Uncertainty-aware 3D Edge Reconstruction with Difference of Gaussians

Caixia Zhou[1], Jiahao Ma[2], Yaping Huang[1] *, Haibin Ling[3], Jing Zhang[2]
[1]Beijing Key Laboratory of Traffic Data Mining and Embodied Intelligence, Beijing Jiaotong University
[2]Australian National University [3]Westlake University
{cxzhou,yphuang}@bjtu.edu.cn, {jiahao.ma, jing.zhang}@anu.edu.au, haibin.ling@gmail.com

Abstract

*3D edge reconstruction from posed multi-view images remains a critical yet under-explored task. While 3D Gaussian Splatting (3DGS)-based methods have recently achieved promising performance, they face two main challenges. First, edges exhibit clear discontinuities from the background, but the intrinsic smoothness of Gaussian kernels makes it challenging to model such discontinuities. Second, due to the absence of multi-view edge annotations, models are trained with pseudo labels instead. These pseudo labels extracted by pre-trained 2D edge detectors often exhibit cross-view inconsistencies, leading to degraded performance. To address these issues, we propose a novel uncertainty-aware 3D edge reconstruction using Difference of Gaussians (DoG) as kernels, called **EdgeDoG**. First, we incorporate DoG kernels to model edge discontinuities explicitly. Second, we design a dual-uncertainty strategy: primitive-level uncertainty is estimated via multi-view Fisher information to eliminate noisy 3D primitives, while pixel-level uncertainty is computed from gradients of rendered depth maps to reweight the training loss, thereby compensating for inconsistent 2D pseudo labels with robust 3D geometric cues. Extensive experiments on diverse datasets demonstrate that our method achieves superior performance compared to previous approaches.*

1. Introduction

3D edge reconstruction [6, 35, 74] aims to identify and extract critical edge features from object surfaces, particularly in regions with significant geometric discontinuities. This process effectively reduces redundant information while preserving essential structural details, enabling more efficient and lightweight reconstruction. Unlike 2D edge detectors [12, 66, 73, 82] trained on single-view images, 3D edge reconstruction exhibits stronger robustness to illumination variations and occlusions [35], benefiting from the complementary information across multiple views. Thanks to the informative and concise representation of geometric details, the reconstructed 3D edges can be seamlessly integrated into various downstream applications, including medical analysis [52] as well as 3D modeling and animation [5, 23, 56].

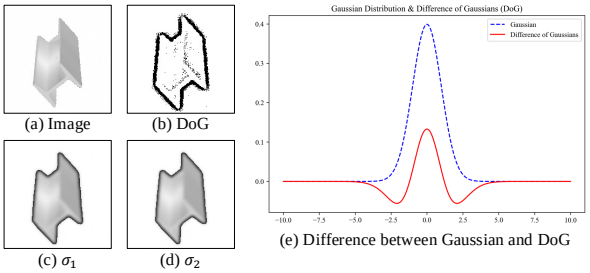

Figure 1. Visualization of Difference of Gaussians (DoG). (a) shows the input image from the ABC-NEF [72] dataset, (b) presents the DoG result, which emphasizes the edges. (c) and (d) show Gaussian-blurred versions of the input image using different σ values. (e) plots the 1D profiles of the Gaussian and DoG curves, demonstrating the sharper response of DoG.

Existing approaches [6, 35, 72] typically follow a two-stage pipeline. They first use pre-trained 2D edge detectors [48, 59] to extract pseudo-edge maps from multi-view RGB images, and then fit a 3D model to the pseudo-edge maps. For instance, NEF [72] constructs a neural edge field from multi-view pseudo-edge maps and extracts parametric curves as final edges. Similarly, EMAP [35] learns an edge field using unsigned distance functions, from which edge distances and directions can be extracted. However, both NEF and EMAP rely on Neural Radiance Fields (NeRF) [45] to build neural representations, which requires extensive training time and makes real-time rendering infeasible. In contrast, 3D Gaussian Splatting (3DGS) [31] has recently gained popularity for its ability to train within minutes and render thousands of images per second. Building upon this success, EdgeGS [6] integrates Gaussian shape and orientation regularizations into the 3DGS training process, achieving impressive performance on the 3D edge reconstruction.

However, 3DGS-based methods still face two key chal-

*Corresponding author.

lenges. First, the inherent smoothness of Gaussian kernels limits their ability to model discontinuities, which are essential for accurate edge reconstruction [17, 50]. Second, in the absence of multi-view edge annotations, pseudo labels are generated independently for each image using 2D edge detectors [48, 59]. This procedure introduces cross-view inconsistencies, thereby producing noisy and less coherent 3D Gaussian representations.

To overcome these challenges, in this paper, we propose a novel uncertainty-aware 3D edge reconstruction with Difference of Gaussians (DoG), called EdgeDoG. The intuition behind our work is twofold. First, we propose replacing the standard Gaussian kernels in 3DGS with DoG kernels, a formulation better suited for capturing edge discontinuities. Originally introduced for image-based edge detection [44], DoG enhances edge detection by subtracting two Gaussian-blurred versions of the input image with different standard deviations. Recent works [30, 62] have further demonstrated DoG's effectiveness in modeling geometric discontinuities in 3D scenes. As illustrated in Fig. 1, for an input image (a), (c) and (d) show two blurred variants, while (b) shows the resulting DoG response, which reveals clearer edges than (c) and (d). Fig. 1(e) additionally illustrates that the DoG curve responds more sharply to edges than the Gaussian curve, demonstrating its inherent superior sensitivity to capture discontinuities. Thus, we replace the original Gaussian kernels in 3DGS with a DoG-based formulation, enhancing the ability to model abrupt geometric transitions at edges.

Second, we propose a novel dual-uncertainty strategy to mitigate the negative effects caused by cross-view pseudo-label inconsistencies. Specifically, we estimate the uncertainties at both the pixel and primitive levels to explicitly regularize training and improve reconstruction robustness. At the pixel level, inconsistencies in 2D pseudo-edge labels often arise from the lack of 3D spatial context. Therefore, we compute the gradient magnitude of the rendered depth map and use it to reweight the loss, leveraging this geometric cue to compensate for the inconsistencies in 2D pseudo-labels and reduce their negative impact during training. Meanwhile, primitive-level uncertainty is quantified based on the multi-view Fisher information [32]. We maintain the most confident primitives, which significantly reduces noisy primitives caused by pseudo-label inconsistencies. Combining these two uncertainty techniques, our approach effectively reduces ambiguity and enhances the coherence of 3D edge reconstruction. In summary, our contributions are as follows:

- We propose to integrate the Difference of Gaussians into each Gaussian primitive to explicitly model discontinuities commonly observed in edge regions, thereby enabling more accurate learning of 3D edge points.
- We propose a dual-uncertainty strategy that estimates uncertainties at both the primitive and pixel levels, reducing the impact of ambiguous multi-view pseudo labels and enabling more robust training.
- We conduct extensive experiments across four diverse datasets. The results demonstrate the effectiveness and generalizability of our proposed approach.

2. Related Work

3D edge reconstruction is closely related to surface reconstruction, particularly through 3DGS-based methods. As 3DGS provides a powerful and efficient framework for surface reconstruction, understanding its principles and recent progress is crucial for advancing edge reconstruction. In this section, we first review the core concepts and applications of 3DGS in general surface reconstruction, followed by recent developments in 3D edge reconstruction.

Surface Reconstruction with 3DGS. 3DGS [8, 31] is an emerging technique for 3D scene representation using Gaussian primitives. While computationally efficient, it struggles with high-quality geometry reconstruction due to unstructured data and normal estimation ambiguities [11]. To address this, recent works propose integrating implicit signed distance functions for joint surface optimization (*e.g.*, NeuSG [9], GSDF [76], GSurf [3], GS-Pull [79], 3DGSR [43]) and applying scale regularization for better surface alignment (*e.g.*, SuGaR [16], Gaussian Surfels [11], ERankGS [27]). 2DGS [24] improves multi-view consistency by projecting Gaussians to 2D. Other methods such as RaDe-GS [78], GS2Mesh [64], and PGSR [7] enhance depth rasterization, while depth and normal constraints are adopted in works like Geo-3DGS [67], VCR-GauS [10], and 2DGS. Pre-trained geometric priors are utilized by GeoGaussian [36], GSRec [65], and GausSurf [61]. Additionally, GOF [77] and TrimGS [14] explicitly refine rendering via ray-Gaussian intersection and alpha-blending computations.

3D Edge Reconstruction involves creating coherent 3D edges from 2D multi-view posed pseudo edges. The first effort, NEF [72], introduces a neural edge field that leverages volumetric rendering of 2D edge maps, optimized through sparsity, balancing, and 3D view consistency constraints to reconstruct visible and occluded edges. LIMAP [39] is a line-based 3D mapping system that integrates line-point associations, multi-view line triangulation with geometric priors, and joint optimization to robustly reconstruct structured 3D line maps. NEAT [68] proposes a rendering-distillation pipeline employing neural fields to reconstruct structured 3D line segments directly from 2D wireframe images, optimizing both neural representations and global junction perception without requiring explicit cross-view correspondences. Similarly, EMAP [35] develops a neural implicit unsigned distance function representation for 3D edges, addressing class imbalance and occlusions with an importance-sampling strategy. However, these implicit neural methods require significant computational resource and training time. Addressing this efficiency challenge, EdgeGS [6] leverages

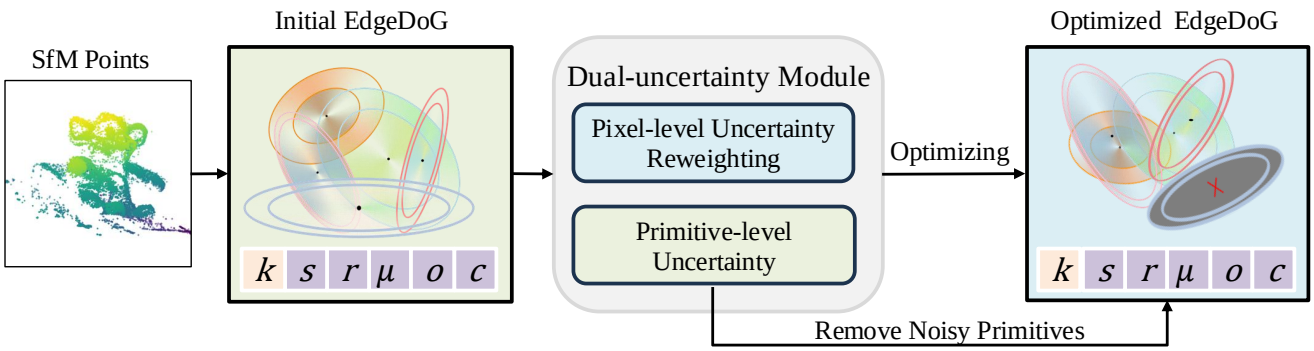

Figure 2. The overview of our proposed EdgeDoG. Given posed RGB images, we first apply Structure-from-Motion (SfM) [53] to generate initial point clouds and obtain pseudo-edge maps via a pre-trained edge detector. The point clouds initialize the positions and scales of EdgeDoG, while other attributes (*e.g.*, color, opacity) are randomly set. We then optimize EdgeDoG using an uncertainty-aware pixel-level loss, where pixel-wise uncertainty is derived from the gradient magnitude of rendered depth and used to reweight the loss for robustness against inconsistent pseudo-edge maps. Additionally, we model primitive-level uncertainty using multi-view Fisher information to identify and prune unreliable primitives (red crosses). Training is supervised by pseudo-edge maps.

the fast rendering abilities of 3DGS by regarding the centers of Gaussians as edge points and their principal axes as edge directions, optimizing these representations via scale and directional constraints. LineGS [69] further improves upon Gaussian-based initialization by mitigating common issues such as positional bias, overextension, outliers, duplication, and discontinuities, achieving geometry-aware line segment reconstruction. Additionally, PlanarSplatting [60] reconstructs planar regions in indoor environments by representing each plane as a learnable 3D rectangular primitive with optimizable center position, rotation, and size. 3D edge reconstruction is closely related to 3D line and wireframe reconstruction [39, 68], which focus on structured boundaries like line segments and junctions. While wireframe methods from point clouds [20, 25, 26, 38, 40, 42, 75] offer strong generalization, they rely on specialized datasets and are beyond our scope.

Distinct from prior work, we adopt Difference of Gaussians (DoG) instead of standard kernels to better capture edge discontinuities. Additionally, we introduce uncertainty modeling at both pixel and primitive levels to resolve inconsistencies in multi-view pseudo-edge maps, enabling sharper and more robust 3D edge reconstruction.

3. The Proposed EdgeDoG

3.1. Revisit 3D Gaussian Splatting

Gaussian Splatting (GS) [31] represents scenes using a set of 3D Gaussians $\mathcal{G}$, each defined by a center $\mu \in \mathbb{R}^3$, covariance $\Sigma \in \mathbb{R}^{3\times3}$, color $c \in \mathbb{R}^3$, and opacity $o \in \mathbb{R}^1$. To reduce memory and stabilize training, Σ is decomposed as $\Sigma = RSS^\top R^\top$, with S being the diagonal of the scaling matrix s and R derived from a quaternion r. Rendering involves transforming μ and Σ to the camera frame via view matrix W, projecting $\mu_{\text{cam}} = W\mu$ to 2D as μ', and computing the 2D covariance $\Sigma' = JW\Sigma W^\top$, where J is the Jacobian of the projection. Each Gaussian's contribution α_i to a pixel x is based on its opacity and projected 2D Gaussian:

$$\alpha_i = o_i g_i(x;\mu',\Sigma'), \quad g_i(x;\mu',\Sigma') = e^{-\frac{1}{2}(x-\mu')^\top \Sigma'^{-1}(x-\mu')}. \tag{1}$$

Pixel color C and depth d are rendered by alpha-blending:

$$C = \sum_i c_i \alpha_i \prod_{j=1}^{i-1}(1-\alpha_j), \quad d = \sum_i z_i \alpha_i \prod_{j=1}^{i-1}(1-\alpha_j), \tag{2}$$

where Gaussians are sorted by camera depth z_i, and the product term represents transmittance. Rendering is supervised by 2D ground truth using a loss combining $\mathcal{L}_1$ (pixel-wise error) and D-SSIM [63] (perceptual similarity):

$$\mathcal{L} = (1-\lambda)\mathcal{L}_1 + \lambda\mathcal{L}_{\text{D-SSIM}}. \tag{3}$$

To improve reconstruction, 3DGS dynamically updates the number of Gaussians via densification (splitting, cloning, pruning). Following EdgeGS [6], we adopt its adaptive density control without further elaboration. We initialize Gaussians using point clouds from SfM [53]. Only positions and scales are derived from these points, while other attributes are randomly set.

3.2. 3DGS with DoG

In the context of 3DGS, traditional Gaussian kernels are challenging to model sharp discontinuities, making them suboptimal for accurately reconstructing 3D edges. To explicitly handle such discontinuities, we integrate Difference of Gaussians (DoG) kernels into the 3DGS framework.

DoG is a classical and effective edge detection operator, known for capturing regions with abrupt intensity changes by highlighting high-frequency details and suppressing noise.

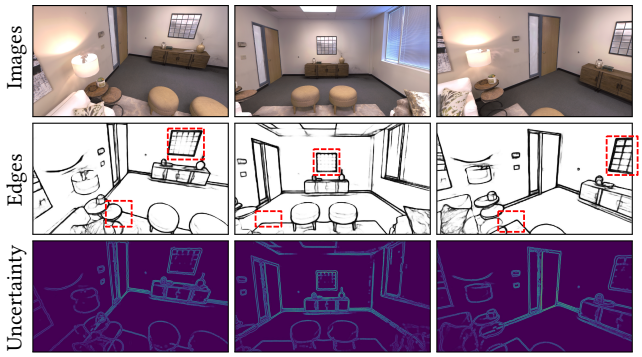

Figure 3. Inconsistent pseudo edges extracted by the PidiNet [59] edge detector on the Replica [58] Room 0 dataset. The inconsistency is highlighted in red bounding boxes. The last row shows the generated pixel-wise uncertainty maps.

As shown in Fig. 1, it operates by subtracting two Gaussian-blurred versions of the same input image, where one is blurred with a larger standard deviation (σ_1), and the other is with a smaller one (σ_2). For an input image $I(x)$, the DoG operation is:

$$\text{DoG}(x) = G_{\sigma_1}(x) * I(x) - G_{\sigma_2}(x) * I(x), \quad (4)$$

where $G_\sigma(x)$ is a Gaussian kernel with standard deviation σ, and $*$ denotes convolution. This process provides an efficient approximation of the Laplacian of Gaussian (LoG), significantly reducing computational complexity without compromising performance.

To integrate DoG into 3DGS, we extend each 3D Gaussian primitive by introducing an additional shape parameter $k \in \mathbb{R}^2$ (Fig. 2), where each element independently modulates the scale of one Gaussian kernel. More precisely, the original 2D projected Gaussian density in Eq. 1 is replaced by the difference between two Gaussian densities with distinct scales. The first element of k scales the covariance matrix of the first Gaussian, and the second element scales the covariance of the second one. The resulting DoG formulation is:

$$g_{DoG_i}(x) = g_i(x; \mu', k_1\Sigma') - g_i(x; \mu', k_2\Sigma'), \quad (5)$$

where $k = [k_1, k_2]^\top \in \mathbb{R}^2$.

By incorporating DoG, our method effectively emphasizes high-frequency features corresponding to edges and boundaries, resulting in sharper and more accurate 3D edge reconstructions. This approach addresses the inherent smoothing limitation of standard Gaussian, which typically produces overly blurred edges, thus significantly improving reconstruction quality, particularly around object contours (Fig. 1 (b) *vs*. Fig. 1 (c)).

3.3. Dual-Uncertainty Modeling

For the 3D edge reconstruction task, pseudo labels are generated by running 2D edge detectors on multi-view images. The variations caused by illumination changes, viewpoint differences, and the inherent sensitivity of edge detectors, result in inconsistent edge detection predictions across different views as illustrated in Fig. 3, and subsequently lead to noisy and less coherent 3D representations. To effectively handle these inconsistencies, we introduce a dual-uncertainty modeling at both the primitive level and pixel level as shown in Fig. 2, aiming to quantify inconsistency and reduce its impact in our reconstruction process.

Pixel-level Uncertainty Estimation. Edges often correspond to significant discontinuities between objects and backgrounds, which often manifest as sharp changes in depth values. The depth map, which cannot be perceived in 2D image space, can provide complementary geometric cues to mitigate inconsistency learning from the 2D pseudo-edge labels. Therefore, we exploit the rendered depth maps from 3D reconstruction to estimate pixel-wise uncertainty and guide more effective learning.

Specifically, we apply the Sobel operator to the rendered depth map d to compute gradient magnitudes along both horizontal and vertical directions, which highlight regions with large depth discontinuities. By applying the Sobel operator to a depth map, we can identify challenging regions with large depth gradients and make the model focus on the reconstruction of these regions. The uncertainty at each pixel (x, y) is then computed as:

$$U_p(x,y) = \text{Norm}\left(\sqrt{\left(\frac{\partial d}{\partial x}\right)^2 + \left(\frac{\partial d}{\partial y}\right)^2}\right), \quad (6)$$

where $\text{Norm}(\cdot)$ denotes min-max normalization to $[0, 1]$.

The resulting uncertainty map, denoted as U_p, serves as a weighting factor for the pixel-wise loss $\mathcal{L}_1$. Furthermore, following the EdgeGS approach [6], we exclude the SSIM loss and instead incorporate direction and scale losses. The final objective function is thus defined as:

$$\mathcal{L} = \exp(U_p)\mathcal{L}_1 + \mathcal{L}_\text{v} + \mathcal{L}_\text{s},$$
$$\mathcal{L}_\text{v} = 1 - \frac{1}{N}\left(\sum_{i=1}^{N}\frac{1}{k}\sum_{j=1}^{q}\left|n_i^\top n_{ij}\right|\right), \mathcal{L}_\text{s} = \frac{1}{N}\sum_{i=1}^{N}\frac{s_i^{(2)}}{s_i^{(1)}}. \quad (7)$$

where $\mathcal{L}_\text{v}$ and $\mathcal{L}_\text{s}$ represent the direction and scale losses adopted from EdgeGS [6]. n_i is the direction of the i-th edge point, and q is a hyperparameter indicating the number of neighboring edge points considered. $s_i^{(1)}$ and $s_i^{(2)}$ denote the largest and second-largest scales within each primitive. As illustrated in Fig. 3, the measured pixel-wise uncertainty mainly corresponds to edge-related regions. Therefore, our uncertainty-weighted loss function increases the contribution of those regions during training. This encourages the model to focus on learning informative structures, leading to more robust edge reconstruction results.

Table 1. The quantitative results on the ABC-NEF [72] dataset. We report recall (R), precision (P), F-score (F), accuracy (Acc), and completeness (Com). R, P and F are calculated using different distance thresholds (*e.g.*, R5 denotes recall with a 5mm threshold).

Method	Model	Detector	Modal	Acc↓	Comp↓	R5↑	R10↑	R20↑	P5↑	P10↑	P20↑	F5↑	F10↑	F20↑
Matching-based	LIMAP [39]	LSD [21]	Line	9.9	18.7	36.2	82.3	87.9	43.0	87.6	93.9	39.0	84.3	90.4
		SOLD2 [47]	Edge	**5.9**	29.6	**64.2**	76.6	79.6	**88.1**	**96.4**	**97.9**	**72.9**	84.0	86.7
NeRF-based	NEF [72]	PidiNet [59]	Curve	15.1	16.5	11.7	53.3	93.9	12.3	61.3	95.8	12.3	51.8	88.7
		DexiNed [48]	Curve	21.9	15.7	11.3	48.3	93.7	11.5	58.9	91.7	10.8	42.1	76.8
	EMAP [35]	PidiNet [59]	Edge	9.2	15.6	30.2	75.7	89.5	35.6	79.1	95.4	32.4	77.0	92.2
		DexiNed [48]	Edge	8.8	8.9	56.4	88.9	94.8	62.9	89.9	95.7	59.1	88.9	94.9
Splatting-based	EdgeGS [6]	PidiNet [59]	Edge	11.7	10.3	17.1	73.9	83.1	26.0	87.2	92.5	20.6	79.3	86.7
		DexiNed [48]	Edge	9.6	8.4	42.4	**91.7**	95.8	49.1	94.8	96.3	45.2	**93.7**	95.7
	Ours	PidiNet [59]	Edge	8.7	10.7	31.1	87.9	94.7	36.1	90.2	96.7	33.2	88.7	95.4
		DexiNed [48]	Edge	7.4	**8.2**	44.5	91.6	**96.5**	53.6	95.6	97.8	48.4	93.2	**97.0**

Primitive-level Uncertainty Estimation. For the 3D edge reconstruction task, inconsistent 2D edge labels also inevitably lead to the generated primitives containing noise. To remove the noisy and less informative primitives, we compute an importance score for each primitive during training. Recognizing that training is fundamentally a process of accumulating reliable information, we leverage Fisher information as our uncertainty metric for evaluating primitives' importance, which quantifies how much information the observed data provides about model parameters [29].

Fisher information can be interpreted through the Hessian of the negative log-likelihood with respect to model parameters [32]. Directly computing this Hessian matrix is computationally infeasible due to the complexity and large number of parameters. To circumvent this issue, we adopt a Fisher approximation approach following previous work [19], which simplifies the Hessian calculation. Specifically, this approximation omits second-order terms by assuming that the rendered image $I_{\mathcal{G}}$ closely matches the ground truth, thus yielding the approximated Hessian as follows:

$$H = \nabla^2_{\mathcal{G}} \mathcal{L}_2 = \sum_{\phi \in \mathcal{P}_{gt}} \nabla_{\mathcal{G}} I_{\mathcal{G}}(\phi) \nabla_{\mathcal{G}} I_{\mathcal{G}}(\phi)^T, \tag{8}$$

where $\mathcal{P}_{gt}$ is all input poses set and ϕ denotes an individual pose. $\nabla_{\mathcal{G}}$ denotes the gradient over all parameters on $I_{\mathcal{G}}$.

Directly calculating gradients for each primitive's complete set of parameters is still highly resource-intensive. Instead, we exploit the stored gradient information of g_i in Eq. 5 from the backward pass during the rendering process [18], which contains geometric attributes (mean μ, rotation r, scale s, and shape parameter k). Consequently, we define the uncertainty for the i-th primitive as:

$$U_i = (\nabla_{g_i} I_{\mathcal{G}})^2. \tag{9}$$

By calculating uncertainty scores for all primitives with multi-view pseudo edge labels periodically, we sort these primitives according to their importance. We then retain the top p proportion with the lowest uncertainty (highest confidence) and remove the less reliable ones. This strategy effectively filters out noisy or inconsistent primitives, significantly enhancing the robustness and accuracy of the final 3D edge reconstruction results.

4. Experiments

4.1. Experiment Settings

Datasets. Following EMAP [35], we conduct quantitative evaluations on ABC-NEF [72] and DTU [1], and qualitative analyses on Replica [58] and Tanks & Temples (TNT) [33] due to the lack of ground truth. **ABC-NEF** [72] contains 115 challenging CAD models from the ABC dataset [34], with 82 models retained by EMAP [35] after filtering for edge annotation consistency. Each model provides 50 rendered views and ground-truth parametric edges. **DTU** [1] consists of 80 indoor scenes captured with varying camera configurations. EMAP selects 6 diverse scenes and generates pseudo-3D edge points by projecting ground-truth point clouds and filtering them using multi-view consistent 2D edge maps [4] with scene-specific thresholds. **Replica** [58] offers 18 photorealistic indoor scenes; we use three room-scale scenes with 100 rendered views each for qualitative evaluation. **TNT** [33] provides real-world outdoor scenes with HDR-scanned ground truth. EMAP selects four representative scenes (barn, caterpillar, meeting room, and truck) for visual comparison.

Evaluation Metrics. Following prior works [6, 35, 72], we assess reconstruction quality using Precision, Recall, and F-score at multiple distance thresholds in millimeters (mm). For fair comparison with EMAP [35] and EdgeGS [6], we convert predicted edges to parametric representations and uniformly sample points along both predictions and (pseudo) ground truth. We also report Accuracy (Acc) and Completeness (Comp) in mm. Accuracy represents the average distance from predicted edge points to their nearest ground-truth

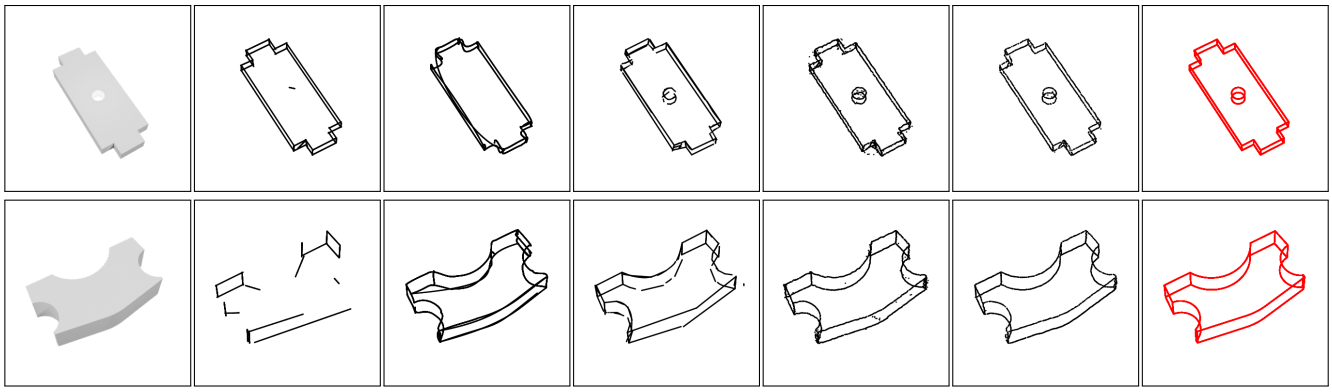

Figure 4. Visual comparisons with existing methods on the ABC-NEF [72] dataset.

points, while completeness measures the average distance from ground-truth points to their nearest predicted points.

Implementation Details. We build on EdgeGS [6] as our baseline, adopting its main experimental settings, including learning rate, training epochs, direction and scale loss weights, neighboring edge points q, and densification strategy. As in EdgeGS, we fix color parameters to one. All introduced shape parameters k are initialized to 0 with a learning rate of 1×10^{-4}; we use a Sigmoid activation for ABC-NEF and an exponential function for other datasets. Primitive-level uncertainty masks are computed using Hessian approximation every 50 epochs. To balance precision and recall, we empirically set the Gaussian selection ratio p to 0.9. All experiments are conducted using PyTorch [46] on a single RTX 4090.

4.2. Comparison with State-of-the-Art

We compare our proposed EdgeDoG with the exsiting 3D edge reconstruction methods, including matching-based method LIMAP [39], NeRF-based methods, including NEF [72], NEAT [68], and EMAP [35], as well as 3DGS-based baseline, EdgeGS [6].

Results on the ABC-NEF Dataset. Table 1 and Fig. 4 provide quantitative and visual comparisons between our proposed EdgeDoG and existing approaches, respectively. Compared with EdgeGS [6], EdgeDoG achieves almost the state-of-the-art results, particularly in terms of accuracy and precision. Visually, EdgeGS [6] exhibit discontinuities or noise, while EdgeDoG generates smoother and more continuous reconstructions. This improvement comes from explicitly modeling edge discontinuities using the Difference of Gaussians, which helps capture subtle curved features that other methods tend to miss, especially in low-contrast or partially occluded regions. In addition, EdgeDoG incorporates uncertainty modeling, which reduces artifacts and ensures more consistent results across different views.

EdgeDoG slightly underperforms EMAP under the strict 5 mm threshold, this is mainly due to the bias introduced by thick 2D pseudo edges, which allow valid 3D primitives to shift slightly away from the actual 3D edge positions. This deviation impacts performance under small thresholds despite the visual quality remaining high. LIMAP [39], while performing well under small distance thresholds by leveraging strong structural priors, sacrifices completeness by focusing mainly on prominent straight-line structures, leading to unsatisfying visualized quality. Similarly, NEF [72] often misses finer geometric details. In contrast, our proposed EdgeDoG consistently produces more precise and complete 3D edge reconstructions than these baselines, effectively balancing accuracy, completeness, and visual quality.

Results on the DTU dataset. Table 2 reports the quantitative results on the DTU dataset. Our proposed EdgeDoG consistently achieves the best or nearly the best performance across all scans. Specifically, EdgeDoG improves the mean recall from 84.6 to 86.7, the mean precision from 75.4 to 82.1, and the mean F-score from 78.6 to 84.0. Although NEAT [68] obtains the highest precision on scan110, this comes with significantly lower recall, leading to an overall lower F-score than ours. This demonstrates that EdgeDoG effectively balances precision and recall, delivering robust and accurate 3D edge reconstructions.

Results on the Replica and TNT dataset. Due to the lack of labels, we only show the visualized comparisons of 3D edge reconstructions for the Replica and TNT datasets as shown in Fig. 5 and Fig 6. The visualizations are obtained by sampling the parameterized curves at a resolution of 0.0001 and then projecting the sampled points onto the 2D image plane. EMAP suffers from cluttered and overly thick lines. Compared with EdgeGS, our proposed EdgeDoG can produce more semantically meaningful edges.

4.3. Ablation Study

Our proposed method includes three key components: the DoG kernel, primitive-level uncertainty, and pixel-level uncertainty. To analyze their individual contributions, we conduct an ablation study on scan118 of the DTU dataset. The

Table 2. The recall (R), precision (P), and F-score (F) results of 3D edge reconstruction on the DTU [1] dataset.

Scan	LIMAP [39]			NEF [72]			NEAT [68]			EMAP [35]			EdgeGS [6]			Ours		
	R5↑	P5↑	F5↑	R5↑	P5↑	F5↑	R5↑	P5↑	F5↑	R5↑	P5↑	F5↑	R5↑	P5↑	F5↑	R5↑	P5↑	F5↑
37	75.8	74.3	75.0	39.5	51.0	44.5	63.9	85.1	73.1	62.7	83.9	71.7	83.0	80.3	81.6	**86.0**	**89.6**	**87.8**
83	75.7	50.7	60.7	32.0	21.8	25.9	72.3	52.4	60.7	72.3	61.5	66.5	81.6	65.8	72.9	**83.4**	**70.4**	**76.3**
105	79.1	64.9	71.2	30.3	32.0	31.1	68.9	73.3	71.0	78.5	78.0	78.2	78.3	75.9	77.1	**79.2**	**86.7**	**82.8**
110	79.7	65.3	71.7	31.2	40.2	35.0	64.3	**79.6**	71.2	**90.9**	68.3	77.9	87.3	60.0	71.1	90.0	67.4	**77.0**
118	59.4	62.0	60.7	15.3	25.2	19.0	59.0	71.1	64.5	75.3	78.1	76.7	84.9	76.6	80.5	**87.6**	**84.7**	**86.1**
122	79.9	79.2	79.5	15.1	29.1	19.9	70.0	82.0	75.5	85.3	82.9	84.1	92.3	84.8	88.4	**94.0**	**94.0**	**94.0**
Mean	74.9	66.1	69.9	27.2	33.2	29.8	66.4	73.9	70.0	77.5	75.4	76.4	84.6	73.5	78.6	**86.7**	**82.1**	**84.0**

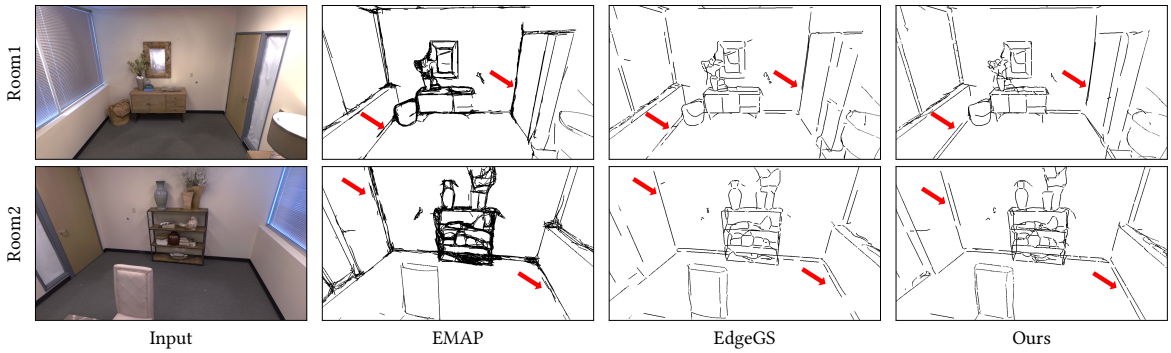

Figure 5. Qualitative comparisons on the Replica [58] dataset. As illustrated, our method produces cleaner lines than EMAP. And compared to EdgeGS, our lines are more continuous, especially around the borders of doors and windows.

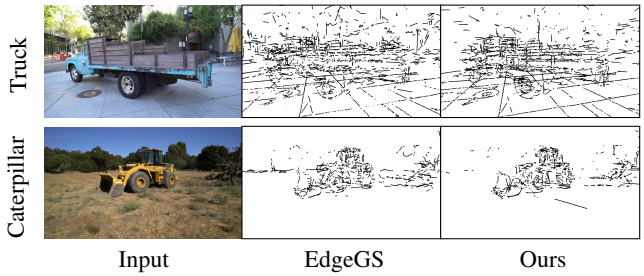

Figure 6. Qualitative comparisons on the TNT [33] dataset.

results are shown in Table 3.

Table 3. The ablation study on the DTU scan118 [1] dataset.

DoG	Primitive-level	Pixel-level	R5↑	P5↑	F5↑
			86.8	76.6	81.4
✓			86.7	80.9	83.7
	✓		87.3	78.8	82.9
		✓	88.5	79.1	83.5
✓	✓		86.7	84.1	85.4
✓		✓	88.6	80.5	84.4
	✓	✓	87.6	80.0	83.6
✓	✓	✓	87.6	84.7	86.1

Effectiveness of DoG for Modeling Discontinuities . The DoG kernel explicitly models geometric discontinuities along object boundaries, which is critical for edge reconstruction. As demonstrated in Table 3, incorporating the DoG kernel into the baseline significantly improves the F-score from 81.4 to 83.7. This indicates that explicitly modeling edge discontinuities effectively enhances the detection of fine structural details, improving both precision and recall metrics. Combining with either primitive-level uncertainty or pixel-level uncertainty can further boost the performance.

Effectiveness of Dual-uncertainty for Learning from Inconsistent Labels. Primitive-level uncertainty helps address inconsistencies in 3D primitive learning caused by noisy or ambiguous labels across different views. By assigning a confidence score computed from multi-view Fisher information, it removes noisy primitives. As shown in Table 3, combining primitive-level uncertainty with either the DoG kernel or pixel-level uncertainty consistently leads to better performance than using each component alone. This demonstrates its effectiveness in improving edge reconstruction by enhancing consistency and handling uncertainty.

Pixel-level uncertainty targets inconsistencies arising from multi-view 2D pseudo-edge maps, where occlusions, noise, or viewpoint differences cause labeling variations. We can filter out uncertain or conflicting signals by assigning confidence weights to each pixel-level pseudo edge label. According to Table 3, introducing pixel-level uncertainty sig-

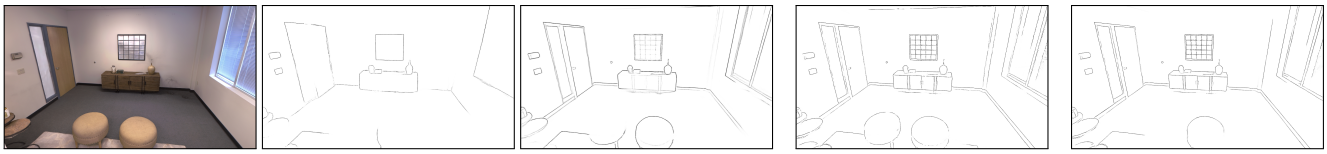

Figure 7. The rendering results for the multi-granularity edge reconstruction. Our method can reconstruct 3D edges at different granularity levels, supporting various task requirements and resource limitations. Particularly, compared with progressive learning (Level 0 → Level 1 → Level 2), directly learning from a complex scene (Level 2) may miss some important details.

Table 4. F-score obtained from the DTU scan118 [1]

Weighted Function	1	$\exp(-U_p)$	$1-U_p$	U_p	$\exp(U_p)$
F-score	85.4	84.8	83.9	83.5	86.1

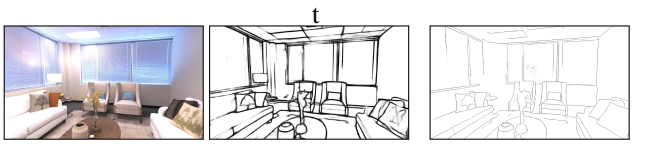

Figure 8. Rendering results supervised by different pseudo edges on the Replica [58] Room 0 dataset.

nificantly boosts both recall and precision, indicating more accurate edge detection. This result underscores its importance in generating cleaner 3D edge reconstructions.

Besides, our pixel-level uncertainty uses $\exp(U_p)$ to emphasize the training from the edge-related regions. To verify the effectiveness of this strategy, we compare the results of different weighted loss functions shown in Table 4. The result proves the effectiveness of our design.

4.4. More Precise Edge Detector

Clearly, the quality of pseudo edge labels significantly influences the edge reconstruction results. Although PidiNet [59] and DexiNed [48] achieve strong performance, many advanced edge detectors [15, 22, 28, 37, 49, 70, 71, 73, 80] have emerged recently in the field of edge detection. We choose generative model-based DiffusionEdge [73] to explore its potential benefits for 3D edge reconstruction. Fig. 8 shows the reconstruction results obtained from both PidiNet and DiffusionEdge models, where DiffusionEdge leads to more complete and structurally consistent edge reconstructions compared to PidiNet because of better pseudo labels.

4.5. Multiple Granularity Edge Reconstruction

Recently, MuGE [81] introduced multi-granularity edge detection to adaptively support diverse downstream tasks. Several works, including GED [82], EDMB [37], and SAUGE [41], have adopted this multi-granularity setting. Similarly, multi-granularity edge information is also crucial in 3D edge reconstruction. Motivated by recent advances in level-of-detail representations [13, 51, 54, 55], we propose leveraging multi-granularity 2D pseudo-edge maps to facilitate effective multi-granularity 3D edge reconstruction.

More specifically, multi-granularity 2D pseudo-edge maps are constructed by DiffusionEdge trained with different datasets, including instance-level NYUD [57], natural image-level BSDS [2], and detailed building-level BIPED [48]. Then we conduct multi-granularity 3D edge reconstruction based on LapisGS [55], which progressively constructs a layered model, starting from the low-resolution base layer (L0) and adding higher-resolution enhancement layers (L1, L2). Here, we regard coarse instance-level edges as the base layer and the detailed edges as the enhancement layer. Fig. 7 shows the multi-granularity reconstruction results on the Replica [58] Room 0 dataset. We can see that our method can not only reconstruct more visually plausible edges, but also produce selective rendering according to the task requirement and resource limitations. It is worth noting that, compared with progressive learning (L0 → L1 → L2), directly learning from complex scene information (L2) may result in the loss of important details.

5. Conclusion and Discussion

In this paper, we propose EdgeDoG, a novel approach tailored explicitly for 3D edge reconstruction, effectively capturing inherent edge characteristics. To address the discontinuities commonly found at edges, we integrated the Difference-of-Gaussians (DoG) into the definition of each Gaussian primitive within the 3DGS framework. Additionally, we tackle the inconsistencies arising from multi-view pseudo-edge maps by introducing a robust dual-uncertainty strategy, encompassing both primitive-level and pixel-level uncertainties. Our comprehensive experiments across diverse and challenging datasets demonstrate that EdgeDoG achieves significant improvements, producing more precise, consistent, and visually coherent 3D edges. These results highlight the potential of explicitly modeling edge discontinuities and uncertainty, opening promising avenues for future research in high-quality 3D modeling.

Acknowledgements. This work is supported by National Natural Science Foundation of China (62271042) and Beijing Natural Science Foundation (L259055).

References

[1] Henrik Aanæs, Rasmus Ramsbøl Jensen, George Vogiatzis, Engin Tola, and Anders Bjorholm Dahl. Large-scale data for multiple-view stereopsis. *International Journal of Computer Vision*, 120:153–168, 2016. 5, 7, 8

[2] Pablo Arbelaez, Michael Maire, Charless Fowlkes, and Jitendra Malik. Contour detection and hierarchical image segmentation. 33(5):898–916, 2010. 8

[3] Xu Baixin, Hu Jiangbei, Li Jiaze, and He Ying. Gsurf: 3d reconstruction via signed distance fields with direct gaussian supervision. *arXiv preprint arXiv:2411.15723*, 2024. 2

[4] Andrea Bignoli, Andrea Romanoni, Matteo Matteucci, and Politecnico di Milano. Multi-view stereo 3d edge reconstruction. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 867–875. IEEE, 2018. 5

[5] Yang Cao, Yuanliang Jv, and Dan Xu. 3dgs-det: Empower 3d gaussian splatting with boundary guidance and box-focused sampling for 3d object detection. *arXiv preprint arXiv:2410.01647*, 2024. 1

[6] Kunal Chelani, Assia Benbihi, Torsten Sattler, and Fredrik Kahl. Edgegaussians–3d edge mapping via gaussian splatting. *arXiv preprint arXiv:2409.12886*, 2024. 1, 2, 3, 4, 5, 6, 7

[7] Danpeng Chen, Hai Li, Weicai Ye, Yifan Wang, Weijian Xie, Shangjin Zhai, Nan Wang, Haomin Liu, Hujun Bao, and Guofeng Zhang. Pgsr: Planar-based gaussian splatting for efficient and high-fidelity surface reconstruction. *arXiv preprint arXiv:2406.06521*, 2024. 2

[8] Guikun Chen and Wenguan Wang. A survey on 3d gaussian splatting. *arXiv preprint arXiv:2401.03890*, 2024. 2

[9] Hanlin Chen, Chen Li, and Gim Hee Lee. Neusg: Neural implicit surface reconstruction with 3d gaussian splatting guidance. *arXiv preprint arXiv:2312.00846*, 2023. 2

[10] Hanlin Chen, Fangyin Wei, Chen Li, Tianxin Huang, Yunsong Wang, and Gim Hee Lee. Vcr-gaus: View consistent depth-normal regularizer for gaussian surface reconstruction. *Advances in Neural Information Processing Systems*, 37:139725–139750, 2025. 2

[11] Pinxuan Dai, Jiamin Xu, Wenxiang Xie, Xinguo Liu, Huamin Wang, and Weiwei Xu. High-quality surface reconstruction using gaussian surfels. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–11, 2024. 2

[12] Ruoxi Deng and Shengjun Liu. Deep structural contour detection. pages 304–312, 2020. 1

[13] Francesco Di Sario, Riccardo Renzulli, Marco Grangetto, Akihiro Sugimoto, and Enzo Tartaglione. Gode: Gaussians on demand for progressive level of detail and scalable compression. *arXiv preprint arXiv:2501.13558*, 2025. 8

[14] Lue Fan, Yuxue Yang, Minxing Li, Hongsheng Li, and Zhaoxiang Zhang. Trim 3d gaussian splatting for accurate geometry representation. *arXiv preprint arXiv:2406.07499*, 2024. 2

[15] Yuanbin Fu and Xiaojie Guo. Practical edge detection via robust collaborative learning. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 2526–2534, 2023. 8

[16] Antoine Guédon and Vincent Lepetit. Sugar: Surface-aligned gaussian splatting for efficient 3d mesh reconstruction and high-quality mesh rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5354–5363, 2024. 2

[17] Abdullah Hamdi, Luke Melas-Kyriazi, Jinjie Mai, Guocheng Qian, Ruoshi Liu, Carl Vondrick, Bernard Ghanem, and Andrea Vedaldi. Ges: Generalized exponential splatting for efficient radiance field rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19812–19822, 2024. 2

[18] Alex Hanson, Allen Tu, Geng Lin, Vasu Singla, Matthias Zwicker, and Tom Goldstein. Speedy-splat: Fast 3d gaussian splatting with sparse pixels and sparse primitives. *arXiv preprint arXiv:2412.00578*, 2024. 5

[19] Alex Hanson, Allen Tu, Vasu Singla, Mayuka Jayawardhana, Matthias Zwicker, and Tom Goldstein. Pup 3d-gs: Principled uncertainty pruning for 3d gaussian splatting. *arXiv preprint arXiv:2406.10219*, 2024. 5

[20] Qiaoqiao Hao, Zhaoliang Liu, Duxin Zhu, Jinhe Su, Zheng Gong, Yundong Wu, and Guorong Cai. Leaf: Learning edge attraction filed for precise roof wireframe reconstruction. In *Proceedings of the 2024 8th International Conference on Electronic Information Technology and Computer Engineering*, pages 612–619, 2024. 3

[21] Manuel Hofer, Michael Maurer, and Horst Bischof. Efficient 3d scene abstraction using line segments. *Computer Vision and Image Understanding*, 157:167–178, 2017. 5

[22] Qinghui Hong, Haoyou Jiang, Pingdan Xiao, Sichun Du, and Tao Li. Edcssm: Edge detection with convolutional state space model. *arXiv preprint arXiv:2409.01609*, 2024. 8

[23] Xu Hu, Yuxi Wang, Lue Fan, Junsong Fan, Junran Peng, Zhen Lei, Qing Li, and Zhaoxiang Zhang. Sagd: Boundary-enhanced segment anything in 3d gaussian via gaussian decomposition. *arXiv preprint arXiv:2401.17857*, 2024. 1

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

[25] Qirui Huang, Runze Zhang, Kangjun Liu, Minglun Gong, Hao Zhang, and Hui Huang. Arcpro: Architectural programs for structured 3d abstraction of sparse points. In *CVPR*, 2025. 3

[26] Shangfeng Huang, Ruisheng Wang, Bo Guo, and Hongxin Yang. Pbwr: Parametric-building-wireframe reconstruction from aerial lidar point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 27778–27787, 2024. 3

[27] Junha Hyung, Susung Hong, Sungwon Hwang, Jaeseong Lee, Jaegul Choo, and Jin-Hwa Kim. Effective rank analysis and regularization for enhanced 3d gaussian splatting. *arXiv preprint arXiv:2406.11672*, 2024. 2

[28] Arti Jain and Pradeep Singh. Training-free quantum-inspired image edge extraction method. *arXiv preprint arXiv:2501.18929*, 2025. 8

[29] Wen Jiang, Boshu Lei, and Kostas Daniilidis. Fisherrf: Active view selection and uncertainty quantification for radiance fields using fisher information. *arXiv preprint arXiv:2311.17874*, 2023. 5

[30] Artur Kasymov, Bartosz Czekaj, Marcin Mazur, Jacek Tabor, and Przemysław Spurek. Neggs: Negative gaussian splatting. *Information Sciences*, page 121912, 2025. 2

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

[32] Andreas Kirsch and Yarin Gal. Unifying approaches in active learning and active sampling via fisher information and information-theoretic quantities. *arXiv preprint arXiv:2208.00549*, 2022. 2, 5

[33] 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. 5, 7

[34] Sebastian Koch, Albert Matveev, Zhongshi Jiang, Francis Williams, Alexey Artemov, Evgeny Burnaev, Marc Alexa, Denis Zorin, and Daniele Panozzo. Abc: A big cad model dataset for geometric deep learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9601–9611, 2019. 5

[35] Lei Li, Songyou Peng, Zehao Yu, Shaohui Liu, Rémi Pautrat, Xiaochuan Yin, and Marc Pollefeys. 3d neural edge reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21219–21229, 2024. 1, 2, 5, 6, 7

[36] Yanyan Li, Chenyu Lyu, Yan Di, Guangyao Zhai, Gim Hee Lee, and Federico Tombari. Geogaussian: Geometry-aware gaussian splatting for scene rendering. In *European Conference on Computer Vision*, pages 441–457. Springer, 2024. 2

[37] Yachuan Li, Xavier Soria Poma, Yun Bai, Qian Xiao, Chaozhi Yang, Guanlin Li, and Zongmin Li. Edmb: Edge detector with mamba. *arXiv preprint arXiv:2501.04846*, 2025. 8

[38] Zikuan Li, Honghua Chen, Yuecheng Wang, Sibo Wu, Mingqiang Wei, and Jun Wang. Star-edge: Structure-aware local spherical curve representation for thin-walled edge extraction from unstructured point clouds. *arXiv preprint arXiv:2503.00801*, 2025. 3

[39] Shaohui Liu, Yifan Yu, Rémi Pautrat, Marc Pollefeys, and Viktor Larsson. 3d line mapping revisited. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21445–21455, 2023. 2, 3, 5, 6, 7

[40] Yujia Liu, Stefano D'Aronco, Konrad Schindler, and Jan Dirk Wegner. Pc2wf: 3d wireframe reconstruction from raw point clouds. *arXiv preprint arXiv:2103.02766*, 2021. 3

[41] Xing Liufu, Chaolei Tan, Xiaotong Lin, Yonggang Qi, Jinxuan Li, and Jian-Fang Hu. Sauge: Taming sam for uncertainty-aligned multi-granularity edge detection. *arXiv preprint arXiv:2412.12892*, 2024. 8

[42] Yicheng Luo, Jing Ren, Xuefei Zhe, Di Kang, Yajing Xu, Peter Wonka, and Linchao Bao. Learning to construct 3d building wireframes from 3d line clouds. *arXiv preprint arXiv:2208.11948*, 2022. 3

[43] Xiaoyang Lyu, Yang-Tian Sun, Yi-Hua Huang, Xiuzhe Wu, Ziyi Yang, Yilun Chen, Jiangmiao Pang, and Xiaojuan Qi. 3dgsr: Implicit surface reconstruction with 3d gaussian splatting. *arXiv preprint arXiv:2404.00409*, 2024. 2

[44] David Marr and Ellen Hildreth. Theory of edge detection. *Proceedings of the Royal Society of London. Series B. Biological Sciences*, 207(1167):187–217, 1980. 2

[45] 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 *ECCV*, 2020. 1

[46] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. 32, 2019. 6

[47] Rémi Pautrat, Juan-Ting Lin, Viktor Larsson, Martin R Oswald, and Marc Pollefeys. Sold2: Self-supervised occlusion-aware line description and detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11368–11378, 2021. 5

[48] Xavier Soria Poma, Edgar Riba, and Angel Sappa. Dense extreme inception network: Towards a robust cnn model for edge detection. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 1923–1932, 2020. 1, 2, 5, 8

[49] Mengyang Pu, Yaping Huang, Yuming Liu, Qingji Guan, and Haibin Ling. Edter: Edge detection with transformer. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1402–1412, 2022. 8

[50] Haoxuan Qu, Zhuoling Li, Hossein Rahmani, Yujun Cai, and Jun Liu. Disc-gs: Discontinuity-aware gaussian splatting. *Advances in Neural Information Processing Systems*, 37:112284–112309, 2025. 2

[51] 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. 8

[52] Chris Rorden, Taylor Hanayik, Daniel R Glen, Roger Newman-Norlund, Chris Drake, Julius Fridriksson, and Paul A Taylor. Improving 3d edge detection for visual inspection of mri coregistration and alignment. *Journal of Neuroscience Methods*, 406:110112, 2024. 1

[53] 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. 3

[54] Yunji Seo, Young Sun Choi, Hyun Seung Son, and Youngjung Uh. Flod: Integrating flexible level of detail into 3d gaussian splatting for customizable rendering. *arXiv preprint arXiv:2408.12894*, 2024. 8

[55] Yuang Shi, Géraldine Morin, Simone Gasparini, and Wei Tsang Ooi. Lapisgs: Layered progressive 3d gaussian splatting for adaptive streaming. *arXiv preprint arXiv:2408.14823*, 2024. 8

[56] Yuang Shi, Simone Gasparini, Géraldine Morin, Chenggang Yang, and Wei Tsang Ooi. Sketch and patch: Efficient 3d gaussian representation for man-made scenes. *arXiv preprint arXiv:2501.13045*, 2025. 1

[57] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. pages 746–760. Springer, 2012. 8

[58] Julian Straub, Thomas Whelan, Lingni Ma, Yufan Chen, Erik Wijmans, Simon Green, Jakob J Engel, Raul Mur-Artal, Carl Ren, Shobhit Verma, et al. The replica dataset: A digital replica of indoor spaces. *arXiv preprint arXiv:1906.05797*, 2019. 4, 5, 7, 8

[59] Zhuo Su, Wenzhe Liu, Zitong Yu, Dewen Hu, Qing Liao, Qi Tian, Matti Pietikäinen, and Li Liu. Pixel difference networks for efficient edge detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5117–5127, 2021. 1, 2, 4, 5, 8

[60] Bin Tan, Rui Yu, Yujun Shen, and Nan Xue. Planarsplatting: Accurate planar surface reconstruction in 3 minutes. *arXiv preprint arXiv:2412.03451*, 2024. 3

[61] 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. 2

[62] Yi Wang, Ningze Zhong, Minglin Chen, Longguang Wang, and Yulan Guo. Tangram-splatting: Optimizing 3d gaussian splatting through tangram-inspired shape priors. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 3075–3083, 2024. 2

[63] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 3

[64] Yaniv Wolf, Amit Bracha, and Ron Kimmel. Gs2mesh: Surface reconstruction from gaussian splatting via novel stereo views. *arXiv preprint arXiv:2404.01810*, 2024. 2

[65] Qianyi Wu, Jianmin Zheng, and Jianfei Cai. Surface reconstruction from 3d gaussian splatting via local structural hints. In *European Conference on Computer Vision*, pages 441–458. Springer, 2024. 2

[66] Saining Xie and Zhuowen Tu. Holistically-nested edge detection. pages 1395–1403, 2015. 1

[67] Qingshan Xu, Wanjuan Su, Xinyao Liao, Xuanyu Yi, Jiequan Cui, Wenbing Tao, Yew-Soon Ong, YAN Shuicheng, and Hanwang Zhang. Geo-3dgs: Multi-view geometry consistency for 3d gaussian splatting and surface reconstruction. 2

[68] Nan Xue, Bin Tan, Yuxi Xiao, Liang Dong, Gui-Song Xia, Tianfu Wu, and Yujun Shen. Neat: Distilling 3d wireframes from neural attraction fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19968–19977, 2024. 2, 3, 6, 7

[69] Chenggang Yang, Yuang Shi, and Wei Tsang Ooi. Linegs: 3d line segment representation on 3d gaussian splatting. *arXiv preprint arXiv:2412.00477*, 2024. 3

[70] Wenya Yang, Xiao-Diao Chen, Wen Wu, Hongshuai Qin, Kangming Yan, Xiaoyang Mao, and Haichuan Song. Boosting deep unsupervised edge detection via segment anything model. *IEEE Transactions on Industrial Informatics*, 2024. 8

[71] Yunfan Ye, Renjiao Yi, Zhirui Gao, Zhiping Cai, and Kai Xu. Delving into crispness: Guided label refinement for crisp edge detection. *IEEE Transactions on Image Processing*, 32: 4199–4211, 2023. 8

[72] Yunfan Ye, Renjiao Yi, Zhirui Gao, Chenyang Zhu, Zhiping Cai, and Kai Xu. Nef: Neural edge fields for 3d parametric curve reconstruction from multi-view images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8486–8495, 2023. 1, 2, 5, 6, 7

[73] Yunfan Ye, Kai Xu, Yuhang Huang, Renjiao Yi, and Zhiping Cai. Diffusionedge: Diffusion probabilistic model for crisp edge detection. In *Proceedings of the AAAI conference on artificial intelligence*, pages 6675–6683, 2024. 1, 8

[74] Haiyang Ying and Matthias Zwicker. Sketchsplat: 3d edge reconstruction via differentiable multi-view sketch splatting. *arXiv preprint arXiv:2503.14786*, 2025. 1

[75] Lequan Yu, Xianzhi Li, Chi-Wing Fu, Daniel Cohen-Or, and Pheng-Ann Heng. Ec-net: an edge-aware point set consolidation network. In *Proceedings of the European conference on computer vision (ECCV)*, pages 386–402, 2018. 3

[76] Mulin Yu, Tao Lu, Linning Xu, Lihan Jiang, Yuanbo Xiangli, and Bo Dai. Gsdf: 3dgs meets sdf for improved rendering and reconstruction. *arXiv preprint arXiv:2403.16964*, 2024. 2

[77] Zehao Yu, Torsten Sattler, and Andreas Geiger. Gaussian opacity fields: Efficient and compact surface reconstruction in unbounded scenes. *arXiv preprint arXiv:2404.10772*, 2024. 2

[78] Baowen Zhang, Chuan Fang, Rakesh Shrestha, Yixun Liang, Xiaoxiao Long, and Ping Tan. Rade-gs: Rasterizing depth in gaussian splatting. *arXiv preprint arXiv:2406.01467*, 2024. 2

[79] Wenyuan Zhang, Yu-Shen Liu, and Zhizhong Han. Neural signed distance function inference through splatting 3d gaussians pulled on zero-level set. *Advances in Neural Information Processing Systems*, 37:101856–101879, 2025. 2

[80] Caixia Zhou, Yaping Huang, Mengyang Pu, Qingji Guan, Li Huang, and Haibin Ling. The treasure beneath multiple annotations: An uncertainty-aware edge detector. In *Proceedings of the IEEE/CVF Conference on computer vision and pattern recognition*, pages 15507–15517, 2023. 8

[81] Caixia Zhou, Yaping Huang, Mengyang Pu, Qingji Guan, Ruoxi Deng, and Haibin Ling. Muge: Multiple granularity edge detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 25952–25962, 2024. 8

[82] Caixia Zhou, Yaping Huang, Mochu Xiang, Jiahui Ren, Haibin Ling, and Jing Zhang. Generative edge detection with stable diffusion. *arXiv preprint arXiv:2410.03080*, 2024. 1, 8