis, George S Maraslidis, and George F Fragulis. Deep learning for object recognition: A comprehensive review of models and algorithms. *International Journal of Cognitive Computing in Engineering*, 6:298–312, 2025.

[265] Haithem Turki, Deva Ramanan, and Mahadev Satyanarayanan. Mega-nerf: Scalable construction of large-scale nerfs for virtual fly-throughs. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12922–12931, 2022.

[266] Matias Turkulainen, Xuqian Ren, Iaroslav Melekhov, Otto Seiskari, Esa Rahtu, and Juho Kannala. Dn-splatter: Depth and normal priors for gaussian splatting and meshing. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2025.

[267] Cyrus Vachha and Ayaan Haque. Instruct-gs2gs: Editing 3d gaussian splats with instructions, 2024.

[268] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

[269] Dor Verbin, Peter Hedman, Ben Mildenhall, Todd Zickler, Jonathan T. Barron, and Pratul P. Srinivasan. Ref-nerf: Structured view-dependent appearance for neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5481–5490. IEEE, 2022.

[270] Sudheendra Vijayanarasimhan, Susanna Ricco, Cordelia Schmid, Rahul Sukthankar, and Katerina Fragkiadaki. Sfm-net: Learning of structure and motion from video. *arXiv preprint arXiv:1704.07804*, 2017.

[271] Matthew Luke Vincent, Chance Coughenour, Fabio Remondino, Mariano Flores Gutierrez, Victor Manuel Lopez-Menchero Bendicho, and Dieter Fritsch. Rekrei: A public platform for digitally preserving lost heritage. In *CAA 2016*, 2015.

[272] Fangjinhua Wang, Silvano Galliani, Christoph Vogel, Pablo Speciale, and Marc Pollefeys. Patchmatchnet: Learned multi-view patchmatch stereo. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14194–14203, 2021.

[273] Guangcong Wang, Zhaoxi Chen, Chen Change Loy, and Ziwei Liu. Sparsenerf: Distilling depth ranking for few-shot novel view synthesis. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9065–9076, 2023.

[274] Hao Wang and Minghui Li. A new era of indoor scene reconstruction: A survey. *IEEE Access*, 12:110160–110192, 2024.

[275] Haozhe Wang, Chao Du, Panyan Fang, Li He, Liang Wang, and Bo Zheng. Adversarial constrained bidding via minimax regret optimization with causality-aware reinforcement learning. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 2314–2325, 2023.

[276] Jiepeng Wang, Yuan Liu, Peng Wang, Cheng Lin, Junhui Hou, Xin Li, Taku Komura, and Wenping Wang. Gaussurf: Geometry-guided 3d gaussian splatting for surface reconstruction. *arXiv preprint arXiv:2411.19454*, 2024.

[277] Jiepeng Wang, Peng Wang, Xiaoxiao Long, Christian Theobalt, Taku Komura, Lingjie Liu, and Wenping Wang. Neuris: Neural reconstruction of indoor scenes using normal priors. In *European Conference on Computer Vision*, pages 139–155. Springer, 2022.

[278] Junjie Wang, Jiemin Fang, Xiaopeng Zhang, Lingxi Xie, and Qi Tian. Gaussianeditor: Editing 3d gaussians delicately with text instructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20902–20911, 2024.

[279] Nan Wang, Yuantao Chen, Lixing Xiao, Weiqing Xiao, Bohan Li, Zhaoxi Chen, Chongjie Ye, Shaocong Xu, Saining Zhang, Ziyang Yan, et al. Unifying appearance codes and bilateral grids for driving scene gaussian splatting. *arXiv preprint arXiv:2506.05280*, 2025.

[280] Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du,

Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024.

[281] Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. *arXiv preprint arXiv:2106.10689*, 2021.

[282] Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 20697–20709, 2024.

[283] Xiang Wang, Chen Wang, Bing Liu, Xiaoqing Zhou, Liang Zhang, Jin Zheng, and Xiao Bai. Multi-view stereo in the deep learning era: A comprehensive review. *Displays*, 70:102102, 2021.

[284] Xihan Wang, Dianyi Yang, Yu Gao, Yufeng Yue, Yi Yang, and Mengyin Fu. Gaussiangraph: 3d gaussian-based scene graph generation for open-world scene understanding. In *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4091–4098. IEEE, 2025.

[285] Yifan Wang, Felice Serena, Shihao Wu, Cengiz Öztireli, and Olga Sorkine-Hornung. Differentiable surface splatting for point-based geometry processing. *ACM Transactions on Graphics (TOG)*, 38(6):1–14, 2019.

[286] Yuqi Wang, Yuntao Chen, Xingyu Liao, Lue Fan, and Zhaoxiang Zhang. Panoocc: Unified occupancy representation for camera-based 3d panoptic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 17158–17168, 2024.

[287] Yuze Wang, Junyi Wang, Yansong Qu, and Yue Qi. Rip-nerf: Learning rotation-invariant point-based neural radiance field for fine-grained editing and compositing. In *Proceedings of the 2023 ACM international conference on multimedia retrieval*, pages 125–134, 2023.

[288] Zhaonan Wang, Manyi Li, and Changhe Tu. Ag2aussian: Anchor-graph structured gaussian splatting for instance-level 3d scene understanding and editing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 26806–26816, 2025.

[289] Zirui Wang, Shangzhe Wu, Weidi Xie, Min Chen, and Victor Adrian Prisacariu. Nerf–: Neural radiance fields without known camera parameters. 2021.

[290] Bob Wei, Mengye Ren, Wenyuan Zeng, Ming Liang, Bin Yang, and Raquel Urtasun. Perceive, attend, and drive: Learning spatial attention for safe self-driving. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4875–4881. IEEE, 2021.

[291] Minghao Wen, Shengjie Wu, Kangkan Wang, and Dong Liang. Intergsedit: Interactive 3d gaussian splatting editing with 3d geometry-consistent attention prior. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 26136–26145, 2025.

[292] Suttisak Wizadwongsa, Pakkapon Phongthawee, Jiraphon Yenphraphai, and Supasorn Suwajanakorn. Nex: Real-time view synthesis with neural basis expansion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8534–8543, 2021.

[293] Bin-Shih Wu, Hong-En Chen, Sheng-Yu Huang, and Yu-Chiang Frank Wang. Tpa3d: Triplane attention for fast text-to-3d generation. In *European Conference on Computer Vision*, pages 438–455. Springer, 2025.

[294] Guanjun Wu, Taoran Yi, Jiemin Fang, Lingxi Xie, Xiaopeng Zhang, Wei Wei, Wenyu Liu, Qi Tian, and Xinggang Wang. 4d gaussian splatting for real-time dynamic scene rendering. *arXiv preprint arXiv:2310.08528*, 2023.

[295] Jianzong Wu, Xiangtai Li, Shilin Xu, Haobo Yuan, Henghui Ding, Yibo Yang, Xia Li, Jiangning Zhang, Yunhai Tong, Xudong Jiang, et al. Towards open vocabulary learning: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 46(7):5092–5113, 2024.

[296] Jing Wu, Jia-Wang Bian, Xinghui Li, Guangrun Wang, Ian Reid, Philip Torr, and Victor Adrian Prisacariu. Gaussctrl: multi-view consistent text-driven 3d gaussian splatting editing. *arXiv preprint arXiv:2403.08733*, 2024.

[297] Rundi Wu, Ben Mildenhall, Philipp Henzler, Keunhong Park, Ruiqi Gao, Daniel Watson, Pratul P Srinivasan, Dor Verbin, Jonathan T Barron, Ben Poole, et al. Reconfusion: 3d reconstruction with diffusion priors. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 21551–21561, 2024.

[298] Yongchang Wu, Zipeng Qi, Zhenwei Shi, and Zhengxia Zou. Blockgaussian: Efficient large-scale scene novel view synthesis via adaptive block-based gaussian splatting. *arXiv preprint arXiv:2504.09048*, 2025.

[299] Jamie Wynn and Daniyar Turmukhambetov. Diffusionerf: Regularizing neural radiance fields with denoising diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4180–4189, 2023.

[300] Haodong Xiang, Xinghui Li, Kai Cheng, Xiansong Lai, Wanting Zhang, Zhichao Liao, Long Zeng, and Xueping Liu. Gaussianroom: Improving 3d gaussian splatting with sdf guidance and monocular cues for indoor scene reconstruction. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2686–2693. IEEE, 2025.

[301] Hanyuan Xiao, Yingshu Chen, Huajian Huang, Haolin Xiong, Jing Yang, Pratusha Prasad, and Yajie Zhao. Localized gaussian splatting editing with contextual awareness. *arXiv preprint arXiv:2408.00083*, 2024.

[302] Hanyuan Xiao, Yingshu Chen, Huajian Huang, Haolin Xiong, Jing Yang, Pratusha Prasad, and Yajie Zhao. Localized gaussian splatting editing with contextual awareness. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 5207–5217. IEEE, 2025.

[303] Peiyao Xie and Xi-En Cheng. Efficient 3d gaussian splatting surface rendering for reflective objects. In *Fourth International Conference on Machine Vision, Automatic Identification, and Detection (MVAID 2025)*, volume 13793, pages 581–588. SPIE, 2025.

[304] Xiaoyan Xing, Konrad Groh, Sezer Karaoglu, Theo Gevers, and Anand Bhattad. Luminet: Latent intrinsics meets diffusion models for indoor scene relighting. *CVPR*, 2025. arXiv:2412.00177.

[305] Haolin Xiong, Sairisheek Muttukuru, Hanyuan Xiao, Rishi Upadhyay, Pradyumna Chari, Yajie Zhao, and Achuta Kadambi. Sparsegs: sparse view synthesis using 3d gaussian splatting. In *2025 International Conference on 3D Vision (3DV)*, pages 1032–1041. IEEE, 2025.

[306] Jiacong Xu, Yiqun Mei, and Vishal Patel. Wild-gs: Real-time novel view synthesis from unconstrained photo collections. *NIPS*, 37:103334–103355, 2024.

[307] Jiale Xu, Shenghua Gao, and Ying Shan. Freesplatter: Pose-free gaussian splatting for sparse-view 3d reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 25442–25452, 2025.

[308] Qingshan Xu and Wenbing Tao. Multi-scale geometric consistency guided multi-view stereo. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5483–5492, 2019.

[309] Zhiyuan Xu, Kun Wu, Junjie Wen, Jinming Li, Ning Liu, Zhengping Che, and Jian Tang. A survey on robotics with foundation models: toward embodied ai. *arXiv preprint arXiv:2402.02385*, 2024.

[310] Chi Yan, Delin Qu, Dan Xu, Bin Zhao, Zhigang Wang, Dong Wang, and Xuelong Li. Gs-slam: Dense visual slam with 3d gaussian splatting. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 19595–19604, 2024.

[311] Ziyang Yan, Wenzhen Dong, Yihua Shao, Yuhang Lu, Liu Haiyang, Jingwen Liu, Haozhe Wang, Zhe Wang, Yan Wang, Fabio Remondino, et al. Renderworld: World model with self-supervised 3d label. *arXiv preprint arXiv:2409.11356*, 2024.

[312] Ziyang Yan, Lei Li, Yihua Shao, Siyu Chen, Zongkai Wu, Jenq-Neng Hwang, Hao Zhao, and Fabio Remondino. 3dsceneeditor: Controllable 3d scene editing with gaussian splatting. *arXiv preprint arXiv:2412.01583*, 2024.

[313] Ziyang Yan, Xingyu Liu, and Fabio Remondino. Colla-gaussian: High-quality surface reconstruction of non-collaborative objects with gaussian splatting. Submitted to a peer-reviewed journal, 2026.

[314] Ziyang Yan, Gabriele Mazzacca, Simone Rigon, Elisa Mariarosaria Farella, Pawel Trybala, Fabio Remondino, et al. Nerfbk: a holistic dataset for benchmarking nerf-based 3d reconstruction. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 48(1):219–226, 2023.

[315] Ziyang Yan, Nazanin Padkan, Paweł Trybała, Elisa Mariarosaria Farella, and Fabio Remondino. Learning-based 3d reconstruction methods for non-collaborative surfaces—a metrological evaluation. *Metrology*, 5(2):20, 2025.

[316] Ziyang Yan, Mengrui Yin, Yihua Shao, and Fabio Remondino. Evaluating 3d gaussian splatting for urban scene reconstruction. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 48:251–258, 2025.

[317] Chenyu Yang, Yuntao Chen, Hao Tian, Chenxin Tao, Xizhou Zhu, Zhaoxiang Zhang, Gao Huang, Hongyang Li, Yu Qiao, Lewei Lu, et al. Bevformer v2: Adapting modern image backbones to bird's-eye-view recognition via perspective supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17830–17839, 2023.

[318] Dianyi Yang, Yu Gao, Xihan Wang, Yufeng Yue, Yi Yang, and Mengyin Fu. Opengs-slam: Open-set dense semantic slam with 3d gaussian splatting for object-level scene understanding. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 8486–8492. IEEE, 2025.

[319] Huitong Yang, Xuyang Bai, Xinge Zhu, and Yuexin Ma. One training for multiple deployments: Polar-based adaptive bev perception for autonomous driving. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5602–5609. IEEE, 2023.

[320] Lihe Yang, Bingyi Kang, Zilong Huang, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything: Unleashing the power of large-scale unlabeled data. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10371–10381, 2024.

[321] Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. *arXiv:2406.09414*, 2024.

[322] Ziyi Yang, Xinyu Gao, Yangtian Sun, Yihua Huang, Xiaoyang Lyu, Wen Zhou, Shaohui Jiao, Xiaojuan Qi, and Xiaogang Jin. Spec-gaussian: Anisotropic view-dependent appearance for 3d gaussian splatting. *arXiv preprint arXiv:2402.15870*, 2024.

[323] Ziyi Yang, Xinyu Gao, Wen Zhou, Shaohui Jiao, Yuqing Zhang, and Xiaogang Jin. Deformable 3d gaussians for high-fidelity monocular dynamic scene reconstruction. *arXiv preprint arXiv:2309.13101*, 2023.

[324] Yao Yao, Zixin Luo, Shiwei Li, Jingyang Zhang, Yufan Ren, Lei Zhou, Tian Fang, and Long Quan. Blendedmvs: A large-scale dataset for generalized multi-view

stereo networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1790–1799, 2020.

[325] Lior Yariv, Jiatao Gu, Yoni Kasten, and Yaron Lipman. Volume rendering of neural implicit surfaces. *Advances in neural information processing systems*, 34:4805–4815, 2021.

[326] Jianglong Ye, Peng Wang, Kejie Li, Yichun Shi, and Heng Wang. Consistent-1-to-3: Consistent image to 3d view synthesis via geometry-aware diffusion models. In *2024 International Conference on 3D Vision (3DV)*, pages 664–674. IEEE, 2024.

[327] Kai Ye, Chong Gao, Guanbin Li, Wenzheng Chen, and Baoquan Chen. Geosplatting: Towards geometry guided gaussian splatting for physically-based inverse rendering. *arXiv preprint arXiv:2410.24204*, 2024.

[328] Mingqiao Ye, Martin Danelljan, Fisher Yu, and Lei Ke. Gaussian grouping: Segment and edit anything in 3d scenes. *arXiv preprint arXiv:2312.00732*, 2023.

[329] Lin Yen-Chen, Pete Florence, Jonathan T Barron, Tsung-Yi Lin, Alberto Rodriguez, and Phillip Isola. Nerf-supervision: Learning dense object descriptors from neural radiance fields. In *2022 international conference on robotics and automation (ICRA)*, pages 6496–6503. IEEE, 2022.

[330] Chandan Yeshwanth, Yueh-Cheng Liu, Matthias Nießner, and Angela Dai. Scannet++: A high-fidelity dataset of 3d indoor scenes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12–22, 2023.

[331] Taoran Yi, Jiemin Fang, Junjie Wang, Guanjun Wu, Lingxi Xie, Xiaopeng Zhang, Wenyu Liu, Qi Tian, and Xinggang Wang. Gaussiandreamer: Fast generation from text to 3d gaussians by bridging 2d and 3d diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6796–6807, 2024.

[332] Taoran Yi, Jiemin Fang, Guanjun Wu, Lingxi Xie, Xiaopeng Zhang, Wenyu Liu, Qi Tian, and Xinggang Wang. Gaussiandreamer: Fast generation from text to 3d gaussian splatting with point cloud priors. *arXiv preprint arXiv:2310.08529*, 2023.

[333] Taoran Yi, Jiemin Fang, Zanwei Zhou, Junjie Wang, Guanjun Wu, Lingxi Xie, Xiaopeng Zhang, Wenyu Liu, Xinggang Wang, and Qi Tian. Gaussiandreamerpro:

Text to manipulable 3d gaussians with highly enhanced quality. *arXiv preprint arXiv:2406.18462*, 2024.

[334] Wei Yin, Chi Zhang, Hao Chen, Zhipeng Cai, Gang Yu, Kaixuan Wang, Xiaozhi Chen, and Chunhua Shen. Metric3d: Towards zero-shot metric 3d prediction from a single image. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9043–9053, 2023.

[335] Tanveer Younis and Zhanglin Cheng. Sparse-view 3d reconstruction: Recent advances and open challenges. *arXiv preprint arXiv:2507.16406*, 2025.

[336] Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelnerf: Neural radiance fields from one or few images. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4578–4587, 2021.

[337] Chaohui Yu, Qiang Zhou, Jingliang Li, Zhe Zhang, Zhibin Wang, and Fan Wang. Points-to-3d: Bridging the gap between sparse points and shape-controllable text-to-3d generation. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 6841–6850, 2023.

[338] Weihao Yu, Yuanhao Cai, Ruyi Zha, Zhiwen Fan, Chenxin Li, and Yixuan Yuan. X2-gaussian: 4d radiative gaussian splatting for continuous-time tomographic reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 24728–24738, 2025.

[339] Zehao Yu, Anpei Chen, Bozidar Antic, Songyou Peng, Apratim Bhattacharyya, Michael Niemeyer, Siyu Tang, Torsten Sattler, and Andreas Geiger. Sdfstudio: A unified framework for surface reconstruction. *Colophon This document was typeset using LATEX, using the KOMA-Script class scrbook. The main font is Palatino*, 2022.

[340] Zehao Yu, Songyou Peng, Michael Niemeyer, Torsten Sattler, and Andreas Geiger. Monosdf: Exploring monocular geometric cues for neural implicit surface reconstruction. *Advances in neural information processing systems*, 35:25018–25032, 2022.

[341] Zehao Yu, Torsten Sattler, and Andreas Geiger. Gaussian opacity fields: Efficient adaptive surface reconstruction in unbounded scenes. *ACM TOG*, 43(6):1–13, 2024.

[342] Yu-Jie Yuan, Yang-Tian Sun, Yu-Kun Lai, Yuewen Ma, Rongfei Jia, and Lin Gao. Nerf-editing: geometry editing of neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18353–18364, 2022.

[343] Zhenlong Yuan, Xiangyan Qu, Chengxuan Qian, Rui Chen, Jing Tang, Lei Sun, Xiangxiang Chu, Dapeng Zhang, Yiwei Wang, Yujun Cai, et al. Videostar: Reinforcing open-vocabulary action recognition with tools. *arXiv preprint arXiv:2510.08480*, 2025.

[344] Zhenlong Yuan, Jing Tang, Jinguo Luo, Rui Chen, Chengxuan Qian, Lei Sun, Xiangxiang Chu, Yujun Cai, Dapeng Zhang, and Shuo Li. Autodrive-r2: Incentivizing reasoning and self-reflection capacity for vla model in autonomous driving. *arXiv preprint arXiv:2509.01944*, 2025.

[345] Zhensheng Yuan, Haozhi Huang, Zhen Xiong, Di Wang, and Guanghua Yang. Robust and efficient 3d gaussian splatting for urban scene reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 26209–26219, 2025.

[346] Zhihao Yuan, Jinke Ren, Chun-Mei Feng, Hengshuang Zhao, Shuguang Cui, and Zhen Li. Visual programming for zero-shot open-vocabulary 3d visual grounding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20623–20633, 2024.

[347] Vladimir Yugay, Yue Li, Theo Gevers, and Martin R Oswald. Gaussian-slam: Photo-realistic dense slam with gaussian splatting. *arXiv preprint arXiv:2312.10070*, 2023.

[348] Wenyuan Zeng, Wenjie Luo, Simon Suo, Abbas Sadat, Bin Yang, Sergio Casas, and Raquel Urtasun. End-to-end interpretable neural motion planner. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8660–8669, 2019.

[349] Chubin Zhang, Juncheng Yan, Yi Wei, Jiaxin Li, Li Liu, Yansong Tang, Yueqi Duan, and Jiwen Lu. Occnerf: Self-supervised multi-camera occupancy prediction with neural radiance fields. *arXiv preprint arXiv:2312.09243*, 2023.

[350] Daiwei Zhang, Gengyan Li, Jiajie Li, Mickaël Bressieux, Otmar Hilliges, Marc Pollefeys, Luc Van Gool, and Xi Wang. Egogaussian: Dynamic scene understanding from egocentric video with 3d gaussian splatting. In *2025 International Conference on 3D Vision (3DV)*, pages 1091–1102. IEEE, 2025.

[351] Qihang Zhang, Yinghao Xu, Chaoyang Wang, Hsin-Ying Lee, Gordon Wetzstein, Bolei Zhou, and Ceyuan Yang. 3DitScene: Editing any scene via language-guided disentangled gaussian splatting. In *arXiv*, 2024.

[352] Xiuming Zhang, Pratul P. Srinivasan, Boyang Deng, Paul Debevec, William T. Freeman, and Jonathan T. Barron. Nerfactor: Neural factorization of shape and reflectance under an unknown illumination. *ACM Transactions on Graphics (TOG)*, 40(6):1–18, 2021.

[353] Yancheng Zhang, Guangyu Sun, and Chen Chen. Eggs: Exchangeable 2d/3d gaussian splatting for geometry-appearance balanced novel view synthesis. *arXiv preprint arXiv:2512.02932*, 2025.

[354] Yue Zhang, Zhiyang Xu, Ying Shen, Parisa Kordjamshidi, and Lifu Huang. Spartun3d: Situated spatial understanding of 3d world in large language models. *arXiv preprint arXiv:2410.03878*, 2024.

[355] Yunpeng Zhang, Zheng Zhu, and Dalong Du. Occformer: Dual-path transformer for vision-based 3d semantic occupancy prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9433–9443, 2023.

[356] Yunpeng Zhang, Zheng Zhu, Wenzhao Zheng, Junjie Huang, Guan Huang, Jie Zhou, and Jiwen Lu. Beverse: Unified perception and prediction in birds-eye-view for vision-centric autonomous driving. *arXiv preprint arXiv:2205.09743*, 2022.

[357] Zhe Zhang, Rui Peng, Yuxi Hu, and Ronggang Wang. Geomvsnet: Learning multi-view stereo with geometry perception. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 21508–21518, 2023.

[358] Lingjun Zhao, Yandong Luo, James Hay, and Lu Gan. Shelfgaussian: Shelf-supervised open-vocabulary gaussian-based 3d scene understanding. *arXiv preprint arXiv:2512.03370*, 2025.

[359] Xiao Zhao, Bo Chen, Mingyang Sun, Dingkang Yang, Youxing Wang, Xukun Zhang, Mingcheng Li, Dongliang Kou, Xiaoyi Wei, and Lihua Zhang. Hybridocc: Nerf enhanced transformer-based multi-camera 3d occupancy prediction. *IEEE Robotics and Automation Letters*, 2024.

[360] Zhixia Zhao, Qiyue Li, Jie Li, Richang Hong, and Zhi Liu. Viewgauss: A head movement dataset for 6dof gaussian splatting video viewing. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pages 13016–13022, 2025.

[361] Wenzhao Zheng, Weiliang Chen, Yuanhui Huang, Borui Zhang, Yueqi Duan, and Jiwen Lu. Occworld: Learning a 3d occupancy world model for autonomous driving. *arXiv preprint arXiv:2311.16038*, 2023.

[362] Zhide Zhong, Jiakai Cao, Songen Gu, Sirui Xie, Liyi Luo, Hao Zhao, Guyue Zhou, Haoang Li, and Zike Yan. Structured-nerf: Hierarchical scene graph with neural representation. In *European Conference on Computer Vision*, pages 184–201. Springer, 2025.

[363] Junsheng Zhou, Weiqi Zhang, and Yu-Shen Liu. Diffgs: Functional gaussian splatting diffusion. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

[364] Linglong Zhou, Guoxin Wu, Yunbo Zuo, Xuanyu Chen, and Hongle Hu. A comprehensive review of vision-based 3d reconstruction methods. *Sensors*, 24(7):2314, 2024.

[365] Shijie Zhou, Hui Ren, Yijia Weng, Shuwang Zhang, Zhen Wang, Dejia Xu, Zhiwen Fan, Suya You, Zhangyang Wang, Leonidas Guibas, et al. Feature4x: Bridging any monocular video to 4d agentic ai with versatile gaussian feature fields. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 14179–14190, 2025.

[366] Chen-Yang Zhu, Xin-Yao Liu, Kai Xu, and Ren-Jiao Yi. A survey on 3d editing based on nerf and 3dgs. *Frontiers of Computer Science*, 20(4):2004701, 2026.

[367] Chenming Zhu, Tai Wang, Wenwei Zhang, Jiangmiao Pang, and Xihui Liu. Llava-3d: A simple yet effective pathway to empowering lmms with 3d-awareness. *arXiv preprint arXiv:2409.18125*, 2024.

[368] Xinge Zhu, Hui Zhou, Tai Wang, Fangzhou Hong, Yuexin Ma, Wei Li, Hongsheng Li, and Dahua Lin. Cylindrical and asymmetrical 3d convolution networks for lidar segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9939–9948, 2021.

[369] Zehao Zhu, Zhiwen Fan, Yifan Jiang, and Zhangyang Wang. Fsgs: Real-time few-shot view synthesis using gaussian splatting. In *European Conference on Computer Vision*, pages 145–163. Springer, 2025.

[370] Wojciech Zielonka, Timur Bagautdinov, Shunsuke Saito, Michael Zollhöfer, Justus Thies, and Javier Romero. Drivable 3d gaussian avatars. In *2025 International Conference on 3D Vision (3DV)*, pages 979–990. IEEE, 2025.

[371] Xingxing Zuo, Pouya Samangouei, Yunwen Zhou, Yan Di, and Mingyang Li. Fmgs: Foundation model embedded 3d gaussian splatting for holistic 3d scene understanding. *International Journal of Computer Vision*, 133(2):611–627, 2025.

Appendix A

Experimental Details, Additional Visualizations, and User Study of 3DSceneEditor

A.1 Implementation details

A.1.1 Implementation Details of 3D Scene Representation

3DSceneEditor processes 3D scenes reconstructed using 3D Gaussian Splatting [110]. Since ScanNet++ [330] does not provide an initial point cloud derived from Structure-from-Motion (SfM)—a crucial requirement for achieving high-quality results with 3D Gaussian Splatting—we sampled 1 million points from the ground-truth (GT) mesh as the initial point cloud [56] and fix the number of the Gaussians during training to prevent many Gaussians merge together. This ensures the geometry of the Gaussian sets are well-defined (shown in Fig. A.1), which is critical for subsequent instance segmentation tasks.

Following the default configuration of 3D Gaussian Splatting, each scene is trained for 30,000 iterations, with input images exceeding 1600 pixels in width being automatically resized to 1600 pixels for computational efficiency.

For baseline methods, which utilize Instruct-Pix2Pix [16] for scene editing, input images are resized to 512×512 pixels as required, while all other hyper-parameters are kept consistent.



Figure A.1: Visualization of geometric results for Gaussian sets with varying initializations. The figure illustrates the results of using randomly initialized points versus points sampled from the ground truth (GT). The Gaussian set trained with random initialization shows significant noise in the 3D geometry, making it unsuitable for instance segmentation tasks. In contrast, the Gaussian set sampled from the GT and then trained achieves much higher geometric accuracy with minimal noise, closely aligning with the GT.

A.1.2 Implementation Details of Object Grounding

Instance segmentation. In our experiments, we set the default confidence threshold $c = 0.8$ for instance segmentation to achieve higher segmentation precision. However, to avoid excluding small objects (e.g., paper, cups, books), we lower the threshold to $c = 0.3$ when targeting such challenging objects for the pre-trained model, even if this results in additional noise or slight mis-segmentation.

In Fig. A.2, when the confidence threshold c is set to 0.8, each instance is segmented more completely. However, objects with a confidence below c are filtered out (highlighted by green bounding boxes). Conversely, at $c = 0.3$, more objects are successfully segmented, but additional noise appears in some instances and on the floor (highlighted by red bounding boxes).

A.1.3 Implementation Details of 3D Gaussians Editing

Object recoloring. We designed a color-mapping table (refer to "**color-mapping.pdf**") to translate color keywords from prompts into their corresponding RGB values. This

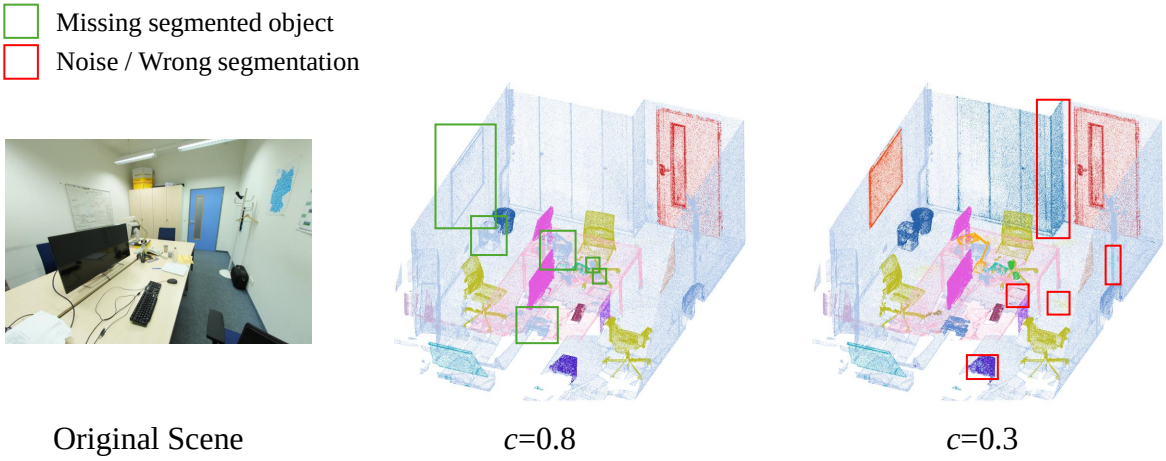


Figure A.2: Visualization of instance segmentation with different confidence threshold. In this figure, we visualize instance segmentation of a 3D scene represented by a Gaussian set using colorful point clouds as Gaussians lack RGB attributes. Green bounding boxes highlight the missed segmented objects when the confidence threshold is set to $c = 0.8$ compared to $c = 0.3$, while red bounding boxes indicate additional noise or mis-segmentations introduced at the lower threshold.

pipeline enables editing with over 200 distinct colors, ensuring precise and flexible color adjustments.

Object addtion and replacement. Our pipeline leverages LGM [251], one of the fastest Gaussian-based generative backbone, for creating new objects. To ensure compatibility with 3D Gaussian Splatting, we set the spherical harmonics degree $sh = 3$, while keeping the remaining parameters unchanged. The pipeline supports inputs in the form of text-only, image-only or text + image combinations for the generative model, with each generation trained for 30 iterations.

Object relighting. To align the illumination of new objects with the scene, we first extract an HDR environment map from a scene image using DiffusionLight [197]. We then use Blender to relight the multi-view images generated by LGM [251], which are subsequently used to generate the final objects. Notably, object relighting is optional in our editing pipeline, as extracting the HDR map typically takes around 15-20 minutes, which would disrupt our interactive-rate editing workflow. The visualization of object relighting is shown in Fig. A.4.

Detail of spatial relation interpreting. Here we provide more detail about our spatial relation interpreting module. We assume the virtual camera is positioned at the geometric center of the scene and infer view-dependent relationships by projecting both the target and reference objects onto a 2D plane along the camera ray direction.



Figure A.3: More visualization result of 3DSceneEditor.

As illustrated in Fig. A.5 (*Prompt: Remove the trash can on the right side of the refrigerator*), the trash can within the red bbox is selected because it lies to the right of the refrigerator (green bbox) relative to the camera ray direction.

Adaptive optimization framework. We provide more visualization of our adaptive optimization module in Fig.A.6. With the help of our optimization framework, we can generate results with higher human aesthetic scores.

User study design. We conducted a user study with 42 professionals working in 3D field to assess the quality of edits, as detailed in Fig. A.7. The study includes 20 questions covering 10 different editing operations. For each operation assessment, it combined with an original video, four edited versions generated by the evaluated methods, and a corresponding target editing instruction. Participants were asked to answer two key questions:

1) **Prompt Alignment** – Evaluating how well the edits match the given prompt.

2) **Editing Quality** – Selecting the video with the best quality based on their first choice.

If none of the edited scenes align with the prompt, participants can select *None of the above* to avoid a forced choice. Note that responses marked as *None of the above* will

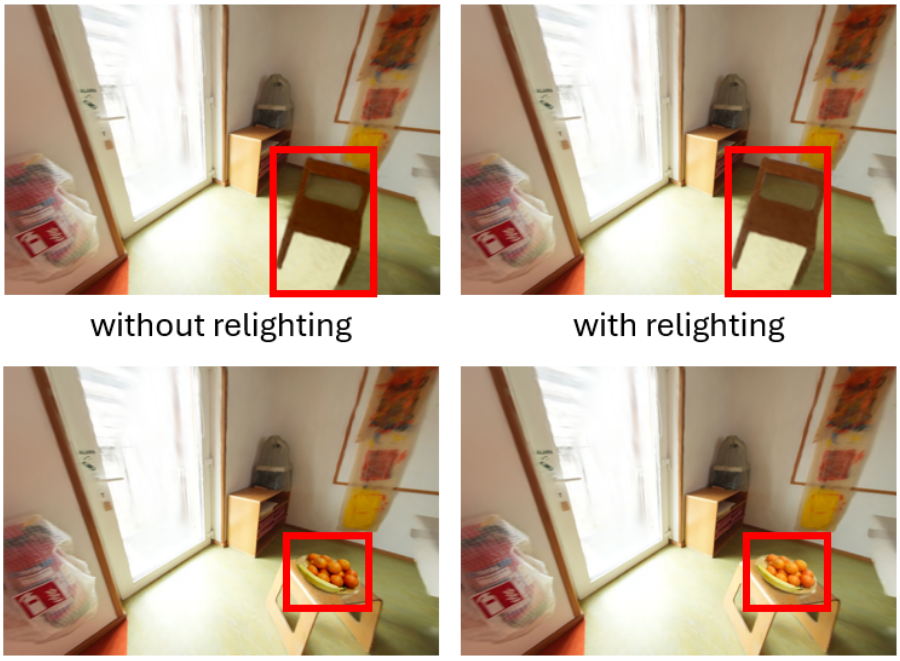


Figure A.4: Results of object relighting

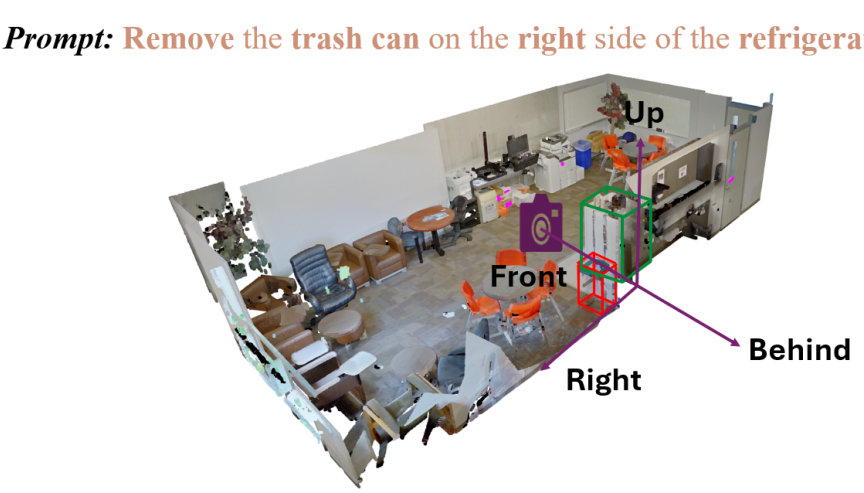


Figure A.5: Visualization of spatial relation.

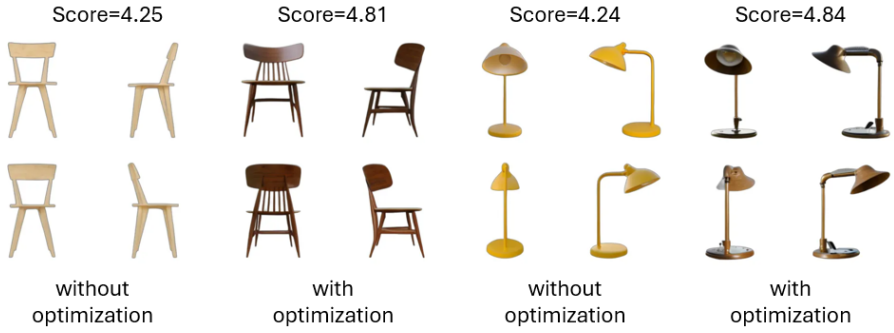


Figure A.6: More visualization of adaptive optimization

be excluded from the Editing Quality evaluation, as this question is skipped in such cases.

3D Editing Quality Assessment

Instructions - MUST READ

A 3D editing pipeline is tasked to edit a scene provided in the first video of each questionnaire, with *a text prompt,* e.g. "Change the color of a pillow on the on the bed to DarkOrchid".

We want to evaluate the quality of the edited scenes. You will be asked to assess it from two perspectives: **prompt alignment** and **editing quality**.
- **Prompt Alignment** refers to whether the content of the results accurately reflects the instructed text prompts.
- **Editing Quality** is determined by various details in the post-editing scenes, including whether all target objects have been successfully removed, whether added or replaced objects are accurately positioned and visually realistic, and whether the video maintains clarity without local artifacts.

On each page, you will first see the initial scene (by video) and the text prompt. Then you will be asked to evaluate four edit results (by video) genertated by four 3D editing methods.
Note: Please refresh if the video is not loaded.

Example of the Questionnaire

Given the initial scene shown in the following video:

and the text prompt:

"Remove the monitors on the desk".



Please watch the following four videos and assess the prompt alignment and editing quality.



Please read carefully:

1. Answer the first question, then choose one option for the second question based on your selection from the first question.

2. If none of the videos align with the prompt, select ***"None of the above"*** in the first question and ***"N/A"*** in the second question.

	1	2	3	4	None of the above *
Which edited results align with the prompt? (could be multiple selection)	☐	☐	☐	☐	☐

*

	1	2	3	4	N/A
Which edited result has the best editing quality	○	○	○	○	○

Get started

Figure A.7: Design of our user study.