

丘成桐中学科学奖（计算机奖）

研究报告

参赛学生姓名： 吴泽荟

中学： 武汉康礼高级中学

省份： 湖北省

国家/地区： 中国

指导老师姓名： 王芳

指导老师单位： 湖北第二师范学院

论文题目： Semantic-Guided Density Control for

Compact Semantic 3D Gaussian Splatting

Contents

Abstract	1
1 Introduction	2
2 Related Work	4
2.1 3D Gaussian Splatting	4
2.2 Semantic 3D Gaussian Representation	5
2.3 Compact 3D Gaussian Representation	6
2.4 Positioning Against Existing Methods	7
3 Method	8
3.1 Preliminaries	8
3.2 Problem Formulation	9
3.3 Semantic-Guided Density Control	10
3.3.1 Confidence-Guided Semantic Labeling	10
3.3.2 Semantic Adaptive Splitting with Annealing	11
3.3.3 Gradient-Aware Pruning	12
4 Experiments	13
4.1 Experimental Setup	13
4.2 Quantitative Results	15
4.3 Qualitative Results	17
4.4 Ablation Study	20
4.4.1 Component Ablation	20
4.4.2 Sensitivity Analysis	21
4.5 Discussion	24
5 Conclusion and Limitations	26
5.1 Conclusion	26
5.2 Limitations and Future Work	26
References	27
Appendix: Response to Reviewers	33
Acknowledgements, Declarations and Contributions	39

Semantic-Guided Density Control for Compact Semantic 3D Gaussian Splatting

Zehui Wu^{1,*}, Fang Wang²

¹ Cogdel Cranleigh High School Wuhan, Wuhan, China

² School of Computer Science and Artificial Intelligence, Hubei University of Education, Wuhan, China

Abstract: 3D Gaussian Splatting (3DGS) enables real-time novel view synthesis with an explicit scene representation, and its semantic extensions have attracted increasing attention for open-vocabulary 3D understanding. However, most semantic 3DGS pipelines inherit the photometric density control of vanilla 3DGS, which is agnostic to semantics and tends to over-densify primitives in semantically ambiguous regions and along object boundaries, resulting in redundant representations and excessive memory consumption. This paper presents a semantic-guided density control framework for compact semantic 3DGS. The framework operates on a generic representation unit (e.g., Gaussians or anchors) and consists of three components: confidence-guided semantic labelling, which partitions units into reliable and uncertain ones using semantic confidence; semantic adaptive splitting with annealing, which applies differentiated densification criteria to the two groups; and gradient-aware pruning, which removes low-contribution units using instance-level gradient statistics together with an absolute gradient constraint. The formulation is backbone-agnostic and is instantiated on both a primitive-based backbone (Gaussian Grouping) and an anchor-based backbone (ObjectGS), where density control is lifted from individual Gaussians to anchor units. Experiments on LERF-Mask, Replica, and Mip-NeRF 360 show that the proposed approach reduces the number of representation units by 33.5-61.0% while preserving rendering quality and improving or maintaining semantic accuracy. The method also reduces memory and peak GPU usage and improves rendering speed. These results indicate that semantic cues can serve not only as high-level descriptors but also as effective optimization signals for controlling the structural complexity of explicit 3D representations.

Keywords: 3D Gaussian Splatting, Semantic-guided Density Control, Compact Scene Representation, Instance Segmentation

1. Introduction

In recent years, 3D Gaussian Splatting [1] (3DGS) has revolutionized the paradigm of novel view synthesis and 3D reconstruction, owing to its exceptional real-time rendering frame rates and explicit 3D scene representation capabilities. Beyond fundamental geometric and appearance modelling, researchers have increasingly focused on endowing 3DGS with high-level semantic understanding of 3D scenes. Representative recent works, such as Feature 3DGS [2], Gaussian Grouping [3], LangSplat [4], and ObjectGS [5], have successfully achieved open-vocabulary querying, 3D instance segmentation, and scene editing by associating Gaussian representations with general feature fields, identity encodings, CLIP language features [6], or object-level descriptors. These frameworks span both primitive-based designs, where each Gaussian is optimised independently, and anchor-based designs, where a sparse set of anchors decodes neural Gaussians. Although these semantic 3DGS frameworks have significantly expanded the interactive capabilities of 3D scenes, they suffer from poor representational compactness.

The root cause of this inefficiency lies in the adaptive density control inherited from vanilla 3DGS. Its densification mechanisms, specifically cloning and splitting, rely primarily on view-space positional gradients. This geometry-dominated growth strategy is semantically agnostic, which tends to introduce redundant primitives at semantic boundaries or within complex regions. Consequently, mainstream semantic 3DGS pipelines often retain an excessive number of redundant units, resulting in bloated models and significant GPU memory overhead [7, 8]. Moreover, anchor-based representations such as Scaffold-GS [9] and ObjectGS [5] reduce storage by decoding neural Gaussians from a sparse set of anchors, yet their density control is still driven by geometric or appearance-based criteria, so semantically ambiguous regions may remain indistinguishable from reliable ones. Both primitive-based and anchor-based semantic pipelines therefore lack a mechanism that allows semantic confidence to directly regulate density evolution.

To address this bottleneck, we propose a lightweight and modular semantic-guided density control (SGDC) framework. Unlike many existing compression methods that trade off image quality or introduce complex neural decoders, SGDC achieves a compact and high-quality semantic 3DGS representation through three synergistic mechanisms. First, **confidence-guided semantic labelling** partitions representation units into reliable and uncertain ones, effectively suppressing erroneous densification in ambiguous regions.

Second, **semantic adaptive splitting with annealing** applies differentiated densification criteria to the two groups, encouraging detail preservation in semantically meaningful regions while suppressing redundant growth elsewhere. Third, **gradient-aware pruning** removes low-contribution units by combining instance-level statistics with absolute gradient constraints. Relying only on per-unit gradient statistics and semantic confidence, the formulation is backbone-agnostic and can be instantiated on existing semantic 3DGS pipelines with explicit adaptation points for obtaining semantic confidence and integrating with existing densification criteria. We instantiate SGDC on both a primitive-based backbone (Gaussian Grouping [3]) and an anchor-based backbone (ObjectGS [5]). Experimental results demonstrate that SGDC reduces the number of representation units by 33.5–61.0% while maintaining comparable rendering quality and improving or preserving semantic accuracy. It also reduces memory and peak GPU usage and improves rendering speed on both backbones.

In summary, our main contributions are as follows:

- We propose a semantic-guided density control framework for compact semantic 3DGS. By explicitly leveraging semantic confidence as an optimisation signal to guide representation-unit evolution, the framework mitigates the semantic agnosticism of conventional geometry-driven density control, and demonstrates that high-level semantic cues can effectively regulate the structural complexity of explicit 3D representations.
- We design three synergistic components that dynamically regulate representation units: confidence-guided semantic labelling for isolating reliable units, semantic adaptive splitting with annealing for applying differentiated densification criteria, and gradient-aware pruning for eliminating low-contribution units. Together, these components yield compact scene representations with reduced redundancy.
- We validate the framework on two representative semantic 3DGS backbones—the primitive-based Gaussian Grouping and the anchor-based ObjectGS—showing that the same formulation can be instantiated with different confidence sources and density-control mechanisms. Across LERF-Mask, Replica, and Mip-NeRF 360, SGDC reduces the number of representation units by 33.5–61.0% while maintaining comparable rendering quality and preserving or improving semantic accuracy.

2. Related Work

2.1. 3D Gaussian Splatting

3DGS [1] achieves rendering quality comparable to NeRF [10] while offering significantly faster rendering. Current research on 3DGS spans several directions: reconstruction quality [11, 9, 12, 13], reconstruction and rendering speed [14, 15], compact representations [16, 17, 18], and the integration of semantic understanding [3, 4, 19, 5, 20, 21]. Benefiting from its explicit Gaussian representation, 3DGS is particularly well-suited for carrying semantic information. Our method focuses on compact semantic 3DGS, aiming to reduce the prevalent redundancy of representation units in existing semantic methods.

2.2. Semantic 3D Gaussian Representation

Following the success of 3DGS in novel view synthesis, endowing it with scene understanding capabilities has become a prominent research hotspot. Gaussian Grouping [3] introduces a learnable 16-dimensional identity encoding for each Gaussian primitive, enabling joint reconstruction and segmentation of 3D scenes. Other works explore semantic 3DGS from different angles, including joint appearance-semantic learning [19], feature splatting [22], object-level partitioning [23], cross-view context aggregation [24], and domain-specific applications [25, 26]. However, the integration of semantic information often increases representation complexity, leading to an expansion in the number of representation units and elevated storage overhead. ObjectGS [5] adopts an object-aware representation built on anchors, where each anchor decodes neural Gaussians and object-level descriptors are learned jointly for reconstruction and scene understanding. This anchor-based organisation improves representational efficiency, but its density control remains geometry- and appearance-driven, so semantic confidence does not participate in anchor growth or pruning. Our work is complementary: we inject semantic confidence into the density-control stage and validate the same formulation on both primitive-based and anchor-based backbones.

2.3. Compact 3D Gaussian Representation

To reduce the storage and computational overhead of 3DGS, existing methods can be grouped into density-control optimisation, pruning, and parameter compression. Density-control optimisation modifies the splitting and

pruning criteria at their source, e.g., SteepGS [18] (about 50% fewer primitives), the cross-section-oriented splitting of EGU-GS [27], and the gradient-coherence criterion of GDAGS [28]. Pruning removes redundant primitives after creation, such as the magnitude-based pruning and distillation of LightGaussian [17], the optimised rasteriser and pruning of Speedy-Splat [15], and the learned stochastic masks of MaskGaussian [29]. Parameter compression keeps the primitive count unchanged and reduces per-primitive storage through attribute quantisation, e.g., the vector quantisation of Compact3D [16]. These methods operate solely on geometric or photometric signals such as gradients, opacity, and attribute statistics, and therefore cannot distinguish semantically critical primitives from photometrically redundant ones. In contrast, our method injects semantic confidence into density control, achieving compactness while preserving semantic accuracy.

2.4. Positioning Against Existing Methods

Table 1 compares representative compact and semantic 3DGS methods. Existing compact methods are semantically agnostic, while semantic 3DGS methods inherit dense primitive distributions and geometry-driven density control. Our SGDC uses semantic confidence to guide growth and pruning during training, and is instantiated on both primitive-based and anchor-based backbones.

Table 1: Positioning of our framework against representative compact and semantic 3DGS methods.

Method	Unit	Decision signal	Density-control stage	Cost
<i>Non-semantic compact 3DGS</i>				
LightGaussian [17]	Gaussian	Global significance score	Post-training pruning	Low
SteepGS [18]	Gaussian	Steepest descent direction	Training-time densification	Medium
EGU-GS [27]	Gaussian	Gradient statistics	Training-time densification	Medium
<i>Semantic 3DGS</i>				
Feature 3DGS [2]	Gaussian	View-space gradient	Vanilla density control	High
Gaussian Grouping [3]	Gaussian	View-space gradient	Vanilla density control	High
LangSplat [4]	Gaussian	View-space gradient	Vanilla density control	High
ObjectGS [5]	Anchor	Anchor gradient and opacity	Anchor-level growth/pruning	High
Ours (SGDC)	Gaussian or Anchor	Semantic confidence and gradients	Training-time evolution	High

Cost denotes the relative overall training and inference cost of the pipeline.

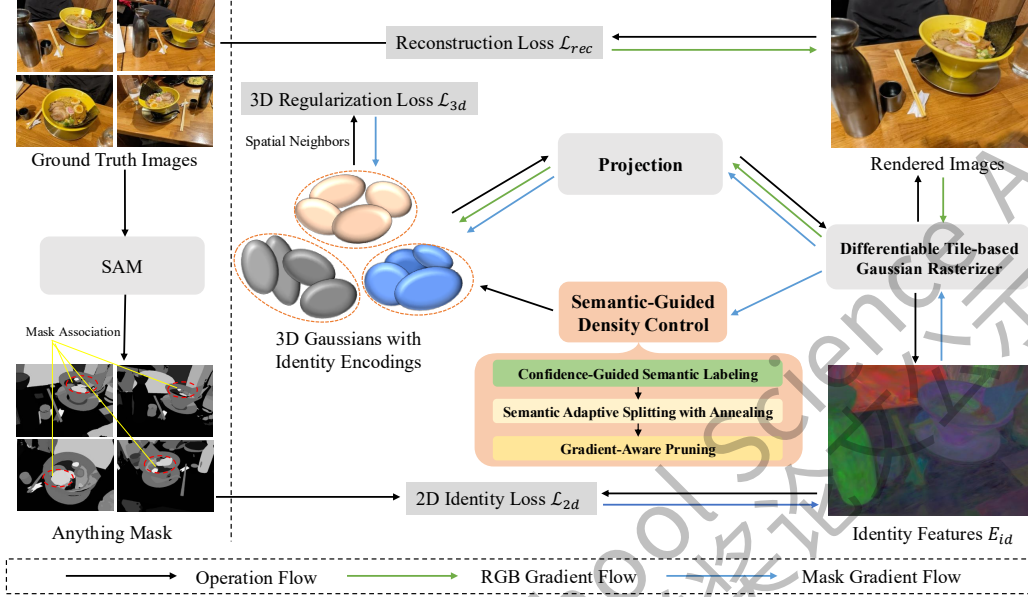


Figure 1: **Pipeline of the proposed semantic-guided density control (SGDC) framework.** The framework is illustrated with Gaussian Grouping as a representative primitive-based backbone. It first performs confidence-guided semantic labelling, which partitions representation units into reliable and uncertain groups. It then applies asymmetric per-instance quantile splitting with annealing, using a low threshold for reliable units to encourage detail densification and a high threshold for uncertain units to suppress redundant growth. Finally, gradient-aware pruning removes low-contribution units. The same formulation is instantiated on an anchor-based backbone (ObjectGS) by applying these three modules to anchor units.

3. Method

In this section, we present a semantic-guided density control (SGDC) framework for compact semantic 3DGS. We first review the vanilla 3DGS representation and two representative semantic extensions: the primitive-based Gaussian Grouping and the anchor-based ObjectGS. We then formulate SGDC on a shared abstraction in which density control operates on generic representation units, each associated with a semantic confidence score and an accumulated gradient. Figure 1 illustrates the formulation using Gaussian Grouping as a representative primitive-based backbone; the same three modules are instantiated on the anchor-based ObjectGS backbone by applying them to anchor units instead of individual Gaussians.

3.1. Preliminaries

3D Gaussian Splatting. 3D Gaussian Splatting (3DGS) [1] represents a 3D scene using a set of learnable Gaussian functions $\{G_i\}$. Each Gaussian G_i is defined by a set of parameters, and the i -th 3D Gaussian primitive can be characterized as $G_i = \{\boldsymbol{\mu}_i, s_i, q_i, \alpha_i, \mathbf{h}_i\}$. Specifically, $\boldsymbol{\mu}_i \in \mathbb{R}^3$ denotes the centre position; Σ_i represents the covariance matrix, which is constructed from a scaling factor $s_i \in \mathbb{R}^3$ and a rotation quaternion $q_i \in \mathbb{R}^4$ to control the local geometric shape; $\alpha_i \in \mathbb{R}$ is the opacity; and $\mathbf{h}_i \in \mathbb{R}^{27}$ is a set of spherical harmonics (SH) coefficients used to model the view-dependent appearance colour vector. After depth sorting, these 3D Gaussians are projected via differentiable splatting and rendered into 2D images based on tile-based alpha-blending. The final pixel colour at position x on the 2D image can be expressed as:

$$C(x) = \sum_{i=1}^N h_i(x) \alpha_i(x) \prod_{j=1}^{i-1} (1 - \alpha_j(x)) \quad (1)$$

where $C(x)$ represents the synthesised colour at pixel coordinate x , and N is the total number of 3D Gaussians that cover the pixel after depth sorting. $h_i(x)$ corresponds to the view-dependent colour of the i -th Gaussian at x , and $\alpha_i(x)$ is its opacity weight, where $\alpha_i(x) \in [0, 1]$ is computed via differentiable splatting. The product term $\prod_{j=1}^{i-1} (1 - \alpha_j(x))$ represents the accumulated transmittance of all Gaussians ordered before the i -th Gaussian.

Gaussian Grouping. Gaussian Grouping [3] is a significant recent extension of 3DGS, designed to inject instance-level semantic structures into the explicit 3D Gaussian representation. This enables open-world 3D scene segmentation and editing capabilities. The core idea is to attach a learnable 16-dimensional identity embedding $e_i \in \mathbb{R}^{16}$ to each 3D Gaussian G_i , while preserving its original geometric and appearance attributes (position $\boldsymbol{\mu}_i$, covariance matrix Σ_i , opacity α_i , and SH coefficients $\mathbf{h}_i$). Through training, Gaussians belonging to the same object or component are encouraged to possess similar embeddings, thereby implicitly achieving 3D instance grouping. To attain this goal, 2D instance segmentation masks are first generated for multi-view training images using the Segment Anything Model (SAM) [30] to serve as supervision signals. Simultaneously, the randomly initialized identity embeddings e_i of each 3D Gaussian are projected onto the image plane via differentiable rendering, forming a rendered 2D identity feature map E_{id} . The model optimization is performed through a joint loss function, which

adds a 2D identity loss $\mathcal{L}_{2d}$ and a 3D regularization loss $\mathcal{L}_{3d}$ to the photometric reconstruction loss $\mathcal{L}_{rec}$ of the vanilla 3DGS. The end-to-end loss function is formulated as follows:

$$\mathcal{L} = \mathcal{L}_{rec} + \lambda_{2d}\mathcal{L}_{2d} + \lambda_{3d}\mathcal{L}_{3d} \quad (2)$$

Where $\mathcal{L}_{2d}$ is the 2D identity loss, implemented as a standard cross-entropy loss to supervise the rendered 2D identity feature map E_{id} to match the 2D mask labels provided by SAM. $\mathcal{L}_{3d}$ is an unsupervised loss based on Kullback-Leibler (KL) divergence, which enforces similar identity embedding distributions among 3D Gaussians in close spatial proximity. This 3D regularization compensates for the potential deficiency of 2D supervision signals in occluded or internal regions of objects, thereby ensuring the spatial continuity and consistency of the grouping results in 3D space. λ_{2d} and λ_{3d} are hyperparameters used to balance the weights of these two loss terms.

Anchor-based semantic 3DGS. In addition to primitive-based representations, we consider anchor-based semantic pipelines such as ObjectGS [5]. Here the scene is organised around a sparse set of persistent anchors. Each anchor carries a learnable feature and a position, and an MLP decodes a fixed number of neural Gaussians per anchor, each with its own offset, scale, rotation, opacity, and colour. Unlike Gaussian Grouping, where every Gaussian primitive is an independent optimisation target, density control in anchor-based pipelines operates on the anchor set: new anchors are grown where the accumulated anchor gradient is large, and redundant anchors are removed together with the Gaussians they decode. This distinction motivates the representation-unit abstraction used in our formulation.

3.2. Problem Formulation

We formalise semantic-guided density control as a general optimisation over a set of representation units. Let $\mathcal{U} = \{u_i\}_{i=1}^N$ denote these units, the persistent entities governed by density control. For a primitive-based backbone such as Gaussian Grouping [3], each unit u_i is an individual Gaussian primitive parameterised by its centre position $\boldsymbol{\mu}_i \in \mathbb{R}^3$, covariance $\boldsymbol{\Sigma}_i$, opacity α_i , and spherical-harmonics coefficients $\mathbf{s}_i \in \mathbb{R}^{27}$. For an anchor-based backbone such as ObjectGS [5], each unit u_i is an anchor with centre position $\boldsymbol{\mu}_i$ and a learnable feature $\mathbf{f}_i$, from which an MLP decodes K neural Gaussians. In both cases, the unit centre $\boldsymbol{\mu}_i$ is the persistent spatial parameter that density control acts upon.

During training, every unit is associated with a semantic confidence score $c_i \in [0, 1]$ and an accumulated gradient magnitude

$$\nabla_i = \left\| \frac{\partial \mathcal{L}}{\partial \mu_i} \right\|_2, \quad (3)$$

where $\mathcal{L}$ denotes the total training objective. The confidence c_i is provided by the backbone: for Gaussian Grouping it is the maximum softmax probability of the rendered identity encoding, whereas for ObjectGS it is derived from the object-association probability of the anchor.

Gradient source. $\mathcal{L}$ combines the photometric reconstruction loss $\mathcal{L}_{rec}$ with the semantic supervision used by the backbone (e.g., the identity-encoding cross-entropy and consistency terms $\mathcal{L}_{2d} + \mathcal{L}_{3d}$ in Gaussian Grouping). The accumulated gradient ∇_i therefore reflects how each unit contributes to both photometric reconstruction and semantic prediction, and both the splitting and pruning criteria below operate on this same signal. This ensures that units which are photometrically redundant but semantically critical are not removed, in contrast to vanilla 3DGS, where densification is driven by the reconstruction loss alone.

Inputs. (i) Calibrated training views with known camera poses; (ii) an initial set of representation units; (iii) per-pixel semantic labels and confidence logits from the semantic backbone (e.g., 2D instance segmentation or feature distillation); (iv) per-unit gradients accumulated across views, computed with respect to the total loss.

Assumptions. The scene is static and observed under Lambertian or moderately specular materials. Semantic supervision is provided at the instance level and may be noisy or incomplete in reflective, transparent, or occluded regions. The density controller treats the rasteriser and the semantic feature decoder as black-box components and makes no architectural assumption beyond the existence of a persistent unit with a centre position, a confidence score, and an accumulated gradient.

Objective. We seek a compact subset $\mathcal{U}^* \subseteq \mathcal{U}$ such that:

- (1) the photometric reconstruction loss $\mathcal{L}_{rec}$ and the semantic supervision losses remain within a small tolerance ϵ of the full-density baseline;
- (2) the cardinality $|\mathcal{U}^*|$ is minimised;
- (3) the spatial distribution of $\mathcal{U}^*$ preserves semantic boundaries and suppresses redundant units in homogeneous regions.

This formulation is backbone-agnostic: it applies to any semantic 3DGS pipeline that exposes a persistent representation unit together with a semantic confidence score and an accumulated gradient, with explicit adaptation points for the confidence source and the densification mechanism.

3.3. Semantic-Guided Density Control

The density control mechanisms of vanilla 3DGS [1] and its semantic extensions such as Gaussian Grouping [3] and ObjectGS [5] rely primarily on globally static thresholds for view-space positional gradients. Such geometry- and appearance-driven strategies treat semantically critical regions and photometrically homogeneous regions alike, and therefore fail to produce a semantically optimal distribution of representation units. To address this, we propose a backbone-agnostic semantic-guided density control (SGDC) formulation that injects semantic confidence directly into the density-control process. SGDC decouples the semantic attributes of a unit from its physical convergence state, and regulates the scene distribution through three components: confidence-guided semantic labelling, semantic adaptive splitting with annealing, and gradient-aware pruning. The formulation is defined on generic representation units and is instantiated on both a primitive-based backbone (Gaussian Grouping) and an anchor-based backbone (ObjectGS).

3.3.1. Confidence-Guided Semantic Labeling

We first establish a semantic filtering mechanism that prevents uncertain predictions from contaminating structural growth. The mechanism is defined on representation units: each unit u_i is associated with a semantic confidence score $c_i \in [0, 1]$ provided by the semantic backbone. For a primitive-based backbone such as Gaussian Grouping, the confidence is computed from the 16-dimensional identity embedding $e_i \in \mathbb{R}^{16}$. Concretely, a lightweight classifier (a 1×1 convolutional layer) maps the embedding to a semantic logit vector $L_i \in \mathbb{R}^K$, where K is the number of semantic categories including background, and c_i is the maximum softmax probability:

$$c_i = \max_k \frac{\exp(L_{i,k})}{\sum_{j=1}^K \exp(L_{i,j})} \quad (4)$$

For an anchor-based backbone such as ObjectGS, the semantic confidence is instead derived from the object-association probability of the anchor. Both definitions are normalised to $[0, 1]$, and the same threshold $\tau_c = 0.6$ is found to be effective for both confidence sources.

Units with confidence scores below τ_c typically correspond to boundary regions, occluded parts, or stochastic background noise. To prevent semantic bleeding during densification, a phenomenon where low-quality predictions mislead density control and cause cross-contamination between distinct semantic regions, we partition the units into reliable units and an uncertain set. Specifically, the semantic label y_i assigned to each unit is defined as:

$$y_i = \begin{cases} \hat{y}_i, & \text{if } c_i \geq \tau_c \\ -1, & \text{if } c_i < \tau_c \end{cases} \quad (5)$$

where $\hat{y}_i$ is the semantic label predicted by the backbone (for Gaussian Grouping, $\hat{y}_i = \text{argmax}(L_i)$). The value -1 marks uncertain units with low confidence. This separation ensures that splitting and pruning budgets are allocated according to reliable semantic priors rather than being spent on ambiguous regions.

3.3.2. Semantic Adaptive Splitting with Annealing

Traditional global threshold strategies are semantically blind: flat regions such as walls and floors share the same splitting criterion as detail-rich regions such as object boundaries and textured zones. Flat regions typically possess lower gradients and homogeneous semantics, where excessive splitting merely increases redundancy, whereas detail-rich regions require dense units to capture fine-grained geometric and semantic details. A single global threshold forces the two to compete for the same splitting budget, resulting either in insufficient densification of detailed regions or in an accumulation of redundant units in flat regions.

To address this issue, we introduce an asymmetric per-group quantile splitting strategy. Recall that ∇_i denotes the accumulated gradient magnitude of unit u_i and y_i its semantic label, with $y_i = -1$ for the uncertain set. For a label group g , the splitting threshold is defined as the q -th quantile of the gradients in that group:

$$\mathcal{T}_{split}^{(g)} = \begin{cases} Q(\{\nabla_j \mid y_j = g\}, q_{fg}), & \text{if } g \neq -1 \\ Q(\{\nabla_j \mid y_j = -1\}, q_{bg}), & \text{if } g = -1 \end{cases} \quad (6)$$

In this expression, $Q(X, q)$ denotes the q -th quantile of set X , where q_{fg} and q_{bg} are quantile hyperparameters designated for reliable groups and the uncertain set, respectively. By imposing a substantially more stringent

threshold for the uncertain set, SGDC suppresses the excessive growth of units within noisy regions and prioritises the splitting budget toward reliable semantic groups, thereby fostering sharp and complete semantic boundaries.

The criterion above is defined on the generic representation unit. On a primitive-based backbone, splitting follows the vanilla 3DGS rule and creates new Gaussian primitives. On an anchor-based backbone, the same criterion is applied to anchor growth: new anchors are added when the effective threshold is exceeded, and the number of neural Gaussians decoded per anchor can optionally be modulated by the same confidence-conditioned criterion.

Furthermore, we implement a linear annealing schedule for the splitting threshold to stabilise the early stages of training by avoiding aggressive splitting caused by gradient noise, and to facilitate sufficient densification of detail-rich areas during later stages. The effective threshold $\tilde{\mathcal{T}}_{split}(t)$ at iteration t undergoes a smooth decay over time:

$$\tilde{\mathcal{T}}_{split}(t) = \mathcal{T}_{split} \cdot \left(1 - (1 - \rho) \cdot \text{clip} \left(\frac{t - t_{start}}{t_{end} - t_{start}}, 0, 1 \right) \right) \quad (7)$$

In this formulation, ρ represents the target decay factor, while t_{start} and t_{end} define the starting and ending iterations of the annealing window. This scheduling strategy encourages the model to prioritise the reconstruction of primary structures with large gradients during the early stages of training and subsequently refine low-gradient geometric details in the later stages.

3.3.3. Gradient-Aware Pruning

Vanilla 3DGS relies solely on low opacity ($\alpha < 0.005$) and large screen-space radii to cull primitives, which neglects redundant primitives whose gradients have stagnated over long optimisation periods. Semantic 3DGS pipelines, whether primitive-based or anchor-based, inherit similar geometry- and appearance-driven criteria, so redundant units persist even when they no longer contribute to reconstruction or semantic prediction. To address this limitation, we introduce a gradient-aware pruning mechanism defined on representation units.

In addition to preserving the original opacity and spatial pruning rules, we implement a dual-verification criterion. A unit is flagged for pruning if its gradient falls within the bottom q_{prune} quantile of its corresponding semantic group and its absolute gradient remains below a stringent lower bound τ_{prune_abs} :

$$\text{PruneMask}_i = (\nabla_i < Q(\{\nabla_j \mid y_j = y_i\}, q_{\text{prune}})) \wedge (\nabla_i < \tau_{\text{prune_abs}}) \quad (8)$$

Here q_{prune} is the pruning quantile and $\tau_{\text{prune_abs}}$ is the absolute gradient lower bound. The dual-condition design serves two complementary purposes: the quantile term localises redundant units within an individual semantic group, while the absolute bound acts as a safety anchor that prevents erroneous deletions in inherently flat geometric regions such as walls or the sky, where gradients are naturally minimal. This avoids representation voids.

The gradient used in Eq. 8 depends on the backbone. On a primitive-based backbone, ∇_i is the accumulated view-space positional gradient of the Gaussian primitive, as in vanilla 3DGS. On an anchor-based backbone, we use the accumulated gradient of the anchor’s own centre position rather than aggregating the gradients of its decoded Gaussians. This choice is deliberate: the decoded Gaussians are regenerated by the MLP at every iteration and are not persistent optimisation targets, whereas the anchor position is. Aggregating sub-Gaussian gradients, for example by averaging, can mask isolated high-contribution Gaussians near object boundaries and cause the entire anchor to be removed; the anchor gradient instead reflects the overall effect of the anchor on both the reconstruction and semantic losses. Since the magnitude of anchor gradients differs from that of per-primitive gradients, $\tau_{\text{prune_abs}}$ is recalibrated per backbone, while the scale-invariant quantile q_{prune} is kept unchanged.

The complete training loop integrating the three components is summarised in Algorithm 1. The algorithm operates on the set of representation units $\mathcal{U}$, with hyperparameters including the semantic update interval T_{sem} , the densification interval T_{den} , the confidence threshold τ_c , the splitting quantiles q_{fg} and q_{bg} , the pruning quantile q_{prune} , and the absolute gradient threshold $\tau_{\text{prune_abs}}$. For primitive-based backbones, a unit is an individual Gaussian primitive; for anchor-based backbones, a unit is an anchor, and pruning removes the neural Gaussians it decodes.

4. Experiments

4.1. Experimental Setup

Implementation Details. We implement SGDC on two semantic 3DGS backbones: the primitive-based Gaussian Grouping [3] and the anchor-based

Algorithm 1 Training Loop of the Proposed Framework (Single Iteration)

Require: Set of representation units $\mathcal{U}$, current iteration t

Ensure: Updated set of representation units $\mathcal{U}'$

```
1: // 1. Forward pass (rendering)
2: Render RGB and semantic outputs from  $\mathcal{U}$ 
3: Compute total loss  $\mathcal{L} = \mathcal{L}_{rec} + \lambda_{2d} \mathcal{L}_{2d} + \lambda_{3d} \mathcal{L}_{3d}$ 
4: Backpropagate to obtain per-unit gradients  $\nabla_i$ 
5: // 2. Confidence computation and density control
6: if  $t \in [t_{start}, t_{end}]$  and  $t \bmod T_{den} = 0$  then
7:   // 2a. Update semantic labels
8:   for each unit  $i \in \mathcal{U}$  do
9:     Obtain the semantic prediction and confidence  $c_i$  from the back-
       bone  $\triangleright$  Primitive-based:  $c_i = \max_k \text{softmax}(\mathbf{L}_i)_k$ ; anchor-based:  $c_i$  is the
       object-association probability
10:    Assign label  $y_i \leftarrow \hat{y}_i$  if  $c_i \geq \tau_c$ , else  $-1$ 
11:  end for
12:  // 2b. Semantic adaptive splitting with annealing
13:  Compute effective threshold  $\tilde{T}_{split}(t)$  via Eq. 7
14:  for each semantic group  $g$  do
15:     $q \leftarrow q_{fg}$  if  $g \neq -1$ , else  $q_{bg}$ 
16:    Compute  $Q_{split} \leftarrow Q(\{\nabla_j \mid y_j = g\}, q)$ 
17:    for each unit  $i$  with  $y_i = g$  and  $\nabla_i > Q_{split}$  do
18:      Split/grow unit  $i$  using  $\tilde{T}_{split}(t)$   $\triangleright$  Primitive-based: add
        Gaussian primitives; anchor-based: add anchors (optionally adjust  $K$ )
19:    end for
20:  end for
21:  // 2c. Gradient-aware pruning
22:  for each semantic group  $g$  do
23:    Compute  $Q_{prune} \leftarrow Q(\{\nabla_j \mid y_j = g\}, q_{prune})$ 
24:    for each unit  $i$  with  $y_i = g$  do
25:      if  $\nabla_i < Q_{prune}$  and  $\nabla_i < \tau_{prune\_abs}$  then
26:        Remove unit  $i$  from  $\mathcal{U}$   $\triangleright$  Anchor-based: remove the
        decoded Gaussians as well
27:      end if
28:    end for
29:  end for
30:  Remove units with  $\alpha < 0.005$  or excessive screen-space size
31: end if
32: return  $\mathcal{U}'$ 
```

ObjectGS [5]. Both models are trained for 30,000 iterations, and the density controller is activated between iterations 4,000 and 15,000 with a 500-iteration densification interval. Between iterations 5,000 and 15,000, a linear annealing schedule decays the splitting threshold to 50% of its initial value. The gradients used for splitting and pruning are computed from the total training objective (reconstruction plus semantic supervision), as defined in Section 3.2. For Gaussian Grouping we use a confidence threshold $\tau_c = 0.6$, splitting quantiles $q_{fg} = 0.5$ and $q_{bg} = 0.95$, a pruning quantile $q_{prune} = 0.06$, and an absolute gradient bound $\tau_{prune_abs} = 6 \times 10^{-5}$. For ObjectGS, which operates on anchors, we use the same confidence threshold $\tau_c = 0.6$, a splitting quantile of 0.6, a fixed split threshold of 9×10^{-7} , a pruning quantile $q_{prune} = 0.03$, and an absolute gradient bound $\tau_{prune_abs} = 1 \times 10^{-5}$. Quantile-based criteria and the confidence threshold are shared across backbones, while the absolute gradient bound is recalibrated because per-anchor and per-primitive gradient magnitudes differ. All experiments are conducted on an NVIDIA A100 GPU.

Datasets and Metrics. We evaluate SGDC on the LERF-Mask dataset [3], the Mip-NeRF 360 dataset [31], and the Replica dataset [32]. For LERF-Mask, we report rendering quality (PSNR, SSIM [33], LPIPS [34]) and open-vocabulary instance segmentation (mIoU, BmIoU) on the held-out test views. BmIoU follows the Boundary IoU definition of Cheng et al. [35], which restricts the IoU computation to a narrow band around the ground-truth instance boundary. Mip-NeRF 360 lacks pixel-level semantic annotations, so we report only rendering quality and compactness, evaluated at 1/4 image resolution. For Replica, we report instance-level IoU and Dice averaged over all object instances. FPS is measured under two protocols: RGB-only rendering for non-semantic methods and RGB plus segmentation rendering for semantic methods, as distinguished in the tables.

4.2. Quantitative Results

We evaluate SGDC on LERF-Mask, Mip-NeRF 360, and Replica, and compare against vanilla 3DGS [1], Reduced-3DGS [36], SteepGS [18], and two semantic 3DGS backbones: the primitive-based Gaussian Grouping [3] and the anchor-based ObjectGS [5]. SGDC is instantiated on both backbones, which allows us to examine whether the formulation generalises across different representation units. Across all settings, SGDC reduces the number of representation units by 33.5–61.0% relative to the corresponding backbone

Table 2: Quantitative comparison on the *ramen* scene of the LERF-Mask dataset (open-vocabulary segmentation). All metrics are computed on the 4 held-out test views.

Method	Rendering quality			Semantics		Compactness		Efficiency		
	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	mIoU $\uparrow$	BmIoU $\uparrow$	Units $\downarrow$	Mem $\downarrow$	Train $\downarrow$	FPS $\uparrow$	Peak GPU $\downarrow$
<i>Non-semantic 3DGS</i>										
3DGS [1]	31.42	0.9382	0.0930	–	–	619k	138	9.38	228.72 ^a	2.65
Reduced-3DGS [36]	31.97	0.9390	0.0927	–	–	758k	50	9.58	222.82 ^a	2.50
SteepGS [18]	31.51	0.9374	0.0961	–	–	527k	125	8.97	255.40 ^a	3.75
<i>Semantic 3DGS</i>										
Gaussian Grouping [3]	32.17	0.9426	0.1191	77.30	69.55	1272k	379	41.50	9.50 ^b	12.02
Gaussian Grouping+SGDC	32.28	0.9429	0.1186	77.35	69.27	830k	247	35.45	10.18^b	8.93
ObjectGS [5]	28.81	0.9095	0.1251	88.01	79.92	1434k	416	97.51	12.33 ^b	19.78
ObjectGS+SGDC	28.90	0.9092	0.1270	89.04	79.15	815k	235	96.45	14.30^b	11.71

PSNR in dB; Mem in MB; train time in minutes; peak GPU in GB. Units: Gaussians for 3DGS/Gaussian Grouping, anchors for ObjectGS. ^aFPS: RGB-only; ^bFPS: RGB+Seg.

Table 3: Quantitative comparison on the *figurines* scene of the LERF-Mask dataset (open-vocabulary segmentation). All metrics are computed on the 4 held-out test views.

Method	Rendering quality			Semantics		Compactness		Efficiency		
	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	mIoU $\uparrow$	BmIoU $\uparrow$	Units $\downarrow$	Mem $\downarrow$	Train $\downarrow$	FPS $\uparrow$	Peak GPU $\downarrow$
<i>Non-semantic 3DGS</i>										
3DGS [1]	25.38	0.8899	0.1626	–	–	1804k	427	13.96	179.91 ^a	6.33
Reduced-3DGS [36]	25.29	0.8903	0.1640	–	–	1955k	127	14.58	181.24 ^a	5.66
SteepGS [18]	25.65	0.8898	0.1666	–	–	1369k	324	12.13	218.25 ^a	7.87
<i>Semantic 3DGS</i>										
Gaussian Grouping [3]	25.89	0.8938	0.2104	71.29	69.17	3332k	1017	66.50	9.37 ^b	31.35
Gaussian Grouping+SGDC	26.94	0.8935	0.2094	78.25	74.99	2080k	619	47.40	9.84^b	19.64
ObjectGS [5]	23.78	0.8691	0.1881	83.64	81.15	2614k	758	161.75	6.85 ^b	54.60
ObjectGS+SGDC	23.80	0.8725	0.1828	87.19	85.05	1021k	296	151.14	10.53^b	20.84

PSNR in dB; Mem in MB; train time in minutes; peak GPU in GB. Units: Gaussians for 3DGS/Gaussian Grouping, anchors for ObjectGS. ^aFPS: RGB-only; ^bFPS: RGB+Seg.

while preserving rendering quality and maintaining or improving semantic accuracy. It also reduces memory and peak GPU usage, shortens training time, and improves FPS on both backbones. The three density-control modules themselves add less than 3% to the total training time.

LERF-Mask Dataset. This dataset provides ground-truth instance masks for three tabletop scenes. As shown in Tables 2–4, SGDC reduces the number of representation units by 34.7–43.5% relative to Gaussian Grouping and by 43.2–61.0% relative to ObjectGS. On Gaussian Grouping, mIoU improves by up to 6.96 points (*figurines*) with PSNR changes between +0.06 and +1.05 dB. On ObjectGS, mIoU improves by up to 3.55 points (*figurines*) and PSNR by up to +1.44 dB (*teatime*). Memory and peak GPU usage de-

Table 4: Quantitative comparison on the *teatime* scene of the LERF-Mask dataset (open-vocabulary segmentation). All metrics are computed on the 3 held-out test views.

Method	Rendering quality			Semantics		Compactness		Efficiency		
	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	mIoU $\uparrow$	BmIoU $\uparrow$	Units $\downarrow$	Mem $\downarrow$	Train $\downarrow$	FPS $\uparrow$	Peak GPU $\downarrow$
<i>Non-semantic 3DGS</i>										
3DGS [1]	27.37	0.9064	0.1820	–	–	1853k	439	14.21	162.65 ^a	5.06
Reduced-3DGS [36]	28.16	0.9068	0.1782	–	–	2272k	148	15.97	169.19 ^a	5.11
SteepGS [18]	28.12	0.9062	0.1814	–	–	1359k	322	12.02	215.41 ^a	6.88
<i>Semantic 3DGS</i>										
Gaussian Grouping [3]	28.51	0.8907	0.2421	70.61	70.13	3834k	1140	63.68	11.27 ^b	28.18
Gaussian Grouping+SGDC	28.57	0.8908	0.2419	73.54	69.87	2164k	644	48.42	12.00^b	19.53
ObjectGS [5]	24.35	0.8759	0.2262	79.37	77.01	3004k	871	132.30	6.49 ^b	50.47
ObjectGS+SGDC	25.79	0.8774	0.2241	79.38	77.29	1255k	363	125.75	10.04^b	21.69

PSNR in dB; Mem in MB; train time in minutes; peak GPU in GB. Units: Gaussians for 3DGS/Gaussian Grouping, anchors for ObjectGS. ^aFPS: RGB-only; ^bFPS: RGB+Seg.

crease, and FPS increases in every scene on both backbones. Figure 2 shows the primitive-count evolution on the *figurines* scene for Gaussian Grouping: the baseline grows to 3,332k primitives, whereas SGDC stabilises after activation at iteration 4,000 and finalises at 2,080k (a 37.6% reduction).

Mip-NeRF 360 Dataset. This dataset evaluates rendering quality and compactness in large-scale unbounded scenes. As shown in Table 5, SGDC reduces the number of representation units from 1,853k to 1,090k (−41.2%) on Gaussian Grouping and from 303k to 197k (−35.0%) on ObjectGS, while slightly improving PSNR, SSIM, and LPIPS on both backbones. Relative to 3DGS, Reduced-3DGS reduces memory by 67.8% but increases primitives by 16.2%, whereas SteepGS compresses primitives by 27.7% without semantic supervision.

Replica Dataset. On this indoor synthetic benchmark, SGDC reduces the number of representation units from 615k to 336k (−45.4%) on Gaussian Grouping and from 1,612k to 1,072k (−33.5%) on ObjectGS, while reducing memory and peak GPU usage and improving FPS on both backbones (Table 6). Rendering quality is preserved or slightly improved (+0.02 dB PSNR on Gaussian Grouping and +0.53 dB on ObjectGS). Instance-level mIoU remains comparable, changing by −0.02 points on Gaussian Grouping and −0.58 points on ObjectGS.

4.3. Qualitative Results

We further verify the advantages of SGDC in terms of rendering quality, semantic segmentation accuracy, and model compactness through visual

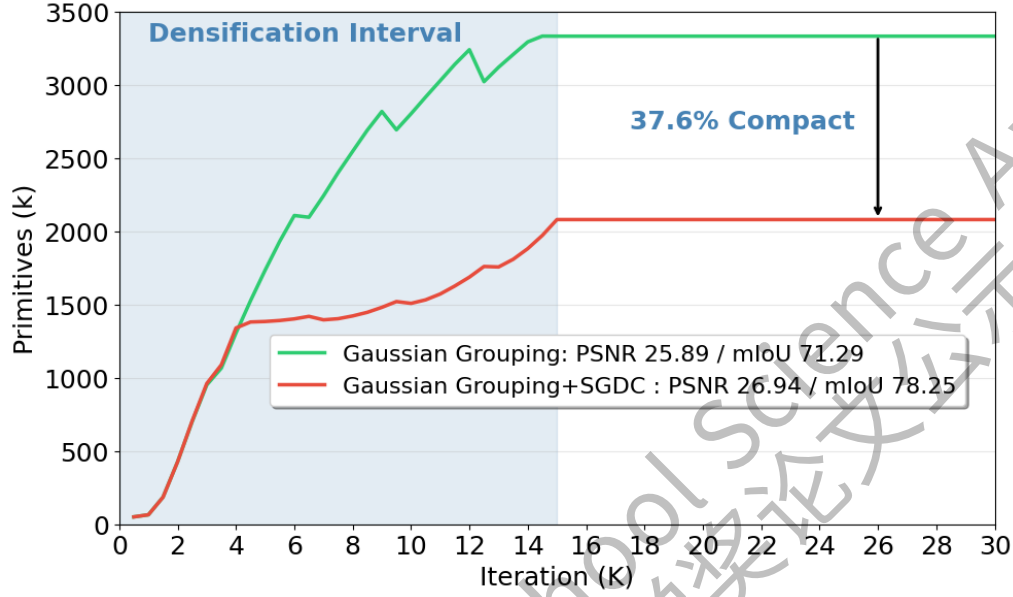


Figure 2: **Evolution of the number of Gaussian primitives during training on the *figurines* scene (Gaussian Grouping backbone).** Gaussian Grouping+SGDC (red) stabilises after activation at iteration 4,000 and converges to 2,080k primitives, 37.6% fewer than the Gaussian Grouping baseline (green, 3,332k).

comparisons.

Figure 3 compares the RGB rendering and semantic segmentation results on the *figurines* scene of LERF-Mask, which contains cluttered objects with ambiguous boundaries. Zoomed-in views of representative local regions are shown on the right to make the differences easier to observe. Gaussian Grouping+SGDC preserves rendering fidelity comparable to Gaussian Grouping, while producing noticeably sharper and more complete semantic masks in confusing regions such as the red apple and the green toy chair (mIoU 71.29 $\rightarrow$ 78.25). This demonstrates that SGDC improves boundary quality in complex scenes while reducing the number of Gaussian primitives from 3,332k to 2,080k.

Figure 4 visualises the Gaussian centre points, ellipsoids, and rendered results for the *ramen* and *teatime* scenes, together with zoomed-in views of selected regions. Compared with Gaussian Grouping, the centre points of Gaussian Grouping+SGDC are significantly sparser along the edges of the ramen bowl in the *ramen* scene and the edges of the tabletop in the

Table 5: Quantitative comparison on the Mip-NeRF 360 dataset (outdoor scenes). All metrics are computed on the held-out test views.

Method	Rendering quality			Compactness		Efficiency		
	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	Units $\downarrow$	Mem $\downarrow$	Train $\downarrow$	FPS $\uparrow$	Peak GPU $\downarrow$
<i>Non-semantic 3DGS</i>								
3DGS [1]	29.39	0.9178	0.1135	997k	236	8.88	208.30 ^a	3.14
Reduced-3DGS [36]	29.27	0.9164	0.1148	1159k	76	9.92	199.26 ^a	3.16
SteepGS [18]	29.03	0.9104	0.1265	721k	171	7.61	262.99 ^a	4.71
<i>Semantic 3DGS</i>								
Gaussian Grouping [3]	28.88	0.9190	0.1460	1853k	552	37.52	16.05 ^b	16.34
Gaussian Grouping+SGDC	29.07	0.9216	0.1441	1090k	325	28.62	16.87^b	11.01
ObjectGS [5]	30.30	0.9347	0.0935	303k	88	62.65	32.14 ^b	6.03
ObjectGS+SGDC	30.32	0.9351	0.0933	197k	60	56.11	39.53^b	4.21

PSNR in dB; Mem in MB; train time in minutes; peak GPU in GB. Units: Gaussians for 3DGS/Gaussian Grouping, anchors for ObjectGS. ^aFPS: RGB-only; ^bFPS: RGB+Seg.

teatime scene. In the ellipsoid visualisation, the primitives exhibit stronger anisotropy, appearing more slender and better aligned with the underlying surfaces. Despite the visibly sparser distribution, the rendered results maintain a high degree of visual fidelity.

4.4. Ablation Study

We conduct ablation studies on the *figurines* scene of LERF-Mask using the Gaussian Grouping backbone, in order to validate the effectiveness of the proposed components and to analyse parameter sensitivity.

4.4.1. Component Ablation

Confidence-Guided Semantic Labelling. In this ablation, we remove the confidence-guided semantic labelling module (denoted as w/o Semantic) and treat all Gaussian primitives as a single instance during optimisation. As shown in Table 7, this causes a clear degradation: PSNR drops from 26.94 dB to 25.11 dB (−1.83 dB), the number of Gaussian primitives increases by approximately 13.5% (2,360k vs. 2,080k), and semantic boundary accuracy declines sharply. This provides evidence for the important role of semantic priors in density control. Without semantic guidance, the algorithm suffers from biases under global gradient statistics, leading to the blind expansion of primitives in low-confidence regions. This not only wastes storage resources but also compromises rendering quality and boundary sharpness due to boundary point misalignment and geometric noise.

Table 6: Quantitative comparison on the Replica dataset. Instance-level segmentation is reported as per-instance IoU and Dice averaged over all object instances. All metrics are computed on the held-out test views.

Method	Rendering quality			Semantics		Compactness		Efficiency		
	PSNR↑	SSIM↑	LPIPS↓	mIoU↑	mDice↑	Units↓	Mem↓	Train↓	FPS↑	Peak GPU↓
<i>Non-semantic 3DGS</i>										
3DGS [1]	44.15	0.9914	0.0255	–	–	254k	61	3.37	381.85 ^a	2.14
Reduced-3DGS [36]	44.47	0.9916	0.0252	–	–	320k	21	4.46	324.29 ^a	1.82
SteepGS [18]	43.91	0.9908	0.0303	–	–	247k	59	4.44	371.42 ^a	2.48
<i>Semantic 3DGS</i>										
Gaussian Grouping [3]	44.50	0.9914	0.0499	93.20	96.27	615k	192	21.34	41.13 ^b	6.96
Gaussian Grouping+SGDC	44.52	0.9916	0.0463	93.18	96.26	336k	105	16.78	43.14^b	5.11
ObjectGS [5]	43.60	0.9929	0.0148	93.70	95.39	1612k	468	62.48	13.33 ^b	14.63
ObjectGS+SGDC	44.13	0.9931	0.0146	93.12	94.74	1072k	311	55.20	19.39^b	10.37

PSNR in dB; Mem in MB; train time in minutes; peak GPU in GB. Units: Gaussians for 3DGS/Gaussian Grouping, anchors for ObjectGS. ^aFPS: RGB-only; ^bFPS: RGB+Seg. Replica semantics are instance-level IoU/Dice averaged over all object instances.

Semantic Adaptive Splitting with Annealing. When semantic adaptive splitting and annealing mechanisms are removed (denoted as w/o Adaptive), the model reverts to a global fixed-threshold splitting strategy. As shown in Table 7, the number of Gaussian primitives drops to the lowest level in the table at 1,375k (a 33.9% reduction compared to the full model); however, this comes at the cost of a drastic collapse in semantic accuracy, with mIoU dropping from 78.25% to 68.53% (−9.72 points). This indicates that the adaptive mechanism is important for capturing complex geometric edges and fine-grained semantic features. By differentially allocating splitting resources based on gradient distributions within semantic instances, it ensures that complex regions are sufficiently densified, thereby maintaining a high level of semantic segmentation accuracy.

Gradient-Aware Pruning. In this ablation, we remove the gradient-aware pruning module (denoted as w/o Prune) and retain only the original opacity and screen-space pruning rules of 3DGS. As shown in Table 7, the number of Gaussian primitives surges to 3,468k, an increase of approximately 66.7% over the full model, while PSNR decreases by 0.89 dB. Notably, mIoU and BmIoU remain close to those of the full model (78.30 vs. 78.25 and 75.41 vs. 74.99), confirming that the module is not merely a compression tool but accurately eliminates redundant primitives with low rendering contribution. It reduces the primitive count by 40.0% relative to this ablated variant, or 37.6% relative to the Gaussian Grouping baseline, with negligible impact on

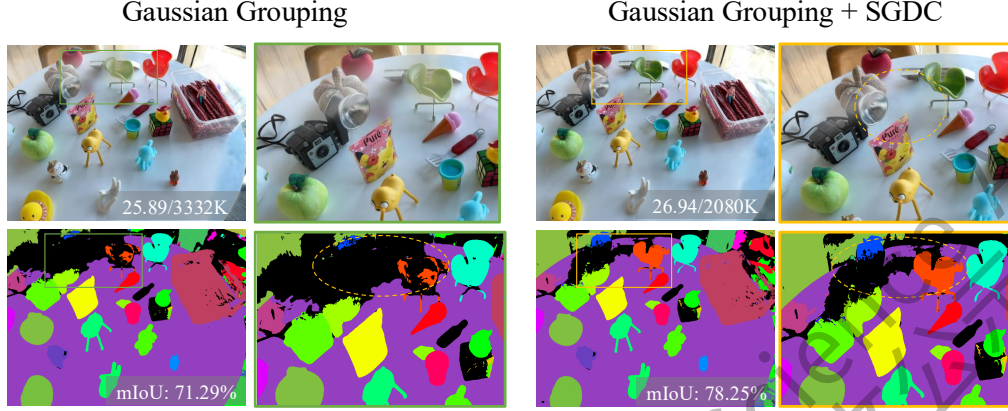


Figure 3: **Qualitative comparison on the *figurines* scene (Gaussian Grouping backbone).** Zoomed-in views of the red apple and green toy chair show the sharper semantic masks of Gaussian Grouping+SGDC (mIoU 71.29 $\rightarrow$ 78.25) obtained with 37.6% fewer Gaussians.

semantic boundary quality.

Beyond the quantitative results, Figure 5 provides a qualitative comparison of the ablated variants on the *figurines* scene. In the rendered RGB images, the full model reconstructs the red chair and the green chair more faithfully than the ablated variants, which appear blurred in these regions. For semantic segmentation, both the full model and w/o Prune recover the red apple, whereas removing semantic labelling or adaptive splitting leads to incomplete or merged masks in the same region. The Gaussian centre distributions show a complementary trend: w/o Prune retains a markedly denser set of centres around the green apple, reflecting the redundant units that survive when gradient-aware pruning is disabled, while the full model and w/o Adaptive produce visibly sparser distributions. Taken together, the full model achieves the best combination of rendering fidelity, boundary accuracy, and representation sparsity, consistent with the quantitative results in Table 7.

4.4.2. Sensitivity Analysis

Confidence threshold sensitivity. We evaluate the impact of the confidence threshold τ_c on the *figurines* scene while fixing all other hyperparameters. As reported in Table 8, lowering τ_c from 0.6 to 0.3 increases the number of primitives from 2,080k to 2,359k and degrades mIoU from 78.25%

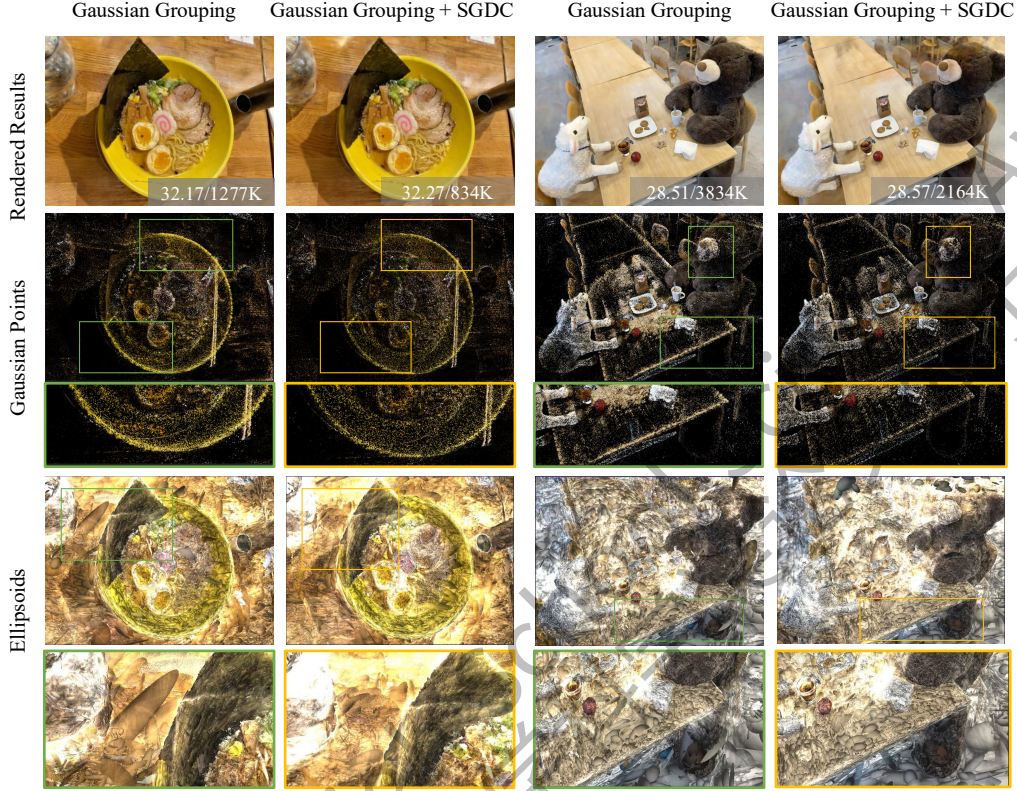


Figure 4: **Visual comparison of Gaussian distributions on LERF-Mask (Gaussian Grouping backbone).** We visualise Gaussian centre points, ellipsoids, and rendered results for the *ramen* and *teatime* scenes, with zoomed-in views of representative regions, including the ramen bowl edge and the tabletop edge.

to 65.73%. This indicates that an overly permissive threshold misclassifies unreliable boundary units as reliable, so they receive the more permissive splitting criterion intended for reliable groups; this causes excessive splitting in ambiguous regions and cross-contamination between adjacent semantic instances, which both inflates the model and blurs the boundaries. Conversely, raising τ_c to 0.9 retains only 1,633k primitives but degrades mIoU to 66.95%, because the stricter threshold marks too many transition units as uncertain and suppresses the splitting required to resolve fine details. The default $\tau_c = 0.6$ therefore gives the best trade-off between compactness and segmentation accuracy.

Gradient-quantile ablation. We fix $\tau_c = 0.6$, $q_{fg} = 0.5$, and $q_{bg} = 0.95$,

Table 7: Ablation study on the *figurines* scene. *Baseline* denotes Gaussian Grouping and *Full* denotes Gaussian Grouping+SGDC. We sequentially remove confidence-guided semantic labelling (w/o Semantic), semantic adaptive splitting with annealing (w/o Adaptive), and gradient-aware pruning (w/o Prune).

Method	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	mIoU $\uparrow$	BmIoU $\uparrow$	Primitives $\downarrow$
Baseline	25.89	0.8938	0.2104	71.29	69.17	3332k
w/o Semantic	25.11	0.8902	0.2098	73.70	70.85	2360k
w/o Adaptive	26.03	0.8978	0.2118	68.53	66.67	1375k
w/o Prune	26.05	0.8932	0.2117	78.30	75.41	3468k
Full	26.94	0.8985	0.2094	78.25	74.99	2080k

Table 8: Confidence threshold sensitivity on the *figurines* scene (Gaussian Grouping backbone), with $q_{fg} = 0.5$, $q_{bg} = 0.95$, and $q_{prune} = 0.06$ fixed. Bold marks the default setting.

τ_c	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	mIoU $\uparrow$	BmIoU $\uparrow$	Primitives $\downarrow$
0.3	25.04	0.8909	0.2133	65.73%	64.20%	2,359k
0.6	26.94	0.8985	0.2094	78.25%	74.99%	2,080k
0.9	25.39	0.8951	0.2133	66.95%	65.27%	1,633k

and vary only the pruning quantile q_{prune} . As shown in Table 9, a conservative setting ($q_{prune} = 0.04$) preserves too many primitives (2,502k) and yields the lowest PSNR (25.54 dB), because inactive units introduce geometric noise. Raising q_{prune} to 0.08 aggressively prunes down to 1,771k primitives and achieves the highest semantic scores (82.10% mIoU, 79.28% BmIoU), at the cost of a 0.90 dB PSNR drop caused by the removal of texture-bearing primitives. The default $q_{prune} = 0.06$ attains the best PSNR (26.94 dB) while retaining competitive semantic accuracy, and therefore provides the most balanced trade-off between rendering fidelity and compactness.

Annealing schedule. We compare the default linear annealing strategy (iterations 5,000–15,000) against a fixed-threshold configuration without annealing. As shown in Table 10, removing the annealing schedule reduces the number of primitives to 1,339k (−35.6%), but PSNR falls by 1.31 dB and mIoU drops to 66.17%. Without annealing, the fixed threshold cannot adapt to the evolving density distribution during early training, so the model over-

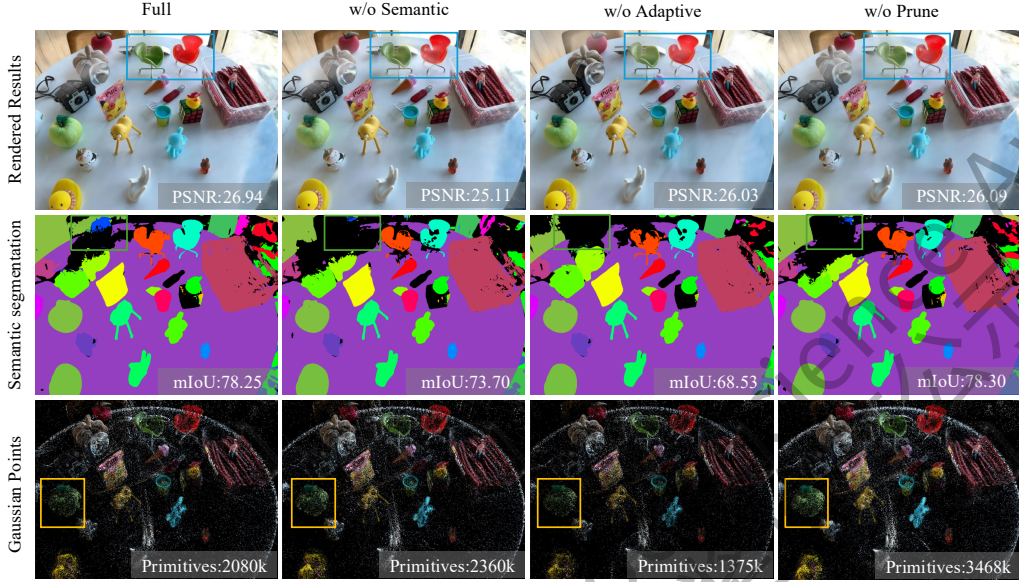


Figure 5: **Qualitative ablation on the *figurines* scene (Gaussian Grouping backbone).** Rows from top to bottom: rendered RGB, semantic segmentation, and Gaussian centres. Columns from left to right: Full (Gaussian Grouping+SGDC), w/o Semantic, w/o Adaptive, and w/o Prune.

prunes boundary units before the optimisation has converged; these units are not recovered in later iterations. The annealing schedule therefore stabilises early-stage density exploration while allowing successful refinement of fine structures in later stages.

4.5. Discussion

Semantic gains depend on supervision quality. Across the LERF-Mask scenes, SGDC improves mIoU by 0.05–6.96 points on Gaussian Grouping and by 0.01–3.55 points on ObjectGS, with the largest gains on the cluttered *figurines* scene. In contrast, on Replica the instance-level mIoU changes by only -0.02 points on Gaussian Grouping and -0.58 points on ObjectGS. This difference is consistent with the nature of the supervision: in the LERF-Mask pipeline, the training masks are generated automatically by a 2D foundation model and are noisy along object boundaries, where the geometry-driven density control of the baselines continues to densify primitives despite ambiguous semantics. SGDC’s confidence-guided criteria suppress growth in exactly these regions and preserve splitting capacity

Table 9: Gradient-quantile ablation on the *figurines* scene (Gaussian Grouping backbone), with $\tau_c = 0.6$, $q_{fg} = 0.5$, and $q_{bg} = 0.95$ fixed. Bold marks the default setting.

q_{prune}	PSNR $\uparrow$	mIoU $\uparrow$	BmIoU $\uparrow$	Primitives $\downarrow$
0.04	25.54	74.78%	72.15%	2,502k
0.06	26.94	78.25%	74.99%	2,080k
0.08	26.04	82.10%	79.28%	1,771k

Table 10: Annealing schedule comparison on the *figurines* scene (Gaussian Grouping backbone). Bold marks the default configuration.

Configuration	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	mIoU $\uparrow$	BmIoU $\uparrow$	Primitives $\downarrow$
w/ Annealing	26.94	0.8985	0.2094	78.25%	74.99%	2,080k
w/o Annealing	25.63	0.8947	0.2129	66.17%	64.11%	1,339k

for reliable instances, which translates into sharper boundaries. Replica, in contrast, provides exact synthetic labels; the baselines already converge to a clean semantic distribution, leaving little redundancy to correct, so SGDC mainly delivers compactness while keeping semantic accuracy essentially unchanged. The semantic benefit of SGDC therefore correlates with the noise level of the training supervision rather than being specific to a particular dataset.

The formulation transfers across representation units. The cross-backbone experiments show that the same formulation is effective on both a primitive-based backbone (Gaussian Grouping) and an anchor-based backbone (ObjectGS). This is a consequence of the abstraction used in Section 3.2: the controller only requires a persistent representation unit that exposes a semantic confidence score and an accumulated gradient. On a primitive-based backbone, splitting and pruning are applied directly to individual Gaussians. On an anchor-based backbone, they are applied to anchors, and pruning removes the neural Gaussians decoded by a removed anchor. A key implementation choice in the latter case is that we use the accumulated gradient of the anchor’s own centre position rather than aggregating the gradients of its decoded Gaussians. Since the decoded Gaussians are regenerated by the MLP at every iteration, aggregating their gradients can mask isolated high-contribution Gaussians near object boundaries and cause the entire an-

chor to be removed; the anchor gradient instead reflects the overall effect of the anchor on both reconstruction and semantic losses. We note that ObjectGS has a lower absolute PSNR than Gaussian Grouping across the LERF-Mask scenes; our cross-backbone comparison therefore focuses on the relative change within each backbone rather than on the absolute values.

Efficiency gains and applicability. SGDC reduces the number of representation units by 33.5–61.0% relative to the corresponding backbone, lowers memory and peak GPU usage, shortens training time, and improves FPS consistently across scenes and backbones, while the density-control modules themselves add less than 3% to the total training time. These gains follow directly from removing units that contribute little to either reconstruction or semantic prediction. Regarding generalisation beyond the semantic setting, the pruning component could in principle be transferred to a non-semantic pipeline by replacing the instance-level quantile with a global one, but it would then be functionally equivalent to existing gradient-based pruning methods, and the semantic-preservation benefit would not be obtained. The confidence-guided labelling and adaptive splitting, by contrast, cannot be defined without a per-unit confidence score. The remaining constraints of the framework, together with directions for future work, are summarised in Section 5.2.

5. Conclusion and Limitations

5.1. Conclusion

We presented a semantic-guided density control (SGDC) framework for compact semantic 3DGS. SGDC injects semantic confidence into the density-control process through three complementary components: confidence-guided semantic labelling, semantic adaptive splitting with annealing, and gradient-aware pruning. These components jointly model semantic reliability, accumulated gradients, and representation redundancy, enabling more compact representations without sacrificing rendering quality or semantic accuracy. SGDC is backbone-agnostic: it is defined on a generic representation unit and is instantiated on both a primitive-based backbone (Gaussian Grouping) and an anchor-based backbone (ObjectGS). Across LERF-Mask, Mip-NeRF 360, and Replica, SGDC reduces the number of representation units by 33.5–61.0% relative to the corresponding backbone while preserving rendering quality and maintaining or improving semantic accuracy, and it also reduces memory and peak GPU usage and improves rendering speed. These

results indicate that semantic cues can serve not only as high-level scene descriptors but also as effective optimisation signals for controlling the structural complexity of explicit 3D representations.

5.2. Limitations and Future Work

The current framework has several constraints. First, it relies on the quality of external semantic supervision: noisy masks, uncertain instance boundaries, view-inconsistent labels, and uncalibrated open-vocabulary scores propagate errors into the confidence field, causing units to be incorrectly densified or pruned. Second, because the confidence signal is derived from semantic supervision, applying SGDC to a non-semantic 3DGS pipeline would require an additional instance-grouping or pseudo-labelling stage. Third, density control on anchor-based backbones operates at the anchor level and therefore cannot selectively retain individual decoded Gaussians, which trades a degree of granularity for the storage and runtime savings reported in our experiments. In addition, the current design targets static scenes: dynamic objects violate the multi-view consistency required for gradient and confidence accumulation, and large-scale environments stress the per-instance quantile statistics and the absolute pruning threshold.

Future work will address these issues through mask-refinement networks, adaptive thresholding for open-vocabulary settings, unsupervised grouping for non-semantic pipelines, and extensions to 4D or hierarchical tile-based density control for dynamic and large-scale scenes.

References

- [1] B. Kerbl, G. Kopanas, T. Leimkuehler, G. Drettakis, 3D gaussian splatting for real-time radiance field rendering, *ACM Transactions on Graphics* 42 (4) (2023) 1–14. doi:10.1145/3592433.
- [2] S. Zhou, H. Chang, S. Jiang, Z. Fan, Z. Zhu, D. Xu, P. Chari, S. You, Z. Wang, A. Kadambi, Feature 3DGS: Supercharging 3D gaussian splatting to enable distilled feature fields, in: *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, Seattle, WA, USA, 2024, pp. 21676–21685. doi:10.1109/CVPR52733.2024.02048.
- [3] M. Ye, M. Danelljan, F. Yu, L. Ke, Gaussian grouping: Segment and edit anything in 3D scenes, in: *European conference on computer*

vision, Springer Nature Switzerland, Cham, 2024, pp. 162–179. doi: doi.org/10.1007/978-3-031-73397-0_10.

- [4] M. Qin, W. Li, J. Zhou, H. Wang, H. Pfister, LangSplat: 3D language gaussian splatting, in: 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Seattle, WA, USA, 2024, pp. 20051–20060.
- [5] R. Zhu, M. Yu, L. Xu, L. Jiang, Y. Li, T. Zhang, J. Pang, B. Dai, Objectgs: Object-aware scene reconstruction and scene understanding via gaussian splatting, in: 2025 IEEE/CVF International Conference on Computer Vision (ICCV), IEEE, 2025, pp. 8350–8360.
- [6] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, I. Sutskever, Learning transferable visual models from natural language supervision, in: Proceedings of the 38th International Conference on Machine Learning (ICML), PMLR, 2021, pp. 8748–8763.
- [7] Y. Sheng, J. Deng, X. Zhang, Y. Zhang, B. Hua, Y. Zhang, J. Ji, SpatialSplat: Efficient semantic 3D from sparse unposed images, in: 2025 IEEE/CVF International Conference on Computer Vision (ICCV), 2025, pp. 26404–26414.
- [8] Z. Dai, T. Liu, Y. Zhang, Efficient decoupled feature 3D gaussian splatting via hierarchical compression, in: 2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2025, pp. 11156–11166.
- [9] T. Lu, M. Yu, L. Xu, Y. Xiangli, L. Wang, D. Lin, B. Dai, Scaffold-GS: Structured 3D gaussians for view-adaptive rendering, in: 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Seattle, WA, USA, 2024, pp. 20654–20664. doi: [10.1109/CVPR52733.2024.01952](https://doi.org/10.1109/CVPR52733.2024.01952).
- [10] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, R. Ng, NeRF: Representing scenes as neural radiance fields for view synthesis, in: European conference on computer vision, Springer, Cham, 2020, pp. 405–421. doi: [10.1007/978-3-030-58452-8_24](https://doi.org/10.1007/978-3-030-58452-8_24).

- [11] Z. Yu, A. Chen, B. Huang, T. Sattler, A. Geiger, Mip-splatting: Alias-free 3D gaussian splatting, in: 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Seattle, WA, USA, 2024, pp. 19447–19456. doi:10.1109/CVPR52733.2024.01839.
- [12] A. Guédon, V. Lepetit, 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), IEEE, Seattle, WA, USA, 2024, pp. 5354–5363. doi:10.1109/CVPR52733.2024.00512.
- [13] Z. Liang, Q. Zhang, Y. Feng, Y. Shan, K. Jia, GS-IR: 3D gaussian splatting for inverse rendering, in: 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Seattle, WA, USA, 2024, pp. 21644–21653. doi:10.1109/CVPR52733.2024.02045.
- [14] B. Kerbl, A. Meuleman, G. Kopanas, M. Wimmer, A. Lanvin, G. Drettakis, A hierarchical 3D gaussian representation for real-time rendering of very large datasets, *ACM Transactions on Graphics* 43 (4) (Jul. 2024). doi:10.1145/3658160.
- [15] A. Hanson, A. Tu, G. Lin, V. Singla, M. Zwicker, T. Goldstein, Speedy-splat: Fast 3D gaussian splatting with sparse pixels and sparse primitives, in: 2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2025, pp. 21537–21546.
- [16] J. C. Lee, D. Rho, X. Sun, J. H. Ko, E. Park, Compact 3D gaussian representation for radiance field, in: 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2024, pp. 21719–21728.
- [17] Z. Fan, K. Wang, K. Wen, Z. Zhu, D. Xu, Z. Wang, LightGaussian: Unbounded 3D gaussian compression with 15x reduction and 200+ FPS, in: *Advances in neural information processing systems* 37, 2024, pp. 140138–140158.
- [18] P. Wang, Y. Wang, D. Wang, S. Mohan, Z. Fan, L. Wu, R. Cai, Y.-Y. Yeh, Z. Wang, Q. Liu, R. Ranjan, Steepest descent density control

- for compact 3D gaussian splatting, in: 2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2025, pp. 26663–26672.
- [19] H. Li, Y. Wu, J. Meng, Q. Gao, Z. Zhang, R. Wang, J. Zhang, InstanceGaussian: Appearance-Semantic Joint Gaussian Representation for 3D Instance-Level Perception, in: 2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2025, pp. 14078–14088.
 - [20] J. Zhang, J. Jiang, Y. Chen, K. Jiang, X. Liu, COB-GS: Clear object boundaries in 3DGS segmentation based on boundary-adaptive gaussian splitting, in: 2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2025, pp. 19335–19344. doi:10.1109/CVPR52734.2025.01801.
 - [21] X. Zuo, P. Samangouei, Y. Zhou, Y. Di, M. Li, FMGS: Foundation Model Embedded 3D Gaussian Splatting for Holistic 3D Scene Understanding, *International Journal of Computer Vision* 133 (2) (2025) 611–627. doi:10.1007/s11263-024-02183-8.
 - [22] J. Piekenbrinck, C. Schmidt, A. Hermans, N. Vaskevicius, T. Linder, B. Leibe, OpenSplat3D: Open-vocabulary 3D instance segmentation using gaussian splatting, in: 2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2025, pp. 5246–5255.
 - [23] Y. Lu, Y. Zhou, Y. Qiao, C. Song, T. Liang, J. Ma, H. Wang, Y. Yin, Segment then splat: Unified 3D open-vocabulary segmentation via gaussian splatting, in: *Advances in Neural Information Processing Systems (NeurIPS)*, 2025, pp. 165975–165994. doi:10.48550/arXiv.2503.22204.
 - [24] W. Sun, Y. Zhou, J. Jiao, Y. Li, CAGS: Open-vocabulary 3D scene understanding with context-aware gaussian splatting, *Image and Vision Computing* 165 (2026) 105830. doi:10.48550/arXiv.2504.11893.
 - [25] Z. Gao, L. Li, X. Liu, F. Liu, L. Jiao, P. Chen, W. Ma, S. Yang, Evolving semantic propagation for aerial semantic 3D gaussian splat-

ting, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 40, 2026, pp. 4221–4229.

- [26] Y. Huang, L. Bai, B. Cui, K. Yuan, G. Wang, M. I. Hoque, N. Padoy, N. Navab, H. Ren, SurgTPGS: Semantic 3D surgical scene understanding with text promptable gaussian splatting, in: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer Nature Switzerland, Cham, 2025, pp. 584–594. doi:10.48550/arXiv.2506.23309.
- [27] Z. Zheng, D. Zhou, Y. Shao, X. Yang, EGU-GS: Efficient gaussian utilization for real-time 3D gaussian splatting, Image and Vision Computing 162 (Oct. 2025). doi:10.1016/j.imavis.2025.105687.
- [28] Z. Zhou, Y.-J. Xiong, J.-C. Zhang, C.-M. Xia, X. Qiu, H. Zhan, Gradient-direction-aware density control for 3D gaussian splatting, in: International Conference on Learning Representations (ICLR), 2026, pp. 1–10. doi:10.48550/arXiv.2508.09239.
- [29] Y. Liu, Z. Zhong, Y. Zhan, S. Xu, X. Sun, MaskGaussian: Adaptive 3D gaussian representation from probabilistic masks, in: 2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2025, pp. 681–690.
- [30] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, P. Dollár, R. Girshick, Segment anything, in: 2023 IEEE/CVF International Conference on Computer Vision (ICCV), IEEE, Paris, France, 2023, pp. 3992–4003. doi:10.1109/ICCV51070.2023.00371.
- [31] J. T. Barron, B. Mildenhall, D. Verbin, P. P. Srinivasan, P. Hedman, Mip-NeRF 360: Unbounded anti-aliased neural radiance fields, in: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, New Orleans, LA, USA, 2022, pp. 5460–5469. doi:10.1109/CVPR52688.2022.00539.
- [32] J. Straub, T. Whelan, L. Ma, Y. Chen, E. Wijmans, S. Green, J. J. Engel, R. Mur-Artal, C. Ren, S. Verma, A. Clarkson, M. Yan, B. Budge, Y. Yan, X. Pan, J. Yon, Y. Zou, K. Leon, N. Carter, J. Briales, T. Gillingham, E. Mueggler, L. Pesqueira, M. Savva, D. Batra,

- H. M. Strasdat, R. D. Nardi, M. Goesele, S. Lovegrove, R. Newcombe, The replica dataset: A digital replica of indoor spaces (Jun. 2019). doi:10.48550/arXiv.1906.05797.
- [33] Z. Wang, A. Bovik, H. Sheikh, E. Simoncelli, Image Quality Assessment: From Error Visibility to Structural Similarity, *IEEE Transactions on Image Processing* 13 (4) (2004) 600–612. doi:10.1109/TIP.2003.819861.
- [34] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, O. Wang, The unreasonable effectiveness of deep features as a perceptual metric, in: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, IEEE, Salt Lake City, UT, USA, 2018, pp. 586–595. doi:10.1109/CVPR.2018.00068.
- [35] B. Cheng, R. Girshick, P. Dollár, A. C. Berg, A. Kirillov, Boundary iou: Improving object-centric image segmentation evaluation, in: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2021, pp. 15329–15337.
- [36] P. Papantonakis, G. Kopanas, B. Kerbl, A. Lanvin, G. Drettakis, Reducing the memory footprint of 3d gaussian splatting, *Proceedings of the ACM on Computer Graphics and Interactive Techniques* 7 (1) (May 2024).

Appendix: Response to Reviewers

参赛学生吴泽荟作为共同第一作者，于 2026 年 6 月 8 日将本次参赛的论文（编号 CAG-D-26-00597）投稿至期刊 *Computers & Graphics* (ISSN: 0097-8493)，并于 2026 年 7 月 7 日收到 major revision 的审稿结果。按审稿意见修改完善后，于 2026 年 9 月 14 日提交论文修改稿。目前提交参赛的为大修后的最终稿件，同时附上对审稿人意见的回复信，供评委老师们审阅。

Manuscript ID: CAG-D-26-00597

Article Title: “Semantic-Guided Density Control for Compact Semantic 3D Gaussian Splatting”

To: Computers & Graphics Editor

Re: Response to reviewers

Dear Editor,

Thank you for giving us the opportunity to revise our manuscript. We have carefully addressed all the reviewers' comments and made substantial revisions accordingly.

We are uploading (a) our point-by-point response to the comments (below), (b) a fully anonymized PDF version of our revised manuscript, and (c) full LaTeX source files with complete author details.

Response to the Editor's concerns

We thank the editor for the summary of the remaining concerns, and we address each of them directly below.

(1) Novelty. In the original submission, SGDC was evaluated only on Gaussian Grouping, which made it difficult to distinguish it from a backbone-specific improvement. We now implement SGDC on ObjectGS, an anchor-based semantic method (ICCV 2025) that is structurally different. This exposes a cross-representation limitation: the importance of an anchor cannot be obtained by aggregating the gradients of the Gaussians it decodes. With an anchor-gradient criterion, the formulation reduces the anchor count by up to 61.0% while preserving rendering and semantic quality. The contribution therefore moves from a single-backbone improvement to a representation-agnostic density-control signal for semantic 3DGS.

(2) Positioning. The revised Related Work covers representative density-control optimisation, pruning, and parameter-compression methods (LightGaussian, SteepGS, EGU-GS, GDAGS, Speedy-Splat, MaskGaussian, and Compact3D) as well as recent semantic 3DGS methods (Feature 3DGS, Gaussian Grouping, LangSplat, and ObjectGS). Table 1 summarises the decision signal, representation unit, density-control stage, semantic supervision, and relative cost of each method and of ours.

(3) Experiments versus claims. The claims have been aligned with the evidence. The title, abstract, and conclusion now scope the contribution to compact semantic 3DGS; the "plug-and-play" claim has been removed and replaced with a backbone-agnostic formulation with explicit adaptation points; the compression figure is reported as 33.5–61.0%; and the statement about semantic accuracy has been qualified to "improving or maintaining". We have also added the efficiency metrics (FPS, training time, peak GPU usage, and the sub-3% density-control overhead), the BmIoU definition, and the correction of the Replica metric that were previously missing.

Best regards,

All authors

Reviewer#1, Comment # 1: The paper proposes incorporating semantic confidence into density control. The presented framework is practical and demonstrates meaningful reductions in Gaussian primitives while maintaining rendering and semantic quality. However, the overall contribution appears somewhat incremental, as it primarily modifies existing splitting and pruning strategies with semantic-aware criteria rather than introducing a fundamentally new representation or optimization paradigm.

Author response: We thank the reviewer for this candid assessment. We agree that the individual building blocks (splitting and pruning) are not new; our contribution is the design principle that semantic confidence can serve as an *active optimisation signal* for density control, together with the demonstration that this principle is representation-agnostic. In the original submission, SGDC was validated only on Gaussian Grouping, which made it difficult to distinguish the method from a backbone-specific modification. In the revision, we instantiate the same formulation on a structurally different backbone, ObjectGS, which represents the scene with anchors rather than individual Gaussians. Across five scenes, we obtain reductions of 34.7–45.4% in representation units on Gaussian Grouping and 33.5–61.0% on ObjectGS, with preserved rendering quality and maintained or improved semantic accuracy on both backbones. This cross-paradigm validation distinguishes SGDC from a Gaussian-Grouping-specific modification and supports the claim of a reusable density-control formulation. The revised manuscript also (i) formalises SGDC on a generic representation unit with explicit adaptation points, (ii) explains the mechanism by which semantic losses affect gradients at ambiguous boundaries and how confidence-guided thresholds counteract it, and (iii) strengthens the positioning against recent compact and semantic 3DGS methods.

Author action: We updated the manuscript by (i) rewriting the contribution list and the Introduction paragraph to state the design principle rather than the module list, (ii) generalising the formulation in Sections 3.2–3.3 to a generic representation unit, (iii) adding the anchor-based instantiation of ObjectGS and its gradient design, (iv) adding cross-backbone results to Tables 2–6 and Section 4.5, and (v) strengthening the positioning in Section 2.3 and Table 1.

Reviewer#1, Comment # 2: In addition, the paper would benefit from a broader discussion of recent object-aware semantic Gaussian, such as ObjectGS (ICCV 2025), which also leverage semantic structure to improve representation efficiency and scene understanding.

Author response: We agree that ObjectGS is a highly relevant recent work and that it was insufficiently discussed in the original submission. ObjectGS adopts an object-aware, anchor-based representation in which a sparse set of anchors decodes neural Gaussians; this improves representational efficiency, but its density control remains geometry- and appearance-driven, so semantic confidence does not participate in anchor growth or pruning. Our work is therefore complementary: rather than redesigning the representation, we inject semantic confidence into the density-control stage. Importantly, we have gone beyond a discussion-only response. We now use ObjectGS as the second backbone in our validation, which is structurally different from the primitive-based Gaussian Grouping: one anchor decodes multiple neural Gaussians, so density control operates on anchors rather than on individual Gaussians. Instantiating SGDC on ObjectGS required adapting the gradient signal to the anchor level (see our reply to Comment #4 for the technical details). With this adaptation, SGDC reduces the number of anchors by 33.5–61.0% across five scenes while preserving rendering quality and maintaining or improving semantic accuracy. This directly addresses the reviewer's point that semantic structure should be exploited to improve representation efficiency, and it demonstrates that the benefit is not specific to Gaussian Grouping.

Author action: We updated the manuscript by (i) adding a dedicated paragraph on ObjectGS in Section 2.2 that clarifies the complementary relationship, (ii) adding an ObjectGS row to Table 1 with its decision signal and density-control stage, (iii) adding the anchor-based instantiation of SGDC in Section 3.3 including the anchor-gradient design, (iv) reporting ObjectGS results on five scenes in Tables 2–6, and (v) discussing the cross-backbone findings in Section 4.5.

Reviewer#1, Comment # 3: Gaussian compression works are also lacking sufficient depth of coverage and discussion.

Author response: We agree that the coverage of compression works was too shallow. We have restructured Section 2.3 into three categories — density-control optimisation, pruning, and parameter compression — and added the missing representative methods, together with a discussion of their common limitation: they rely solely on geometric or photometric signals and therefore cannot distinguish semantically critical primitives from photometrically

redundant ones. This limitation is precisely what motivates SGDC: by using semantic confidence as a density-control signal, our method achieves compression comparable to or stronger than these approaches while preserving semantic boundaries, as demonstrated on both backbones and summarised in Table 1.

Author action: We updated the manuscript by rewriting Section 2.3 to cover LightGaussian, SteepGS, EGU-GS, GDAGS, Speedy-Splat, MaskGaussian, and Compact3D, with a mechanism-level description of each category, an explicit statement of the semantic-agnostic limitation, and a positioning of SGDC against these methods in Table 1.

Reviewer#1, Comment # 4: Although the paper claims the density-control module is plug-and-play, most experiments are conducted on Gaussian Grouping only, making it difficult to assess the claimed generality of the approach. Additional comparisons on more diverse benchmarks, stronger positioning with respect to recent sota methods, and more analysis of the semantic confidence mechanism would further strengthen the paper.

Author response: We thank the reviewer for this comment, which we have treated as the central concern of the revision. We have removed the "plug-and-play" claim and replaced it with a precise, backbone-agnostic formulation that operates on a generic representation unit with explicit adaptation points. The comment raised three specific points, which we address below.

(1) Generality beyond Gaussian Grouping. We now instantiate SGDC on a second, structurally different backbone, ObjectGS, which is anchor-based rather than primitive-based. On ObjectGS, SGDC reduces the number of anchors by 33.5–61.0% across five scenes while preserving rendering quality and maintaining or improving semantic accuracy. We also analysed why aggregating sub-Gaussian gradients is unsuitable for anchor-based pruning and instead use the anchor's own accumulated gradient. This is described in our reply to Comment #2.

(2) More diverse benchmarks. The evaluation now covers five scenes spanning three datasets with different supervision characteristics: LERF-Mask (real scenes with automatic, noisy masks), Replica (synthetic scenes with exact labels), and Mip-NeRF 360 (large-scale outdoor scenes without semantic annotations). Both backbones are evaluated on all five scenes.

(3) Stronger positioning. We have added ObjectGS to the Related Work discussion and to the positioning table (Table 1), and we restructured Section 2.3 to position SGDC against representative density-control optimisation, pruning, and parameter-compression methods, with an explicit statement of their semantic-agnostic limitation.

(4) Analysis of the semantic confidence mechanism. We have added an explicit clarification of the gradient source in Section 3.2 (the accumulated gradient is computed from the total loss, i.e., reconstruction plus semantic supervision), a mechanistic explanation in Section 3.3.2 of how the confidence-based criterion modulates the splitting thresholds so that semantically unreliable gradients do not drive excessive densification, and a sensitivity analysis of the confidence threshold τ_c (Table 8) that shows how it governs the trade-off between excessive splitting in ambiguous regions and under-densification of fine structures.

Author action: We updated the manuscript by (i) generalising the formulation in Sections 3.2–3.3 to a generic representation unit, (ii) adding the anchor-based instantiation and its gradient design in Section 3.3.3, (iii) updating Algorithm 1 accordingly, (iv) adding ObjectGS results to Tables 2–6 and the Discussion in Section 4.5, (v) expanding the Related Work coverage of compression methods in Section 2.3 and the positioning table (Table 1), and (vi) adding the gradient-source clarification in Section 3.2, the confidence-conditioned threshold mechanism in Section 3.3.2, and the confidence-threshold sensitivity analysis in Table 8.

Reviewer#2, Comment # 1: In Figure 3, the visualizations of both Gaussian Grouping and the proposed method are relatively small, and the highlighted regions are not sufficiently clear. Enlarging the marked areas and providing zoomed-in comparisons would make the qualitative improvements easier to observe.

Author response: We thank the reviewer for this comment. We agree that the previous Figure 3 was too crowded, which made the comparison panels difficult to read. We have reworked the figure in two ways. First, we now focus on the *figurines* scene, where the qualitative difference is most pronounced (mIoU 71.29% to 78.25%); the *ramen* scene was removed from this figure because its differences fall within the noise level, and its evidence is retained in Table 2 and Figure 4. Second, we enlarged the comparison panels and added zoomed-in views of the red apple and the green toy chair, where the two methods are shown side by side at the same magnification and colour mapping.

Author action: We updated the manuscript by reworking Figure 3 to focus on the *figurines* scene, enlarging the comparison panels, adding colour-coded boxes and zoomed-in views of the red apple and the green toy chair, and revising the caption and the corresponding text in Section 4.3.

Reviewer#2, Comment # 2: The differences shown in Figure 4 are not particularly obvious. It would be beneficial to enlarge representative local regions and provide more detailed visual comparisons to better illustrate the advantages of the proposed method.

Author response: We thank the reviewer for this comment. We agree that the differences in Figure 4 were difficult to discern at the original scale. We have therefore added zoomed-in views of representative local regions, such as the ramen bowl edge and the tabletop edge, where the differences in sparsity and anisotropy are most pronounced. Each zoomed-in view shows Gaussian Grouping and Gaussian Grouping+SGDC side by side, cropped from the same region and at the same magnification. In these views, the centre points of Gaussian Grouping+SGDC are visibly sparser along the edges, and the ellipsoids are more elongated and better aligned with the underlying surfaces.

Author action: We updated the manuscript by adding colour-coded boxes in Figure 4 to indicate the selected regions, inserting corresponding zoomed-in comparison panels for both methods, and revising the caption and the text in Section 4.3 to point to these regions and to highlight the observed differences in point sparsity and ellipsoid anisotropy.

Reviewer#2, Comment # 3: The ablation study provides quantitative evidence that the proposed modules contribute to preserving semantic boundaries and capturing fine-grained features. However, the discussion is primarily numerical. Additional qualitative visualizations would help readers better understand the specific effects of each component on scene representation and semantic segmentation quality.

Author response: We agree that the ablation was primarily numerical, and we thank the reviewer for this suggestion. We have added a qualitative ablation figure (Figure 5) that shows the effect of each component on rendering, semantic segmentation, and the distribution of Gaussian centres. The figure reveals a distinct failure mode for each removed component: without semantic labelling, the semantic mask is contaminated along ambiguous boundaries; without adaptive splitting, fine-detail structures are under-densified; and without pruning, the Gaussian centres remain markedly denser around regions such as the green apple without improving rendering or segmentation. The full model achieves the most favourable combination of rendering fidelity, boundary accuracy, and sparsity, which is consistent with the quantitative results in Table 7.

Author action: We updated the manuscript by adding Figure 5, which compares *w/o Semantic*, *w/o Adaptive*, *w/o Prune*, and the full model on the *figurines* scene using the same checkpoints as the quantitative ablation in Table 7. We also added a paragraph in Section 4.4.1 that explains how each ablated component affects the rendering, the semantic mask, and the distribution of Gaussian centres, and we highlighted the red chair, green chair, and red apple in the figure.

Reviewer#2, Comment # 4: The gradient-aware pruning module appears effective in removing redundant Gaussian primitives. Since the pruning criterion is largely based on gradient statistics, it would be valuable to discuss whether this strategy can be generalized to non-semantic Gaussian Splatting frameworks and whether similar compression benefits could be achieved without semantic supervision.

Author response: We thank the reviewer for raising this point, which we had not addressed explicitly. The three components of SGDC depend differently on semantic supervision. The gradient-aware pruning component can, in principle, be transferred to a non-semantic pipeline by replacing the instance-level quantile with a global quantile; in that form it would still provide compression, but it would be functionally equivalent to existing gradient-based pruning methods such as LightGaussian and SteepGS, and our absolute gradient bound would remain the only additional constraint. The confidence-guided semantic labelling and the asymmetric splitting, however, cannot be defined without a per-unit confidence score, because they rely on distinguishing reliable from uncertain regions. Consequently, similar compression could be achieved without semantic supervision, but the semantic-preservation benefit — which is the core contribution of SGDC — would not be obtained.

Author action: We updated the manuscript by adding a discussion of non-semantic 3DGS pipelines in Section 4.5 that distinguishes the transferable pruning component from the semantic components, and a corresponding limitation and future-work item in Section 5.2, where we note that an additional instance-grouping or pseudo-labelling stage would be required to use the full framework.

Reviewer#3, Comment # 1: The main concern is that the compactness claim is broader than what the experiments support. The title, abstract, and conclusion describe the method as a compact and efficient 3D Gaussian scene representation in general. However, the reported advantage is mainly relative to Gaussian Grouping, which is a semantic 3DGS baseline. In the comparisons with vanilla 3DGS and compact 3DGS variants, the proposed method does not consistently have fewer primitives, lower memory, or better rendering quality. This means the contribution is better framed as compact semantic 3DGS, or compact density control for Gaussian Grouping-style representations, rather than compact 3DGS representation in general. The paper should clarify this scope throughout the manuscript.

Author response: We agree that the original compactness claim was broader than what the experiments support. We have therefore aligned the scope throughout the manuscript and now frame the contribution as compact semantic 3DGS rather than compact 3DGS in general. We have also removed the "plug-and-play" claim and replaced it with a backbone-agnostic formulation that has explicit adaptation points. To show that this scope is not limited to a Gaussian Grouping-style representation, we instantiate SGDC on ObjectGS, an anchor-based semantic 3DGS that is structurally different from Gaussian Grouping. This keeps the contribution within semantic pipelines, as suggested by the reviewer, while demonstrating that the advantage is not specific to a single semantic backbone. Regarding the comparisons, vanilla 3DGS, Reduced-3DGS, and SteepGS are non-semantic methods and are included as contextual reference points. Because they do not provide instance-level semantics and follow different rendering protocols, cross-group results are not used to support the main compactness or efficiency claims. The primary comparison is conducted within semantic 3DGS, relative to Gaussian Grouping and ObjectGS under the same evaluation protocol.

Author action: We updated the manuscript by (i) revising the title, abstract, Introduction, and Conclusion to state the semantic scope explicitly, (ii) replacing the "plug-and-play" claim with a backbone-agnostic formulation with explicit adaptation points, (iii) adding ObjectGS as a second backbone, with results on five scenes reported in Tables 2–6 and discussed in Section 4.5, and (iv) explicitly separating non-semantic and semantic methods in Tables 2–6 and limiting the main compactness comparison to the semantic group.

Reviewer#3, Comment # 2: The motivation also needs clarification. The paper says semantic extensions inherit photometric density control and therefore may over-densify semantically ambiguous regions or object boundaries. It is not clear whether semantic supervision itself causes extra densification, or whether the issue is only that vanilla density control does not use semantic information. This distinction is important because the manuscript describes the gradient source inconsistently: the method section defines gradients from reconstruction loss, while the algorithm appears to backpropagate the total loss including semantic losses. The authors should clearly state which gradients are used for splitting and pruning, and explain how semantic losses affect density evolution if they are involved.

Author response: We thank the reviewer for identifying this ambiguity, which had two aspects. The first is the source of the gradient. We have now stated explicitly that the accumulated gradient used for density control is computed from the total training objective — the reconstruction loss together with the semantic supervision terms — rather than from the reconstruction loss alone, and that both the splitting and the pruning criteria operate on this same accumulated gradient. The second aspect is the mechanism. The primary limitation is that semantic 3DGS pipelines inherit density control that is semantically agnostic, because the reconstruction term dominates the gradient and contains no semantic information. In addition, because the total loss also contains semantic terms, the gradient at semantically ambiguous regions combines the reconstruction error with the semantic error, so densification there is driven by both — and can be largest exactly where the supervision is least reliable. This is why a confidence-based criterion is needed to modulate the thresholds in such regions, and we now explain this explicitly.

Author action: We updated the manuscript by (i) rewriting the motivation in the Introduction, (ii) adding a dedicated "Gradient source" paragraph in Section 3.2 that defines the loss as the total training objective and states that splitting and pruning share the same accumulated gradient, (iii) explaining how the confidence-based criterion modulates the splitting thresholds in Section 3.3.2, (iv) updating Algorithm 1 so that the total loss is back-propagated, and (v) adding the confidence-threshold sensitivity analysis in Table 8.

Reviewer#3, Comment # 3: In Section 4.4.2 and Table 8, the text says that decreasing τ_c from 0.6 to 0.3 makes the number of Gaussians drop from 2,080k to 2,359k, but the number actually increases. The explanation of this trend is also unclear.

Author response: We thank the reviewer for catching this error. The number of primitives increases, not decreases, when the confidence threshold τ_c is lowered from 0.6 to 0.3. We have corrected the text and rewritten the explanation. Lowering τ_c makes the threshold more permissive, so unreliable boundary units are classified as reliable and receive the splitting criterion intended for reliable groups; this causes excessive splitting in ambiguous regions and cross-contamination between adjacent instances, which both inflates the model and blurs the boundaries. Conversely, raising τ_c to 0.9 marks too many transition units as uncertain, so the stricter criterion suppresses the splitting required to resolve fine details, which reduces the primitive count but degrades mIoU to 66.95%. The default $\tau_c = 0.6$ lies between these two failure modes and gives the best trade-off, which is now explained consistently with Table 8.

Author action: We updated the manuscript by correcting the trend in Section 4.4.2, rewriting the mechanistic explanation to cover both the low- and high-threshold cases, and aligning the text with Table 8 so that the table, the text, and the stated trend are mutually consistent.

Reviewer#3, Comment # 4: The paper discusses efficiency, and Table 1 even includes a computational-cost column, but FPS, training time, and the overhead of the added density-control steps are not reported. BmIoU is not clearly defined in Section 4.1. Section 4.1 says Replica is evaluated with BmIoU, but Table 6 reports mIoU for Replica, while Tables 7-10 report both mIoU and BmIoU. In Section 4.4.1, the w/o Adaptive paragraph refers to Table 4, but it appears to mean Table 7.

Author response: We thank the reviewer for these detailed corrections. We have addressed each of them as follows.

(i) Efficiency metrics. FPS, training time, and peak GPU usage are now reported in Tables 2–6. The overhead of the density-control modules is reported in Section 4.2: the three SGDC components add less than 3% to the total training time, which is more than offset by the reduction in the number of representation units.

(ii) BmIoU definition. BmIoU is now defined in Section 4.1 as the Boundary IoU of Cheng et al., which restricts the IoU computation to a band around the ground-truth instance boundary.

(iii) Replica metric consistency. We found that the original text and Table 6 used different metric families for Replica. We have unified them: Replica is now evaluated with instance-level IoU and Dice averaged over all object instances, which is consistent with the instance-level protocol used on LERF-Mask and with the instance annotations provided by Replica. Both Section 4.1 and Table 6 have been updated to state this metric explicitly.

(iv) Table reference. The incorrect reference in Section 4.4.1 (Table 4 instead of Table 7) has been corrected.

Author action: We updated the manuscript by (i) adding FPS, training time, and peak GPU columns to Tables 2–6 and reporting the sub-3% density-control overhead in Section 4.2, (ii) defining BmIoU in Section 4.1, (iii) stating the Replica metric consistently as instance-level IoU and Dice in both Section 4.1 and Table 6, and (iv) correcting the reference in Section 4.4.1 from Table 4 to Table 7.

Acknowledgements, Declarations and Contributions

1. 研究背景

3D Gaussian Splatting (3DGS) 作为一种显式三维场景表示方法, 因其高保真建模、实时渲染、易于编辑等优势, 被广泛应用于 AR/VR 沉浸式交互、机器人导航及数字孪生等领域。尽管近年的语义 3DGS 研究取得了显著进展, 但由于沿用了传统的几何驱动密度控制策略, 导致重建的三维模型出现严重的图元冗余和过度生长问题。为此, 本文提出了一种语义引导的密度控制框架, 将语义置信度作为主动优化信号引入模型演化过程, 在确保渲染质量与语义精度的同时, 实现了更紧凑、高效的三维场景表示。

2. 指导老师与参赛学生的关系

指导老师王芳所在单位湖北第二师范学院作为湖北省教师教育的重要基地之一, 是“教育部人工智能助推教师队伍建设的试点”高校, 设有“基础教育数字化”湖北省协同创新中心, 肩负全省中小学教育共同体建设的职责, 搭建教学、科研与社会服务公共平台。王芳老师多次赴武汉康礼高级中学为师生科普人工智能前沿进展, 其中空间智能方向的“三维重建”专题引发吴泽荟同学的强烈兴趣。之后, 王芳老师引导其调研并实践先进的 3DGS 算法, 指导其选题和论文写作规范。指导老师与参赛学生不存在亲属关系, 指导过程中仅提供选题建议、算力资源、格式校对, 不参与算法、编码、实验、论文实质撰写, 从未收取参赛学生、学生家长以及学生所在学校提供的任何报酬。在此, 对王芳老师的悉心指导与无私帮助表示衷心的感谢。

3. 整体研究过程、分工、遇到的困难及解决方案

参赛学生吴泽荟独立完成问题建模、算法原创设计与数学推导、代码实现、实验与结果分析、论文撰写与修改, 承担本研究全部创造性智力工作; 指导老师王芳提供选题方向建议、算力与数据集访问权限、论文写作规范性指导。整体研究过程如下:

(1) **选题来源:** 参赛学生在王芳老师引导下调研 Gaussian Grouping 开源代码, 发现该算法依靠全局固定梯度阈值触发高斯分裂, 墙面、桌面等平坦区域与细节区域竞争分裂资源, 导致模型在 Figurines 场景膨胀至 3,332k 高斯。通过对经典 3DGS [1] 与 Gaussian Grouping [3] 机制的反思, 参赛学生意识到传统方法用“一刀切”式全局阈值忽略了不同物体的结构差异。受文献[28] GDAGS 等方法利用梯度统计特征的启发, 参赛学生自主提出引入统计学分位数分析的研究设想, 并主动与指导老师交流, 经讨论后确定本课题。指导老师仅提供方向性建议, 未指定具体算法。

(2) **数据获取:** 参赛学生独立下载并处理 LERF-Mask、Mip-NeRF 360、Replica 三个公开数据集, 独立生成 SAM 语义掩码作为监督; 指导老师协助开通数据集下载权限, 未参与数据处理。

(3) **理论推导与算法设计:** 参赛学生针对语义 3DGS 继承几何驱动密度控制而引发的模型膨胀问题, 摒弃全局固定梯度阈值改用分位数自适应分裂策略, 按每个语义实例内部的梯度百分位动态分配拆分配额, 并对低置信度图元设定较高分位数门槛, 试图从源头上遏制无意义的膨胀。研究过程中又遇到了**第二个难点**: 参赛学生在观察训练前期的渲染过程时, 发现高斯的视图空间位置梯度在前 5,000 次迭代中波动剧烈、包含大量噪声。如果此时盲目激进拆分, 会把未定型的主体几何切得过于碎裂。受深度学习中的学习率衰减启发, 参赛学生设计了一个随迭代次数平滑衰减的线性退火函数 $\tilde{T}_{split}(t)$ (式 6、7), 让模型在前期专注于建立宏观骨架, 后期再降低门槛精细雕琢微观高频细节。**第三个难点**是很难清理那些在长期优化中梯度已经停滞的冗余基元, 如果简单地按相对梯度分位数剪枝, 又极易导致在墙面、桌面等平坦区域误删高斯, 产生严重的视觉空洞。经历多组敏

感性分析实验后，参赛学生意识到单一条件无法区分相对冗余与结构必需，于是引入绝对梯度下限 τ_{prune_abs} （式 8）作为安全锚点，只有同时满足“处于实例内部贡献末尾”且“绝对数值极低”两个条件的交集图元才会被剔除，兼顾了极致压缩与结构完整性。参赛学生独立完成分位数自适应分裂、退火机制、梯度感知剪枝三个模块的数学建模与公式推导及 Algorithm 1 伪代码设计，为本研究核心原创贡献。各模块实际性能效果通过后续实验进行验证。

（4）代码实现：参赛学生基于 Gaussian Grouping 开源代码库独立实现全部算法优化，改写了约 1,200 行 Python/CUDA 代码，包括卷积分类器、语义标签分配、分位数阈值计算、退火调度器等，并在实验室 A100 集群上完成训练。**这个环节最大的困难**在于投稿后审稿人对方法泛化性提出了要求。参赛学生作为高二学生，不得不在课余时间补齐 ObjectGS anchor 解码机制知识盲区，重写数据接口与密度控制，耗费了多个深夜独立复现并改造了一个结构迥异的 SOTA 方法，对工程能力是极大考验。指导老师仅提供计算资源，未编写任何代码。

（5）实验实施：参赛学生独立完成全部实验，包括 30,000 次迭代训练、LERF-Mask、Mip-NeRF 360、Replica 三套数据集的定量对比（表 2-6）、训练过程与可视化对比（图 2-4）、组件消融（表 7）与参数敏感性分析（表 8-10），并针对审稿意见补充了 FPS、训练时间、峰值 GPU 内存等效率指标，统一 Replica 指标，重新跑完所有对比实验。**该阶段的困难**在于 A100 集群需与其他团队排队共享，单次全量评测常需通宵守候。为回应审稿意见新增的 ObjectGS 实验、效率评测与消融分析，参赛学生累计多跑了上百组训练，极大考验了时间管理与抗压能力。最终，参赛学生通过大量消融实验与参数敏感性分析，表明本研究提出的分位数自适应分裂策略将 Figurines 场景原本 3,332k 高斯压缩至 2,080k，分割精度 mIoU 从 71.29% 提升至 78.25%；移除退火机制后，mIoU 从 78.25% 下降至 66.17%，PSNR 下降 1.31dB，验证了退火模块的有效性；双重约束的梯度感知剪枝机制可在几乎不牺牲语义边界精度的前提下压缩约 40% 高斯图元，清理低贡献图元带来的几何噪声，渲染画质 PSNR 提升 0.89 dB。整套框架在多套数据集上完成泛化验证，完整验证了本文三个核心模块的必要性。

（6）论文撰写与修改：参赛学生独立完成从摘要到结论的全部初稿。**困难发生在在大修阶段**，三位审稿专家提出了 12 条高价值修改意见，极具挑战性。为澄清方法定位，参赛学生重读了 36 篇文献并重写 Related Work，新增了 Table 1 方法对比表，并将初稿中夸大的“plug-and-play”修正为严谨的“backbone-agnostic”；同时，明确定义了 BmIoU 并补充数学公式，修正 τ_c 阈值讨论与表格引用错误。指导老师在完稿后对英文语法、参考文献 IEEE 格式、图表清晰度进行规范性修改建议。

本项目无其他参赛队员。除指导老师提供数据集访问权限、A100 算力集群硬件资源之外，没有其他个人或机构在算法设计、代码开发、实验实施、论文撰写等研究成果层面提供协助。本研究基于 LERF-Mask、Mip-NeRF 360、Replica 公开数据集以及 Gaussian Grouping、SAM 开源项目开展，全部遵循对应开源协议。

4. AI 使用情况说明

本课题的问题提出、数学公式推导、核心代码编写、实验设计与数据分析、论文主体撰写全部由参赛学生独立完成。仅在论文终稿完成后的形式化检查阶段有限次使用豆包，研究过程中未交由 AI 生成任何算法逻辑、实验分析与论文核心观点。

附：指导老师、参赛学生简介

指导老师王芳为湖北第二师范学院计算机与人工智能学院副教授，毕业于武汉大学，获工学博士学位，主要研究方向为计算机应用、区块链。主持和参与国家级项目 2 项、湖北省自然科学基金 3 项、其他省厅级项目 5 项及教研项目 4 项。公开发表学术论文 20 余篇，主编和参与教材编写 10 余部。长期从事教学相关工作，多次获得全国、省教学奖，指导学生参加国家级 A 类竞赛获得国家级金奖 1 项、银奖 1 项，省级奖 10 余项。官网主页：<https://jsj.hue.edu.cn/2018/0620/c1481a53102/page.htm>

参赛学生吴泽荟为武汉康礼高级中学高二 14 班学生、班长、编程社团成员。原就读于武汉音乐学院附中（六年制）琵琶专业，获校乙等奖学金及专业一等奖学金。因其疫情期间开发了一款琵琶陪练应用 PluckBuddy，从此对计算机及智能技术产生了浓厚的兴趣，遂从音乐学院附中退学并考入康礼高级中学继续学习。在校期间，多次荣获高级荣誉学生、优秀班干部称号，AP 微积分 BC、物理、统计、CSA 全球考满分，BPhO 英国物理竞赛银奖、NHSDLC 美式辩论赛华中区域第四名，并以唯一发明人公开实用新型专利“一种基于 3DGS-SLAM 的手机端盲人导航应用系统”、“一种基于头戴设备和交互传感贴片的琵琶教学设备”。