Comput. Aided Geom. Des. 119 (2025) 102434

Contents lists available at ScienceDirect

Computer Aided Geometric Design

journal homepage: www.elsevier.com/locate/cagd

ESA-GS: Elongation splitting and assimilation in Gaussian splatting for accurate surface reconstruction ☆

Yuyang Chen, Wenming Wu, Yusheng Peng, Yue Fei, Liping Zheng *

Hefei University of Technology, Hefei, 230601, Anhui, China

ARTICLE INFO

Editor: Shiqing Xin

Keywords:
Gaussian splatting
Surface reconstruction
Elongation splitting and assimilation

ABSTRACT

Recently, 3D Gaussian Splatting (3DGS) has significantly advanced the development of 3D reconstruction by providing efficient and high-quality rendering. 2D Gaussian Splatting (2DGS) introduced two-dimensional surfels as scene primitives to address 3DGS's limitations in surface representation. However, its adaptive control strategy may still result in suboptimal results, especially when dealing with extreme-shaped or large Gaussians on the surface. We propose Elongation Splitting and Assimilation in Gaussian Splatting (ESA-GS) to enhance geometric reconstruction quality by addressing these special Gaussians. Specifically, ESA-GS splits highly elongated Gaussians on the surface into three assimilated Gaussians during the densification process. In addition, ESA-GS adds an opacity degeneration strategy and an additional pruning strategy to remove invalid Gaussians and improve the geometry quality. Experimental results demonstrate that ESA-GS can produce geometrically accurate reconstructed surfaces without sacrificing efficiency in most cases.

1. Introduction

Novel view synthesis (NVS) and accurate surface reconstruction have long been significant challenges in the fields of computer vision and graphics. Successful 3D reconstruction demands both high-quality novel-view rendering and accurate geometry. Neural Radiance Fields (NeRF) (Mildenhall et al., 2020) has made significant progress in this area through implicit radiance fields and differentiable rendering techniques. However, NeRF's reliance on substantial computational resources limits rendering speed, thereby hindering its practical application in real-world scenarios.

Recently, 3D Gaussian splatting (3DGS) (Kerbl et al., 2023) has emerged as a promising alternative, providing an explicit representation method that achieves real-time and high-quality rendering. This approach utilizes 3D Gaussians as primitives for scene representation combined with a tile-based splatting technique, rather than relying on expensive neural field queries. As a result, 3DGS meets the demands for both rendering quality and speed, making it suitable for practical applications in fields such as virtual reality, augmented reality, gaming, and interactive avatars.

Despite the great potential of 3DGS and its rapid advancement across various fields, such as anti-aliasing rendering (Yu et al., 2023), few shot reconstruction (Li et al., 2024; Zhu et al., 2023), dynamic scene modeling (Wu et al., 2023), and animated avatar creation (Hu et al., 2023), it still faces challenges in geometric reconstruction. These challenges stem from the inherent discreteness

☆ This document is the results of the research project funded by the National Natural Science Foundation of China under Grants 62372152 and the Open Project Program of the State Key Laboratory of CAD&CG (Grant No. A2412).

* Corresponding author.
E-mail address: zhenglp@hfut.edu.cn (L. Zheng).

https://doi.org/10.1016/j.cagd.2025.102434
Received 8 March 2025; Accepted 15 April 2025
Available online 17 April 2025

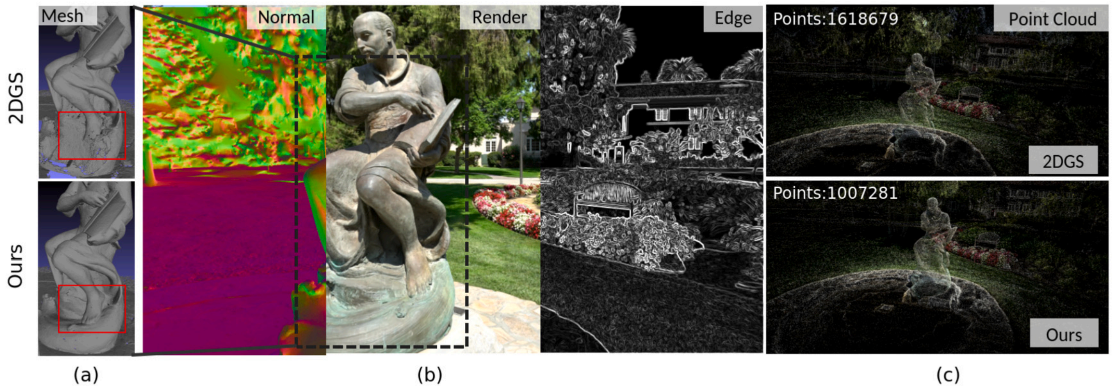

Fig. 1. ESA-GS exhibits better geometric details with fewer points. In the Ignatius scene, our method produces high-quality meshes and realistic images. (a) shows the mesh results, where our method effectively fills holes and smooths artifacts compared to 2DGS. (b) shows the normal map, render results and edge map. (c) illustrates the point cloud distribution, demonstrating our method achieves comparable reconstruction quality with fewer points than 2DGS.

of Gaussian primitives, which lack sufficient geometric support. While many existing 3DGS approaches focus primarily on rendering quality, the geometric accuracy of reconstructed surfaces has received relatively less attention.

It is worth noting the emergence of 2D Gaussian Splatting (2DGS) (Huang et al., 2024), which transforms three-dimensional Gaussian ellipsoids into two-dimensional disks for rendering. Additionally, it incorporates two regularization terms to better fit the 2D primitives to surfaces, significantly improving geometric accuracy in surface reconstruction. However, 2DGS still has limitations in its densification strategy, as it is not optimized to handle extreme or redundant Gaussians, which can lead to inaccuracies in geometry reconstruction. On the other hand, AtomGS (Liu et al., 2024) improves geometric accuracy by atomizing Gaussians whose minor axes are smaller than a fixed size. However, AtomGS can be computationally expensive due to its simplistic approach and lacks effective strategies to address these challenges.

Therefore, we introduce ESA-GS, a novel method aimed at enhancing geometric accuracy through Elongation Splitting and Assimilation (ESA), complemented by opacity degeneration and controlled pruning strategies. As illustrated in Fig. 1, ESA-GS significantly improves the quality of geometric reconstruction with fewer points.

Our proposed ESA-GS optimizes 2DGS by strategically assimilating specific Gaussians, ensuring that under-optimized Gaussians on the surface are properly addressed. ESA-GS offers precise guidance for handling overly elongated and narrow Gaussians, leading to improved geometric optimization. Specifically, during the densification process, overly elongated Gaussians on the surface are split into three assimilated Gaussians along the longest axis. After densification, any remaining narrow Gaussians are further assimilated into fixed-size Gaussians. Additionally, we introduce an opacity degeneration strategy and a novel pruning strategy to eliminate Gaussians that contribute little or could result in erroneous depth estimation. This leads to a lower storage costs and competitive geometric quality. In summary, our contributions are as follows:

- We introduce Elongation Splitting and Assimilation (ESA), which effectively addresses elongated Gaussians on the surface areas, leading to more accurate 3D geometric reconstruction quality.
- We design an opacity degeneration strategy and a novel pruning strategy that enhances reconstruction accuracy and simultaneously reduces storage costs by eliminating redundant Gaussians.
- Through both experimental and theoretical analysis, our proposed ESA-GS achieves superior performance across multiple benchmark datasets, delivering improved geometric accuracy with fewer Gaussians.

2. Related work

2.1. Novel view synthesis

Novel View Synthesis (NVS) aims to generate images from new viewpoints based on images captured from various source viewpoints around a 3D scene. The effectiveness of rendering these new viewpoints depends largely on the accuracy of the underlying 3D scene representation. Early methods (Lombardi et al., 2019; Sitzmann et al., 2018; Sun et al., 2021) often used volume-based approaches that discretized space into voxels to model internal structures in detail. However, these approaches were constrained by resolution limits and high computational costs. In contrast, implicit neural representations (Jiang et al., 2020; Ran et al., 2022) utilize continuous functions to model complex geometry and appearance, removing the need for spatial discretization and opening new possibilities for novel view synthesis.

NeRF (Mildenhall et al., 2020) has made groundbreaking contributions by employing a coordinate-based MLP to encode radiance, enabling unprecedented detail and realism in novel view synthesis. It excels at capturing complex light interactions and intricate surface details. Building on the original NeRF, Mip-NeRF (Barron et al., 2021) addresses common aliasing artifacts with the introduction of cone-based anti-aliased rendering. On the other hand, due to NeRF's slow training speed, methods like InstantNGP (Müller et al., 2022) and Plenoxels (Yu et al., 2021) have emerged, accelerating training by utilizing simplified data structures like multiresolution hash encoding grids and others, albeit with some trade-offs in quality. Although state-of-the-art NeRF-based methods can reconstruct unbounded scenes with high quality and render them at interactive rates, they still require substantial computational resources and prolonged training times. The introduction of 3DGS (Kerbl et al., 2023) marks a new phase in novel view synthesis. 3DGS achieves superior rendering quality and real-time speeds, effectively addressing the training and rendering speed limitations of earlier NeRF-based approaches.

2.2. Surface reconstruction

The goal of surface reconstruction is to recover the surface of 3D objects from various forms of input data, with applications in fields such as virtual reality, augmented reality, and cultural heritage preservation. Many pioneering approaches rely on point clouds, voxels, triangular meshes, or implicit fields, using neural networks to predict surface models directly from one or more images in an end-to-end manner. NeRF (Mildenhall et al., 2020) opened new possibilities for surface reconstruction, inspiring a series of subsequent works. However, due to the lack of geometric constraints in implicit radiance fields and the inherent limitations of the MLP architecture, these methods often introduce artifacts and inaccuracies, preventing them from achieving the desired level of surface fidelity.

3DGS also faces challenges in extracting precise surfaces due to the discreteness and unstructured nature of 3D Gaussians. Many previous methods (Dai et al., 2024; Gu'edon and Lepetit, 2023; Turkulainen et al., 2024; Yu et al., 2024b; Chen et al., 2023) have focused on applying geometric regularization, which has proven effective. SuGaR (Gu'edon and Lepetit, 2023) employs signed distance functions (SDF) and density regularization to supervise Gaussian distributions, aligning them with object surfaces before extracting meshes through Poisson surface reconstruction (Kazhdan and Hoppe, 2013). AtomGS (Liu et al., 2024) improves geometric reconstruction quality by atomizing irregularly shaped Gaussians into fixed-size Gaussians. GS2Mesh (Wolf et al., 2024) utilizes the view synthesis capabilities of 3D Gaussian Splatting (3DGS) to overcome the limitations of direct surface reconstruction based on Gaussian positions, thereby enabling high-quality surface reconstruction. Meanwhile, 2DGS (Huang et al., 2024) compresses 3D Gaussian volumes into oriented 2D Gaussians, better fitting surfaces and avoiding ambiguous depth estimations. In addition to geometric regularization, 2DGS utilizes truncated signed distance function (TSDF) fusion and Marching Cubes (Lorensen and Cline, 1987) for mesh extraction, demonstrating good adaptability to outliers and noise. GaussianSurfels (Dai et al., 2024) adopts a similar approach, introducing several additional regularization terms. On the other hand, GOF (Yu et al., 2024b) tackles surface reconstruction from a different perspective by focusing on unbounded scenes. It establishes a Gaussian opacity field based on ray tracing and directly extracts geometric surfaces using SDF level sets, without relying on Poisson reconstruction or TSDF fusion as in SuGaR and 2DGS.

In this paper, our method not only focuses on regularization but also enhances the representation of detailed areas by addressing extreme Gaussians. This approach allows for the learning of Gaussians that align more accurately with the geometry. In addition, we incorporate auxiliary strategies to eliminate redundant and erroneous Gaussians.

2.3. Effective additional measures of 3D Gaussians

Despite the significant potential of 3DGS, there remains considerable room for optimization, particularly with respect to Gaussian artifacts and multi-resolution handling. Several works have improved reconstruction quality and robustness by improving Gaussian representations through techniques such as low-pass filtering (Yu et al., 2023), multi-scale Gaussian representations (Lu et al., 2023; Ren et al., 2024), and refined Gaussian kernel functions (Yu et al., 2024a). Other efforts have focused on optimizing the high storage costs associated with 3DGS, with approaches such as Papantonakis et al. (2024), LightGS (Fan et al., 2023), SpikingGS (Zhang et al., 2024a), and TrimGS (Fan et al., 2024). Notably, TrimGS emphasizes improving geometric quality in reconstruction. Densification is another crucial aspect of 3DGS, and several works have optimized this component, Pixel-GS (Zhang et al., 2024b) primarily modify gradient calculations within the densification process, it introduces a gradient scaling strategy to suppress artifacts near the camera. Additionally, GS-LPM (Yang et al., 2024) addresses errors caused by insufficient densification in multi-view scene by incorporating novel solutions and periodically resetting Gaussians to mitigate occlusion issues. In this paper, we propose a novel method that modifies the densification strategy and combines it with an opacity degeneration strategy and a new pruning approach to achieve accurate surface reconstruction.

3. Preliminaries

3.1. 3D Gaussian splatting

3D Gaussian splatting (Kerbl et al., 2023) provides an explicit representation of 3D scenes, starting from a set of SfM(Structure from Motion) point clouds to derive a collection of anisotropic 3D Gaussians $\{\mathcal{G}_i\}$. Each Gaussian is defined by its position $\mu_i \in R^3$, opacity $\alpha_i \in [0,1]$, anisotropic covariance matrix $\Sigma_i \in R^{3\times3}$, and view-dependent color represented by spherical harmonic coefficients. The covariance matrix can be decomposed into: $\Sigma_i = \mathbf{R}_i\mathbf{S}_i\mathbf{S}_i^T\mathbf{R}_i^T$, where $\mathbf{R}_i \in R^{3\times3}$ denotes the rotation matrix and $\mathbf{S}_i \in R^{3\times1}$ denotes the scaling matrix. The 3D Gaussian primitives can be represented as follows (Kerbl et al., 2023):

$$\mathcal{G}_i(x) = e^{-\frac{1}{2}(x-\mu_i)^T\Sigma_i^{-1}(x-\mu_i)} \tag{1}$$

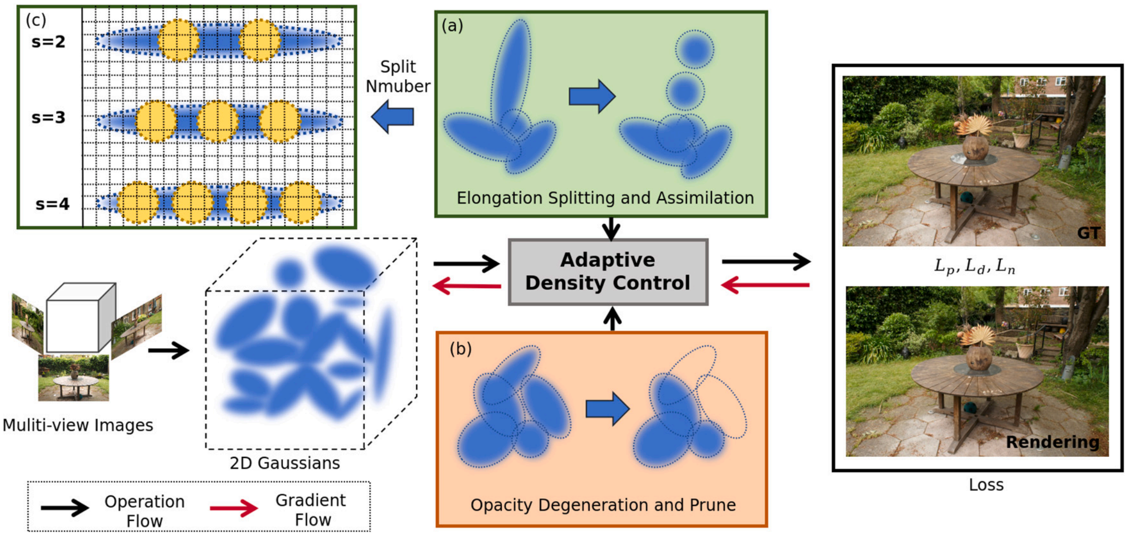

Fig. 2. The pipeline of the proposed ESA-GS. The three strategies (ESA, Opacity Degeneration and Prune) proposed in our work have been integrated into the adaptive density control process. Specifically, (a) Elongation Splitting and Assimilation is designed to address the challenges posed by irregular Gaussians. (b) Opacity Degeneration and Prune aim to eliminate redundant Gaussians, thereby improving reconstruction quality. Additionally, (c) demostrates the influence of varying split numbers s on elongated Gaussians. The attributes of Gaussian are optimized through a combination of photometric loss L_p, edge-aware depth distortion loss L_d and normal consistency loss L_n.

During the rendering process, each 3D Gaussian is projected into 2D screen space to obtain the corresponding 2D Gaussian projection. These projected 2D Gaussians are then assigned to different tiles, and a point-based alpha blending process is used to accumulate the color for each pixel. Assuming p is the target pixel, its color can be expressed as (Kerbl et al., 2023):

$$C(p) = \sum_{i \in N} c_i \sigma_i \prod_{j=1}^{i-1} (1 - \sigma_j), \sigma_i = \alpha_i \mathcal{G}_i'(p) \tag{2}$$

where N represents the set of Gaussians influencing pixel p, sorted in descending order by depth. c_i denotes the color of the Gaussians, generated from the coefficients of the spherical harmonics and dependent on the viewing angle. The blending weight σ_i is determined by the corresponding 2D projection of the Gaussian $\mathcal{G}_i'$ and its opacity α_i.

3.2. 2D Gaussian splatting

2D Gaussian splatting (Huang et al., 2024) builds upon 3DGS by transforming the volume of 3D ellipsoids into a set of oriented 2D Gaussian disks (referred to as surfaces) to address multi-view inconsistencies in 3DGS. It also introduces an improved method for computing ray-splat intersections, leading to more accurate depth estimations. The covariance of 2D Gaussians is described using two tangent vectors $t_{n,u}$ and $t_{n,v}$, along with a scaling vector $\mathbf{S}_n = (s_{n,u}, s_{n,v})$. Additionally, 2DGS incorporates two new loss terms: a depth distortion term that helps align the Gaussians more closely along the rays, and a normal consistency term that aligns the surface normals with those estimated from the depth map. These measures significantly improve both geometric reconstruction and novel view synthesis capabilities.

4. Method

This section outlines the details of the proposed ESA-GS. In Sec 4.1, we introduce the Elongation Splitting and Assimilation (ESA) strategy to address extreme Gaussians on the surface, along with a depth-based stratification of the Gaussians. In Sec 4.2, we propose an opacity degeneration strategy to improve geometric reconstruction quality. In Sec 4.3, we present a new pruning strategy to eliminate redundant and erroneous Gaussians. Finally, we present the components of the training loss in Sec 4.4. The pipeline is shown in Fig. 2.

4.1. Elongation splitting and assimilation

Due to the intrinsic properties of Gaussians, the rendered depth map is computed using alpha blending and can be expressed as (Kerbl et al., 2023):

$$D(p) = \sum_{i \in N} d_i \sigma_i \prod_{j=1}^{i-1} (1 - \sigma_j), \tag{3}$$

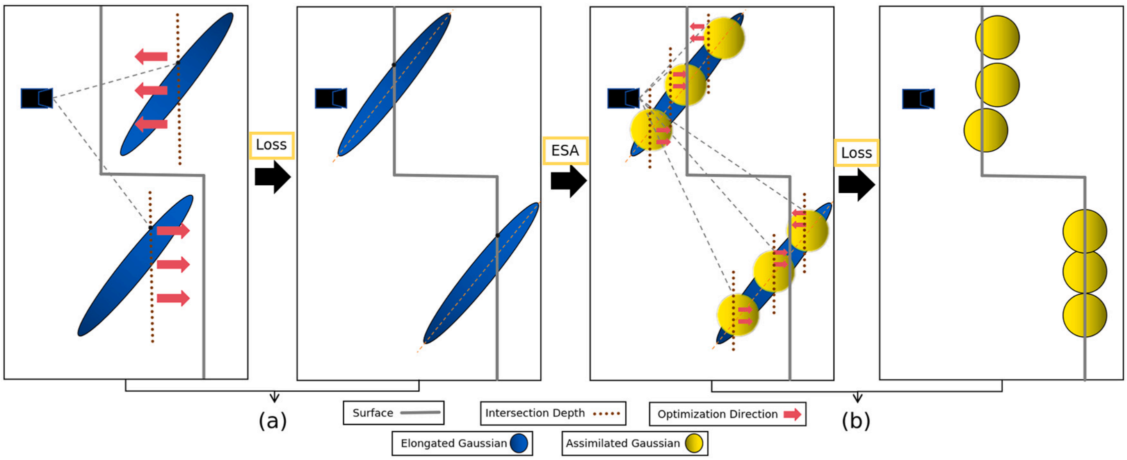

Fig. 3. Illustration of the Elongation Splitting and Assimilation. (a). The original depth alignment caused the elongated Gaussians to produce protrusions on the scene surface. (b). By splitting the elongated Gaussians near the surface into three assimilated Gaussians, the fit to the scene surface is improved. (For interpretation of the colors in the figure(3), the reader is referred to the web version of this article.)

where d_i is the depth of the i-th Gaussian under the current camera view. However, directly using the center of the Gaussians as the depth leads to unaccurate depth maps when the Gaussians are very narrow or big. This limitation stems from the shape of the Gaussians, which introduces geometric errors. Even techniques like the ray-splat intersection used in 2DGS (Huang et al., 2024) do not completely eliminate these errors. As shown in Fig. 3(a), although the depths are aligned with the surface after optimization, there is still a noticeable shape deviation from the actual surface, particularly for excessively elongated Gaussians.

In this section, we introduce the Elongation Splitting and Assimilation (ESA) strategy to overcome the above limitations, as illustrated in Fig. 3(b). This method targets excessively elongated Gaussians on the surface, defined as those with a longest-to-smallest axis ratio exceeding a specified threshold. Rather than including these elongated Gaussians in the original densification process, ESA splits them into three smaller, assimilated Gaussians during densification. The Fig. 2(c) demonstrates the effects of varying split number s on elongated Gaussians. It can be observed that when $s = 2$, a portion of the original Gaussians' geometric structure is lost. When $s = 3$, the geometric structure of the original Gaussian is largely preserved, with only minor distortions at the edges. However, these edge distortions often introduce artifacts, such as burrs, that can compromise the quality of the reconstruction. When $s = 4$, the geometric structure of the Gaussian is preserved, but the increased number of Gaussians may potentially tend to more overlapping regions. Furthermore, the edges of elongated Gaussians are more prone to incorrect retention, further degrading the reconstruction results. Crucially, the splitting direction follows the longest axis of the elongated Gaussian, thus preserving the original geometry as much as possible. After the densification process, the narrow Gaussians are assimilated to further enhance surface reconstruction accuracy.

Additionally, Gaussians should be allocated to different layers based on their depth information with each layer corresponding to a unique assimilation size. This approach accounts for the fact that Gaussians at different depth levels often require varying levels of detail. The total number of layers L_{max} is calculated as follows:

$$L_{max} = max(\lceil \log_2(d_{max}/d_{min}) \rceil, 3), \tag{4}$$

where, d_{max} and d_{min} represent the farthest and closest distances from the SfM point cloud to each camera, respectively. For each viewpoint, the layer of a Gaussian is determined by its distance from the current camera, which falls into one of the intervals between d_{max} and d_{min}.

The initial assimilation size S_{init} is determined based on the SfM point cloud. Specifically, we obtain S_{init} by computing the 0.3 quantile of the size distribution of the initialized Gaussians. This size is optimal for capturing fine details and avoiding interference from background elements. For each subsequent layer, the assimilation size is halved relative to the previous layer, and as the process progresses, the assimilation sizes gradually decrease. This will help ensure a smoother transition between adjacent layers when the densification reaches the midpoint of the iterations ($0.99^{70} \approx 0.5$), with subsequent iterations further refining the process. Once the densification process is complete, the assimilation sizes for each layer are fixed. Similar to AtomGS (Liu et al., 2024), we assimilate Gaussians with a minor axis smaller than the assimilation size ($S_{min} < S_{init}$) to form uniformly shaped Gaussians with a size set to $S_x = S_y = S_{init}$. The key difference is that while AtomGS begins atomization at the beginning and applies it to all Gaussians, our post-assimilation strategy specifically targets narrow Gaussians near the surface, leading to more accurate reconstructions with fewer side effects.

4.2. Opacity degeneration strategy

Depth alignment has been shown to be effective for surface reconstruction in many works. However, since the rendered depth of Gaussians is obtained through alpha blending, it is influenced by several factors, including position, opacity, and size. Among

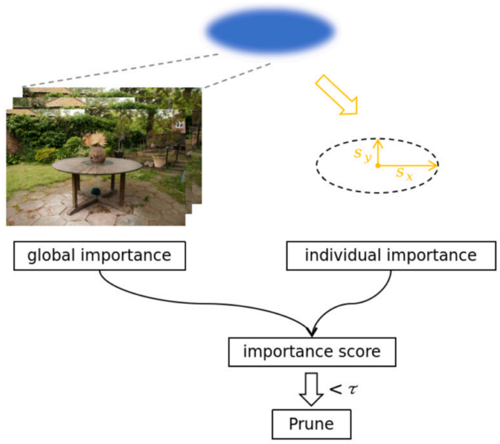

Fig. 4. Illustration of the Prune Strategy.

these, certain Gaussians located at substantial distances from the surface tend to converge to suboptimal opacity values, which may adversely affect the overall reconstruction quality. Many methods address this by introducing opacity loss terms that push opacity towards 0 or 1. However, this approach may hinder optimization process or result in overfitting to the training views. To address this, we propose applying a degeneration coefficient λ to reduce the opacity of the Gaussians every certain number of iterations ($\alpha' = \lambda\alpha$). This enhances the reconstruction quality and avoids the risk of overfitting.

During training, Gaussians close to the surface are typically visible from multiple viewpoints, allowing their gradients to accumulate across these views, resulting in larger gradient values. This enables their opacity to be adjusted appropriately according to the optimization process. In contrast, Gaussians farther from the surface, especially floaters, receive weaker multi-view constraints and fail to accumulate sufficient gradients. As a result, their opacity may remain stuck in an incorrect range, which negatively affects the quality of the rendered output.

We aim to eliminate erroneous Gaussians that are far from the surface while retaining those closer to it. After the opacity degeneration process, the opacities of all Gaussians are reduced, effectively clearing the accumulated opacity gradients. This allows Gaussians near the surface to quickly accumulate sufficient gradients and remain align with the surface. In contrast, Gaussians farther from the surface fail to gather enough gradients to enhance their opacity. As a result, these distant Gaussians are progressively pruned. In summary, the opacity degeneration strategy significantly improves surface reconstruction quality by selectively retaining Gaussians near the surface and effectively removing anomalous Gaussians located at a distance.

4.3. Prune strategy

Inspired by TrimGS (Fan et al., 2024), we recognize that pruning certain Gaussians can significantly enhance the geometric quality of the reconstruction. However, the pruning strategy in TrimGS operates as a post-processing step, requiring a fully formed Gaussian model before any pruning can take place. This introduces additional time costs and limits flexibility. In contrast, our approach introduces a novel pruning criterion that is seamlessly integrated into the training process, avoiding the need for separate post-processing or additional tuning operations. The overall importance score I is calculated based on two key components: $I = I_g I_i$ (Fig. 4).

Based on Fan et al. (2024). The global importance I_g can be expressed as:

$$I_g = \frac{1}{|P_k|} \sum_{p \in P_k} (\alpha_n)^{\gamma} \left(\prod_{j=1}^{n(p)-1} (1 - \alpha_j) \right)^{(1-\gamma)} \tag{5}$$

where P_k presents the 2D projected area of the n-th Gaussian in the k-th view, $n(p)$ denotes the sorted set of Gaussians along the ray, determined by alpha blending. Set γ to a default value of 0.5. The final global score is obtained by averaging the contributions from the top five views with the highest scores.

The individual importance I_i is defined as:

$$I_i = (V_n)^{\beta}, V_n = min\left(max\left(\frac{V_G}{V_{G90}}, 0 \right), 1 \right) \tag{6}$$

where V_G represents the volume of the corresponding Gaussian. Since our work builds upon 2DGS (Huang et al., 2024), the Gaussian volume can be expressed as $V_G = S_x S_y$. To ensure that the Gaussian volume remains within a reasonable range, it is normalized based on the 90% of volumes among all Gaussians (V_{G90}), ensuring that the volume is constrained within the range (0, 1). This normalization helps prevent floating Gaussians or excessively large Gaussians from having disproportionately high importance scores. Additionally, it ensures better alignment with the overall panorama of assimilated Gaussians, leading to a more balanced representation of their contributions to the scene.

4.4. Training loss

To guide the optimization of Gaussians, our total loss consists of three components, similar to 2DGS (Huang et al., 2024): photometric loss L_p, edge-aware depth distortion loss L_d and normal consistency loss L_n. Therefore, the total loss L_{total} is expressed as:

$$L_{total} = \lambda_p L_p + \lambda_d L_d + \lambda_n L_n \tag{7}$$

Photometric Loss. The photometric loss L_p is consistent with 3DGS (Kerbl et al., 2023). This loss aims to minimize the difference between the rendered image $\widetilde{I}$ and the corresponding ground truth (GT) image I, and is typically formulated as:

$$L_p = 0.8 \cdot L_1(\widetilde{I}, I) + 0.2 \cdot L_{DSSIM}(\widetilde{I}, I) \tag{8}$$

Edge-aware Depth Distortion Loss. The original depth distortion loss in 2DGS aims to concentrate Gaussians along the ray. However, we observed that this approach could result in excessive smoothing in high-frequency regions, leading to a loss of sharp details in the reconstructed surface. To address this limitation, we introduced a new weight function $\omega(|\nabla I|) = e^{-\nabla I}$, which prioritizes regions with strong gradients (edges) and reduces the weight in flatter areas, effectively preserving high-frequency details. Consequently, the updated depth distortion term is formulated as:

$$L_n = \omega(|\nabla I|) \sum_{i,j} \omega_i \omega_j |z_i - z_j| \tag{9}$$

where $|\nabla I|$ denotes edge map from GT image, w denotes the blending weight, z is the intersection depth.
Normal Consistency Loss. We also adopt the normal consistency loss L_n from 2DGS to help align the rendered normals with those derived from the depth map:

$$L_d = \sum_i \omega_i (1 - n_i^T N) \tag{10}$$

where n_i denotes the normal of the splat, and N represents the normal estimated from the gradient of the depth map.

5. Experiment

5.1. Evaluation metrics and datasets

We conduct experiments on three widely used public datasets: the DTU dataset (Aanæs et al., 2016), the Tanks and Temples (TnT) dataset (Knapitsch et al., 2017), and the MipNeRF360 dataset (Barron et al., 2021). The first two datasets are used to assess the quality of surface reconstruction. Following previous work (Huang et al., 2024), we select 15 scenes from the DTU dataset and 6 scenes from the TnT dataset for evaluation, using the Chamfer Distance (CD) and F1 score as evaluation metrics respectively. We also report training time to evaluate the efficiency of our method. The MipNeRF360 dataset is primarily used to evaluate novel view synthesis performance, with PSNR, SSIM (Wang et al., 2004) and LPIPS (Zhang et al., 2018) as evaluation metric. Using the camera poses provided by these datasets, we employ COLMAP to generate sparse point clouds for each scene as the initialization step. This experimental setup enables a thorough evaluation of effectiveness across various tasks and datasets.

5.2. Implementation details

We build our method upon the public code of 2DGS (Huang et al., 2024). We adopt the training strategies, hyperparameter settings, and evaluation scripts from 2DGS, with all scenes trained for 30,000 iterations. In addition, we use the densification method and appearance model proposed by GOF (Yu et al., 2024b). The Elongation Splitting and Assimilation (ESA) process begins after 1,000 iterations and is performed every 100 iterations thereafter. Opacity degeneration is applied every 500 iterations during the densification process. The assimilation step is applied every 1,000 iterations after densification, and the pruning strategy is executed three times during training. All experiments are conducted on a single NVIDIA RTX 4090 GPU.

We select several state-of-the-art methods for surface reconstruction based on NeRF and 3DGS for comparison. The implicit NeRF-based methods include VolSDF (Yariv et al., 2021), NeuS (Wang et al., 2021) and Neuralangelo (Li et al., 2023). The 3DGS-based methods are 3DGS (Kerbl et al., 2023), SuGaR (Gu'edon and Lepetit, 2023), G-Surfels (Dai et al., 2024), 2DGS (Huang et al., 2024), GOF (Yu et al., 2024b), TrimGS (Fan et al., 2024) and GS2Mesh (Wolf et al., 2024).

5.3. Comparisons

For the DTU dataset, our method achieves superior geometric reconstruction accuracy without incurring additional time costs as shown in Table 1. Fig. 5 presents a qualitative comparison of meshes using G-Surfel (Dai et al., 2024), 2DGS (Huang et al., 2024),

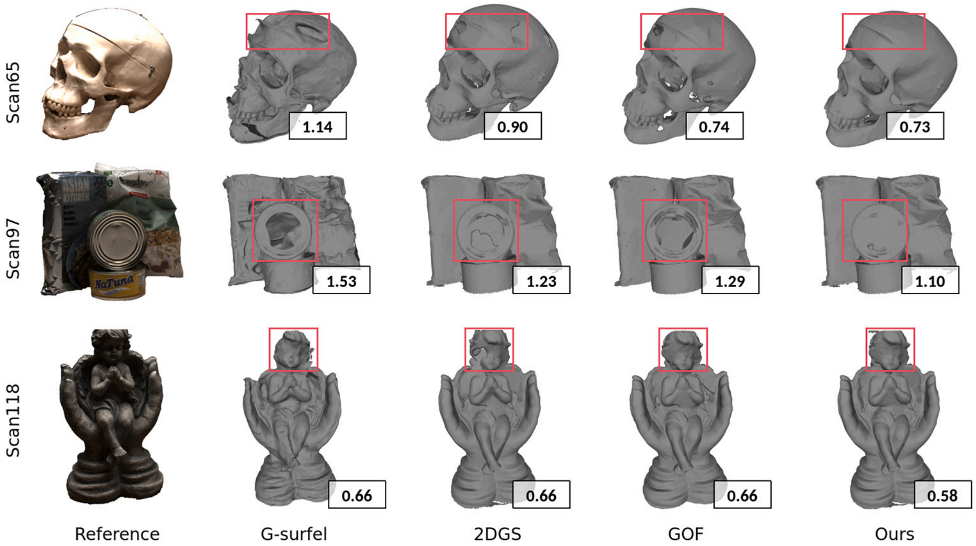

Fig. 5. Comparison of surface reconstruction methods on the DTU Dataset. We show the Chamfer Distance in the bottom right corner of the image.

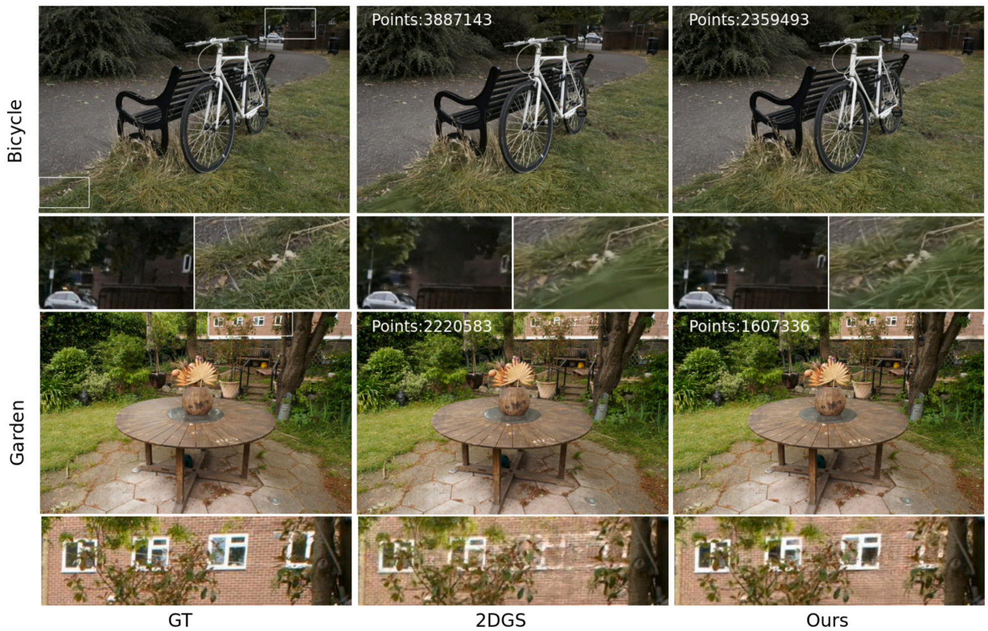

Fig. 6. Quantitative results of rendering quality on the MipNeRF360 Dataset. Notably, the number of Gaussians used in our method is significantly reduced compared to 2DGS. At the same time, we achieved an improvement in rendering quality in certain areas.

GOF (Yu et al., 2024b), and our method on DTU scenes 65, 97, and 118. Our method achieves superior performance, demonstrating both stability and accuracy in surface reconstruction. In contrast, other methods show inconsistencies that result in inaccuracies in the reconstructed surfaces. These results underscore the effectiveness of our method in producing reliable, high-quality geometric representations.

To evaluate the performance of novel view synthesis, our method demonstrates rendering performance comparable to 2DGS (Huang et al., 2024) with fewer Gaussian points, and it outperforms other methods, as shown in Table 2. The visual results in Fig. 6 further illustrate that our method improves rendering quality in certain scenes and maintains a competitive training time. This indicates our method balances reconstruction accuracy, training efficiency, and storage efficiency.

For the TnT dataset, our method achieves an F1 score comparable to that of GOF (Yu et al., 2024b) as shown in Table 3, positioning it among the top-performing approaches. Additionally, the computational time is similar to that of 2DGS. As illustrated in Fig. 7, our method illustrates better geometric consistency than 2DGS, demonstrating its effectiveness in maintaining high-quality reconstructions and optimizing processing time.

Table 1
Quantitative Comparison on 15 scenes from DTU dataset. We report the Chamfer Distance (CD) and average training time. * means our re-implementation. , , indicate the best, the second best, and the third best respectively. (For interpretation of the references to color please refer to the web version of this article.)

	24	37	40	55	63	65	69	83	97	105	106	110	114	118	122	MeanCD	Time
VolSDF	1.14	1.26	0.81	0.49	1.25	0.70	0.72	1.29	1.18	0.70	0.66	1.08	0.42	0.61	0.55	0.86	>12 h
NeuS	1.00	1.37	0.93	0.43	1.10	0.65	0.57	1.48	1.09	0.83	0.52	1.20	0.35	0.49	0.54	0.84	>12 h
Neuralangelo	0.37	0.72	0.35	0.35	0.87	0.54	0.53	1.29	0.97	0.73	0.47	0.74	0.32	0.41	0.43	0.61	>12 h
3DGS	2.14	1.53	2.08	1.68	3.49	2.21	1.43	2.07	2.22	1.75	1.79	2.55	1.53	1.52	1.50	1.96	11.2 m
SuGaR	1.47	1.33	1.13	0.61	2.25	1.71	1.15	1.63	1.62	1.07	0.79	2.45	0.98	0.88	0.79	1.33	1 h
2DGS*	0.49	0.80	0.34	0.43	0.96	0.90	0.81	1.23	1.23	0.64	0.67	1.32	0.41	0.68	0.50	0.76	18.8 m
GOF	0.50	0.82	0.37	0.37	1.12	0.74	0.73	1.18	1.29	0.68	0.77	0.90	0.42	0.66	0.49	0.74	2 h
Trim2DGS	0.48	0.82	0.44	0.45	0.95	0.75	0.74	1.18	1.13	0.72	0.70	0.99	0.42	0.62	0.50	0.72	30 m
G-surfels	0.66	0.93	0.54	0.41	1.06	1.14	0.85	1.29	1.53	0.79	0.82	1.58	0.45	0.66	0.53	0.88	14.4 m
GS2Mesh	0.59	0.79	0.70	0.38	0.78	1.00	0.69	1.25	0.96	0.59	0.50	0.68	0.37	0.50	0.46	0.68	20 m
Ours	0.44	0.74	0.31	0.36	0.87	0.73	0.72	1.23	1.10	0.66	0.64	1.16	0.36	0.58	0.44	0.68	19.0 m

Table 2
Quantitative Comparison from MipNeRF360 dataset. We report the PSNR, SSIM and LPIPS. * means our re-implementation.

	Outdoor scenes			Indoor scenes		
	PSNR ↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
NeRF	21.46	0.458	0.515	26.84	0.790	0.370
Deep Blending	21.54	0.524	0.364	26.40	0.844	0.261
Instant NGP	22.90	0.566	0.371	29.15	0.880	0.216
Mip-NeRF360	24.47	0.691	0.283	31.72	0.917	0.180
3DGS	24.24	0.705	0.283	30.99	0.926	0.199
SuGaR	22.76	0.631	0.349	29.44	0.911	0.216
2DGS*	25.25	0.793	0.235	31.24	0.920	0.203
GOF	24.76	0.742	0.225	30.80	0.928	0.167
Trim2DGS	24.11	0.714	0.246	29.84	0.915	0.193
Ours	25.37	0.799	0.229	31.18	0.920	0.203

Table 3
Quantitative results on the Tanks and Temples Dataset. We report the F1 score and average training time. * indicates our re-implementation, while other baselines scores are taken directly from the respective papers.

F1-Score↑	NeuS	Geo-NeuS	Neuralangelo	3DGS	SuGaR	2DGS*	GOF	Ours
Barn	0.29	0.33	0.70	0.13	0.14	0.41	0.51	0.45
Caterpillar	0.29	0.26	0.36	0.08	0.16	0.24	0.41	0.28
Courthouse	0.17	0.12	0.28	0.09	0.08	0.16	0.28	0.21
Ignatius	0.83	0.72	0.89	0.04	0.33	0.52	0.68	0.73
Meetingroom	0.24	0.20	0.32	0.01	0.15	0.17	0.28	0.22
Truck	0.45	0.45	0.48	0.19	0.26	0.45	0.58	0.53
Mean	0.38	0.35	0.50	0.09	0.19	0.33	0.46	0.40
Time	>24 h	>24 h	>24 h	7.9 m	2 h	34.2 m	2 h	35.1 m

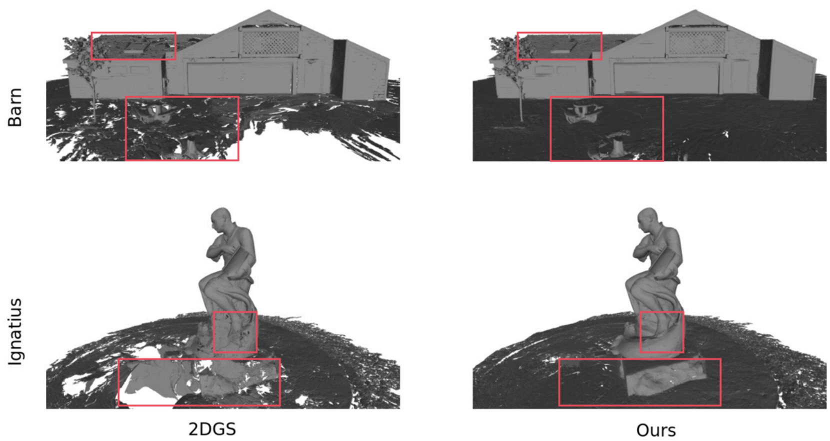

Fig. 7. Qualitative comparison of meshes on the Tanks and Temples Dataset. It can be observed that our method successfully fills in many missing areas and restores fine details.

Table 4
Quantitative Results of Ablation on the DTU and Tanks and Temples datasets. We report Chamfer Distance (CD) and the number of Gaussians #G for the DTU dataset. For the Tanks and Temples dataset, we report Precison, Recall, F1-Score and #G. “OD” refers to Opacity Degeneration.

	DTU		Tanks and Temples			
Model Setting	CD↓	#G↓	Precision↑	Recall↑	F1-Score↑	#G↓
w/o ESA	0.73	**148 K**	0.33	0.42	0.35	**480 K**
w/o OD	0.71	192 K	0.36	0.44	0.38	686 K
w/o Prune	0.71	216 K	0.35	**0.45**	0.39	634k
2DGS	0.76	200 K	0.27	0.39	0.33	748 K
Full model	**0.68**	170 K	**0.37**	**0.45**	**0.40**	568 K

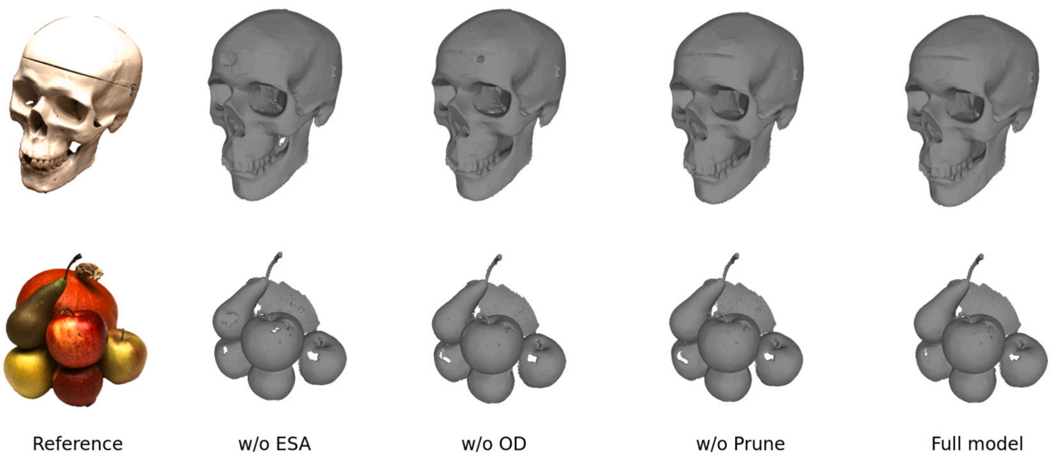

Fig. 8. Qualitative Results of Ablation on the DTU Dataset. It can be observed that ESA and opacity degeneration (OD) contribute to filling in missing areas and restoring the surface, with the pruning method causing no measurable degradation in geometric precision.

Table 5
Performance of ESA-GS on the DTU Dataset with different split number s. We report Chamfer Distance(CD) and the number of Gaussians #G.

	24	37	40	55	63	65	69	83	97	105	106	110	114	118	122	CD↓	#G↓
s = 2	0.46	0.74	0.32	0.38	0.90	0.75	0.73	1.26	1.12	0.66	0.64	1.25	0.36	0.61	0.44	0.70	**152k**
s = 3	0.44	0.74	0.31	0.36	0.87	0.73	0.72	1.23	1.10	0.66	0.64	1.16	0.36	0.58	0.44	**0.68**	170k
s = 4	0.46	0.74	0.32	0.37	0.89	0.75	0.73	1.26	1.10	0.68	0.62	1.14	0.35	0.60	0.43	0.69	193k

5.4. Ablation

Effectiveness of major components. To validate the effectiveness of ESA, opacity degeneration, and the pruning strategy, we conduct experiments by excluding each component individually from full model while keeping the others intact. As shown in Table 4 and Fig. 8, all three components improve geometric reconstruction accuracy. Each component serves a distinct purpose: the ESA module plays a critical role in surface reconstruction. Although it increases the number of Gaussians, it remains essential to our method. Meanwhile, both the opacity degeneration and pruning module help reduce the overall number of Gaussians, keeping it significantly lower than 2DGS. These results demonstrate that both strategies are beneficial for surface reconstruction. Together, they provide valuable insights into the contributions of each module, showing how their integration collectively enhances model performance. Overall, the opacity degeneration and pruning measures effectively counterbalance the increase in Gaussian points and training time caused by ESA, achieving a dynamic balance in efficiency.

Elongation split number. We also investigated the impact of different elongation split numbers s on the results. Quantitative results are presented in Table 5 and Table 6. The experimental results show that as the number of splits increases, the total number of Gaussians also increases. When $s = 2$, the number of Gaussians is relatively small, but the performance across all three datasets remains suboptimal. When $s = 4$, the performance is still inferior compared to $s = 3$, which can be attributed to overlapping phenomena or redundant Gaussians generated by incorrectly retained edges of elongated Gaussians. From this, we can infer that increasing the value of s does not improve the overall reconstruction quality and may instead degrade it due to overlapping phenomena. Therefore, we decide to split each elongated Gaussian into three assimilated Gaussians. This strategy ensures that the number of Gaussians does not increase significantly and preserves the original geometric integrity.

Table 6
Performance of ESA-GS on the MipNeRF360 and Tanks and Temples dataset with different split number s.

	MipNeRF360				Tanks and Temples			
	Outdoor scenes		Indoors scenes		Barn		Caterpillar	
	PSNR↑	#G↓	PSNR↑	#G↓	F1-Score↑	#G↓	F1-Score↑	#G↓
s = 2	25.19	**189 K**	30.95	**614 K**	0.31	**241 K**	0.27	**464 K**
s = 3	**25.37**	193 K	**31.18**	649 K	**0.45**	380 K	**0.28**	506 K
s = 4	25.24	196 K	31.09	689k	0.37	383 K	0.27	515 K

6. Conclusion

We propose ESA-GS, a novel method that enhances surface reconstruction accuracy by splitting and assimilating elongated Gaussians near the surface. Our approach significantly improves the geometric precision of the reconstruction. To complement this, we introduce an opacity degeneration strategy and a new pruning strategy during training, which effectively reduces memory consumption. Extensive experiments show that our method achieves significant improvements in surface reconstruction.

However, similar to previous methods, it struggles with specular reflections and requires tuning several hyperparameters. Exploring adaptive adjustment of the split count s based on Gaussian characteristics will be a key focus of our future research. Additionally, the reduced number of points may impact rendering quality in certain scenes compared to 2DGS, indicating the need for further optimization.

CRediT authorship contribution statement

Yuyang Chen: Conceptualization, Formal analysis, Investigation, Methodology, Writing – original draft, Writing – review & editing. **Wenming Wu:** Methodology, Supervision, Writing – review & editing. **Yusheng Peng:** Methodology, Writing – review & editing. **Yue Fei:** Methodology, Writing – review & editing. **Liping Zheng:** Conceptualization, Funding acquisition, Supervision, Validation.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

References

Aanæs, H., Jensen, R.R., Vogiatzis, G., Tola, E., Dahl, A.B., 2016. Large-scale data for multiple-view stereopsis. Int. J. Comput. Vis. 120, 153–168. https://api.semanticscholar.org/CorpusID:14450404.

Barron, J.T., Mildenhall, B., Verbin, D., Srinivasan, P.P., Hedman, P., 2021. Mip-nerf 360: unbounded anti-aliased neural radiance fields. In: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5460–5469. https://api.semanticscholar.org/CorpusID:244488448.

Chen, Z., Chen, S., Schmid, C., Laptev, I., 2023. gsdf: geometry-driven signed distance functions for 3d hand-object reconstruction. In: 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 12890–12900. https://api.semanticscholar.org/CorpusID:258298107.

Dai, P., Xu, J., Xie, W., Liu, X., Wang, H., Xu, W., 2024. High-quality surface reconstruction using Gaussian surfels. In: International Conference on Computer Graphics and Interactive Techniques. https://api.semanticscholar.org/CorpusID:269449099.

Fan, L., Yang, Y., Li, M., Li, H., Zhang, Z., 2024. Trim 3d Gaussian splatting for accurate geometry representation. arXiv:2406.07499 [abs]. https://api.semanticscholar.org/CorpusID:270379916, 2024.

Fan, Z., Wang, K., Wen, K., Zhu, Z., Xu, D., Wang, Z., 2023. Lightgaussian: unbounded 3d Gaussian compression with 15x reduction and 200+ fps. arXiv:2311.17245 [abs]. https://api.semanticscholar.org/CorpusID:265498655, 2023.

Gu'edon, A., Lepetit, V., 2023. Sugar: surface-aligned Gaussian splatting for efficient 3d mesh reconstruction and high-quality mesh rendering. In: 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5354–5363. https://api.semanticscholar.org/CorpusID:265308825.

Hu, L., Zhang, H., Zhang, Y., Zhou, B., Liu, B., Zhang, S., Nie, L., 2023. Gaussianavatar: towards realistic human avatar modeling from a single video via animatable 3d Gaussians. In: 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 634–644. https://api.semanticscholar.org/CorpusID:265608952.

Huang, B., Yu, Z., Chen, A., Geiger, A., Gao, S., 2024. 2d Gaussian splatting for geometrically accurate radiance fields. arXiv:2403.17888 [abs]. https://api.semanticscholar.org/CorpusID:268692234, 2024.

Jiang, C.M., Sud, A., Makadia, A., Huang, J., Nießner, M., Funkhouser, T.A., 2020. Local implicit grid representations for 3d scenes. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 6000–6009. https://api.semanticscholar.org/CorpusID:214606025.

Kazhdan, M.M., Hoppe, H., 2013. Screened Poisson surface reconstruction. ACM Trans. Graph. 32, 29:1–29:13. https://api.semanticscholar.org/CorpusID:1371704.

Kerbl, B., Kopanas, G., Leimkuehler, T., Drettakis, G., 2023. 3d Gaussian splatting for real-time radiance field rendering. ACM Trans. Graph. 42, 1–14. https://api.semanticscholar.org/CorpusID:259267917.

Knapitsch, A., Park, J., Zhou, Q.Y., Koltun, V., 2017. Tanks and temples. ACM Trans. Graph. 36, 1–13. https://api.semanticscholar.org/CorpusID:3362330.

Li, J., Zhang, J., Bai, X., Zheng, J., Ning, X., Zhou, J., Gu, L., 2024. Dngaussian: optimizing sparse-view 3d Gaussian radiance fields with global-local depth normalization. In: 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 20775–20785. https://api.semanticscholar.org/CorpusID:268363574.

Li, Z., Muller, T., Evans, A., Taylor, R.H., Unberath, M., Liu, M.Y., Lin, C.H., 2023. Neuralangelo: high-fidelity neural surface reconstruction. In: 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 8456–8465. https://api.semanticscholar.org/CorpusID:259075890.

Liu, R., Xu, R., Hu, Y., Chen, M., Feng, A.W.W., 2024. Atomgs: atomizing Gaussian splatting for high-fidelity radiance field. arXiv:2405.12369 [abs]. https://api.semanticscholar.org/CorpusID:269930042, 2024.

Lombardi, S., Simon, T., Saragih, J.M., Schwartz, G., Lehrmann, A.M., Sheikh, Y., 2019. Neural volumes. ACM Trans. Graph. 38, 1–14. https://api.semanticscholar.org/CorpusID:195068954.

Lorensen, W.E., Cline, H.E., 1987. Marching cubes: a high resolution 3d surface construction algorithm. In: Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques. https://api.semanticscholar.org/CorpusID:15545924.

Lu, T., Yu, M., Xu, L., Xiangli, Y., Wang, L., Lin, D., Dai, B., 2023. Scaffold-gs: structured 3d Gaussians for view-adaptive rendering. In: 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 20654–20664. https://api.semanticscholar.org/CorpusID:265551778.

Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R., 2020. Nerf. Commun. ACM 65, 99–106. https://api.semanticscholar.org/CorpusID:213175590.

Müller, T., Evans, A., Schied, C., Keller, A., 2022. Instant neural graphics primitives with a multiresolution hash encoding. ACM Trans. Graph. 41, 1–15. https://api.semanticscholar.org/CorpusID:246016186.

Papantonakis, P., Kopanas, G., Kerbl, B., Lanvin, A., Drettakis, G., 2024. Reducing the memory footprint of 3d Gaussian splatting. Proc. ACM Comput. Graph. Inter. Tech. 7, 1–17. https://api.semanticscholar.org/CorpusID:269764213.

Ran, Y., Zeng, J., He, S., Chen, J., Li, L., Chen, Y., Lee, G.H., Ye, Q., 2022. Neurar: neural uncertainty for autonomous 3d reconstruction with implicit neural representations. IEEE Robot. Autom. Lett. 8, 1125–1132. https://api.semanticscholar.org/CorpusID:255679020.

Ren, K., Jiang, L., Lu, T., Yu, M., Xu, L., Ni, Z., Dai, B., 2024. Octree-gs: towards consistent real-time rendering with lod-structured 3d Gaussians. arXiv:2403.17898. https://api.semanticscholar.org/CorpusID:268691629, 2024.

Sitzmann, V., Thies, J., Heide, F., Nießner, M., Wetzstein, G., Zollhöfer, M., 2018. Deepvoxels: learning persistent 3d feature embeddings. In: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2432–2441. https://api.semanticscholar.org/CorpusID:54444417.

Sun, C., Sun, M., Chen, H.T., 2021. Direct voxel grid optimization: super-fast convergence for radiance fields reconstruction. In: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5449–5459. https://api.semanticscholar.org/CorpusID:244477646.

Turkulainen, M., Ren, X., Melekhov, I., Seiskari, O., Rahtu, E., Kannala, J., 2024. Dn-splatter: depth and normal priors for Gaussian splatting and meshing. arXiv:2403.17822. https://api.semanticscholar.org/CorpusID:268691514, 2024.

Wang, P., Liu, L., Liu, Y., Theobalt, C., Komura, T., Wang, W., 2021. Neus: learning neural implicit surfaces by volume rendering for multi-view reconstruction. arXiv:2106.10689. https://api.semanticscholar.org/CorpusID:235490453, 2021.

Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P., 2004. Image quality assessment: from error visibility to structural similarity. IEEE Trans. Image Process. 13, 600–612. https://api.semanticscholar.org/CorpusID:207761262.

Wolf, Y., Bracha, A., Kimmel, R., 2024. Gs2mesh: surface reconstruction from Gaussian splatting via novel stereo views. https://arxiv.org/abs/2404.01810. arXiv:2404.01810, 2024.

Wu, G., Yi, T., Fang, J., Xie, L., Zhang, X., Wei, W., Liu, W., Tian, Q., Wang, X., 2023. 4d Gaussian splatting for real-time dynamic scene rendering. In: 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 20310–20320. https://api.semanticscholar.org/CorpusID:263908793.

Yang, H., Zhang, C., Wang, W., Volino, M., Hilton, A., Zhang, L., Zhu, X., 2024. Gaussian splatting with localized points management. arXiv:2406.04251. https://api.semanticscholar.org/CorpusID:270285818, 2024.

Yariv, L., Gu, J., Kasten, Y., Lipman, Y., 2021. Volume rendering of neural implicit surfaces. In: Neural Information Processing Systems. https://api.semanticscholar.org/CorpusID:235605960.

Yu, A., Fridovich-Keil, S., Tancik, M., Chen, Q., Recht, B., Kanazawa, A., 2021. Plenoxels: radiance fields without neural networks. In: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5491–5500. https://api.semanticscholar.org/CorpusID:245006364.

Yu, R., Huang, T., Ling, J., Xu, F., 2024a. 2dgh: 2d Gaussian-Hermite splatting for high-quality rendering and better geometry reconstruction. arXiv:2408.16982 [abs]. https://api.semanticscholar.org/CorpusID:272310438, 2024.

Yu, Z., Chen, A., Huang, B., Sattler, T., Geiger, A., 2023. Mip-splatting: alias-free 3d Gaussian splatting. In: 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 19447–19456. https://api.semanticscholar.org/CorpusID:265466192.

Yu, Z., Sattler, T., Geiger, A., 2024b. Gaussian opacity fields: efficient adaptive surface reconstruction in unbounded scenes. ACM Trans. Graph. https://api.semanticscholar.org/CorpusID:272592871.

Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O., 2018. The unreasonable effectiveness of deep features as a perceptual metric. In: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 586–595. https://api.semanticscholar.org/CorpusID:4766599.

Zhang, W., Li, Z.H., Ma, D., Tang, H., Jiang, X., Zheng, Q., Pan, G., 2024a. Spiking gs: towards high-accuracy and low-cost surface reconstruction via spiking neuron-based Gaussian splatting. arXiv:2410.07266. https://api.semanticscholar.org/CorpusID:273233499, 2024.

Zhang, Z., Hu, W., Lao, Y., He, T., Zhao, H., 2024b. Pixel-gs: density control with pixel-aware gradient for 3d Gaussian splatting. arXiv:2403.15530 [abs]. https://api.semanticscholar.org/CorpusID:268681766, 2024.

Zhu, Z., Fan, Z., Jiang, Y., Wang, Z., 2023. Fsgs: real-time few-shot view synthesis using Gaussian splatting. In: European Conference on Computer Vision. https://api.semanticscholar.org/CorpusID:265552031.