Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

CP-GOF: A Center-Object-Faithful and Age-Aware Training-Time Pruning Method for Gaussian Opacity Fields

Yudai Wang
Henan University
Dekang Yao
Henan University
Liyan Qin
Henan University
Yingao Liu
Henan University
Fanggui Cai
Henan University
Zhiying Cui
Henan University
Zan Wang
10110137@vip.henu.edu.cn

Henan University

Research Article

Keywords: 3D Gaussian Splatting, Gaussian Opacity Fields, Gaussian pruning, age awareness, center-object protection, lightweight reconstruction

Posted Date: September 4th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-10433305/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

CP-GOF: A Center-Object-Faithful and Age-Aware Training-Time Pruning Method for Gaussian Opacity Fields

Yudai Wang[1], Dekang Yao[1], Liyan Qin[1], Yingao Liu[1], Fanggui Cai[1], Zhiying Cui[1], Zan Wang[1*]

1 School of Physics and Electronics, Henan University, Jinming Street, Kaifeng, 475004, Henan, China

** Corresponding author. E-mail: 10110137@vip.henu.edu.cn*

Abstract Surface reconstruction methods based on 3D Gaussian Splatting achieve high-fidelity rendering, but at the cost of millions of redundant Gaussian primitives, resulting in enormous model storage overhead and slow mesh extraction. Existing Gaussian pruning methods largely rely on global importance measures and inevitably compromise the geometric integrity of weakly textured or weakly observed regions, producing holes in the reconstructed mesh. To address this, we propose CP-GOF, a center-object-faithful training-time pruning method for Gaussian opacity fields (GOF) that systematically resolves the tension between Gaussian bloat and center-geometry fidelity in GOF at three levels: theory, mechanism, and application. At the theoretical level, since existing training-time pruning ignores the statistical instability of newborn Gaussians, we directly reuse the opacity gradients already computed during backpropagation to construct an exponential moving average (EMA) importance measure with zero additional cost, and we establish an age-aware mechanism that guarantees the statistical reliability of pruning decisions. At the mechanism level, we embed a center-protected gradient-EMA importance pruning (CP-GEIP) module into the standard training loop and, by using a sparse-reconstruction prior to define a 3D spatial protection box, achieve absolute preservation of the center object together with adaptive pruning of redundant background Gaussians. At the application level, we perform distance-aware filtering and quadric error metrics (QEM) mesh simplification on the lightweight Gaussian field to further compress the surface representation. While keeping the native GOF representation unchanged, the entire pipeline compresses Gaussian model storage overhead and lightens the mesh, and we validate its center-object fidelity through multi-scene experiments.

Keywords: 3D Gaussian Splatting, Gaussian Opacity Fields, Gaussian pruning, age awareness, center-object protection, lightweight reconstruction

1 Introduction

Neural rendering methods based on 3DGS [1] and earlier efficient representations [2, 3] have made remarkable progress in real-time view synthesis and surface reconstruction. However, their densification mechanism causes the number of Gaussian primitives to grow explosively from an initial few tens of thousands to millions, with model storage overhead reaching hundreds of MB, which severely constrains deployment on storage-limited devices and in real-time rendering pipelines [4–7]. GOF [8] and neural surface reconstruction methods [9, 10] further introduce an opacity-field constraint to enable high-quality mesh extraction, but they inherit the densification mechanism of 3DGS, making the volume-bloat problem especially acute. Achieving a lightweight Gaussian field while preserving the surface-reconstruction capability of GOF has therefore become a key bottleneck in moving GOF from academic research toward practical applications.

Existing Gaussian-lightweighting methods face two fundamental limitations when tackling this bottleneck. The first category is post-hoc pruning: LightGaussian [11] computes saliency scores from ray-intersection frequency, Mini-Splatting [12] performs pixel-level pruning based on color-blending weights, and RadSplat [13] removes far-field Gaussians via a distance field. These methods implicitly assume that "rendering contribution is equivalent to geometric importance." However, weakly textured regions of the scene's main body (such as building facades, indoor walls, and road surfaces) and weakly observed regions (such as occluded backsides, the reverse sides of mirrors and glass panes, and structural interiors) often exhibit low rendering contribution due to gradient saturation or sparse viewpoint coverage. In global ranking, they are misjudged as prunable background regions, resulting in geometric holes.

The second category is training-time constraints: Scaffold-GS [14] adopts an anchor-neural-network hybrid structure, Octree-GS [15] organizes Gaussians hierarchically via an octree, and Compact 3D Gaussian Representation [16] uses a learnable mask to constrain the number of Gaussians. The research focus of these methods is to achieve Gaussian lightweighting while preserving GS rendering quality, compressing explicit parameters by restructuring the Gaussian representation. However, existing work has not evaluated the compatibility of their representation with subsequent mesh-extraction tasks: fundamental modifications to the Gaussian structure (such as anchoring or octree-based organization) significantly alter the distribution characteristics of the opacity field and disrupt the field continuity required for mesh extraction, degrading the performance of 3DGS-based mesh-extraction methods.

To break out of the above "pruning harms geometry, restructuring harms compatibility" dilemma, this paper builds upon GOF and proposes CP-GOF, a center-object-faithful training-time pruning method for Gaussian opacity fields that is compatible with the native GOF representation. During training, a CP-GEIP module is embedded into the standard GOF training loop to achieve adaptive control of the number of Gaussians and high-fidelity preservation of the main-body region; in post-processing, distance-aware filtering and QEM mesh simplification are performed on the lightweight Gaussian field to further compress the surface representation. Working together, the two stages achieve model lightweighting while preserving the surface-reconstruction capability of GOF. Fig. 1 provides an overview of the CP-GOF framework.

Our core contributions can be summarized at three levels. At the theoretical level, we directly reuse the opacity gradients already computed during backpropagation to construct an EMA importance measure at zero additional rendering overhead, establish a confidence interval through a gradient-statistics-bias analysis of newborn Gaussians, and determine the age threshold for training-time pruning ($T_{\text{age}} = 1500$ iterations) via EMA bias analysis and empirical parameter-convergence considerations, thereby ensuring reliable pruning decisions. At the mechanism level, we use a sparse-reconstruction prior to define a 3D spatial protection box, analytically determine the scale factor from the point-cloud coverage ratio, and thereby achieve a balance of "aggressive background pruning and absolute center preservation", while introducing growth-trend protection as a complement to the age mechanism. At the application level, we construct a complete "training-time slimming plus post-processing refinement" lightweighting pipeline, performing distance-aware filtering and QEM mesh simplification on the lightweight Gaussian field output by CP-GEIP to further compress the surface representation. Multi-scene experiments show that CP-GOF effectively compresses Gaussian model size and substantially simplifies the mesh surface representation while keeping the geometric fidelity of the center object visually indistinguishable from the original GOF.

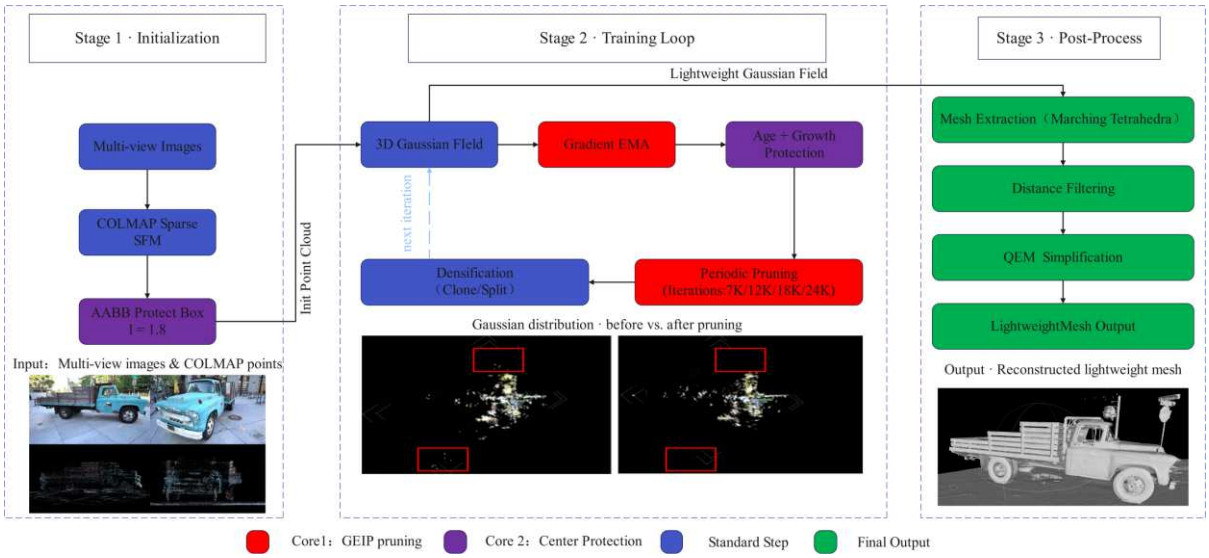


Fig. 1 Overview of the CP-GOF framework

2 Related Work

2.1 3D Gaussian Splatting and Surface Reconstruction

Neural Radiance Fields (NeRF) [17–20] achieve high-quality novel-view synthesis via an implicit volumetric representation, but their MLP-based per-point query mechanism severely limits rendering speed and struggles to meet real-time requirements. 3D Gaussian Splatting (3DGS) [1] replaces the implicit network with explicit anisotropic Gaussians and achieves real-time rendering through efficient rasterization, quickly becoming the dominant paradigm in neural rendering [21]. However, the original 3DGS design optimizes only photometric consistency; its anisotropic Gaussians form ellipsoidal distributions in 3D space and lack explicit constraints on surface geometry, so the Gaussian centers often deviate from the true surface, and directly extracting a mesh produces severe artifacts and floating noise.

To remedy this deficiency, subsequent work has attempted to introduce geometric supervision, but all such methods become trapped in the trade-off between reconstruction accuracy and computational efficiency. SuGaR

[22] binds Gaussians to the mesh surface via a surface-alignment constraint, but its joint optimization framework requires alternately optimizing Gaussian parameters and mesh topology, incurring heavy computational overhead, and its surface-alignment regularization tends to over-smooth high-frequency geometric detail. PGSR [23] introduces planar constraints to improve geometric accuracy in large scenes, but its core assumption—that the local scene can be approximated as planar—systematically fails on curved surfaces, vegetation, and unstructured detail regions, leading to geometric collapse in regions of complex topology. 2D Gaussian Splatting [24] improves depth estimation by projecting 3D Gaussians into 2D disks, but this dimensionality-reducing representation severs the continuity of 3D space; it not only sacrifices the global consistency of depth perception, but also requires its mesh extraction to rely on additional post-processing and struggles to recover complete surface topology.

The essential commonality of the above methods is that they all attempt to "patch" geometric defects within the original 3DGS representation, or rely on discretization methods such as Marching Cubes during mesh extraction, or introduce over-smoothing and topological constraints during regularization—thereby introducing inherent errors such as limited resolution, sharply increasing memory overhead, and loss of surface detail. In sharp contrast, the Gaussian Opacity Fields (GOF) proposed by Yu et al. [8] fundamentally restructure the geometric-supervision paradigm: instead of directly manipulating discrete Gaussians, GOF constructs a continuous opacity field that converts discrete Gaussians into a differentiable implicit surface representation, achieving—for the first time—continuous modeling of the internal geometric structure of the Gaussian field without sacrificing rendering efficiency. Furthermore, GOF introduces a depth-normal consistency constraint that couples the photometric and geometric information of multi-view geometry in a joint optimization, forcing the Gaussian distribution to converge toward the true surface rather than floating as ellipsoids in space. Crucially, GOF employs marching tetrahedra to directly extract high-quality triangular meshes; compared to traditional Marching Cubes, tetrahedral partitioning can adaptively capture complex topological variations and completely avoids the resolution bottleneck and feature loss introduced by voxel-grid discretization. This design enables GOF to achieve simultaneous breakthroughs across three dimensions—surface-reconstruction accuracy, mesh topological integrity, and extraction efficiency—making it the preferred solution and de facto benchmark for directly extracting high-quality meshes from 3D Gaussian fields.

However, while achieving its breakthrough in reconstruction quality, GOF also fully inherits the densification mechanism of 3DGS: the number of Gaussian primitives grows explosively from an initial few tens of thousands to millions through adaptive densification, and the final model storage overhead can reach hundreds of MB. This "accuracy improvement accompanied by volume bloat" paradox is not merely a matter of scale; it exposes a deep flaw in the representational efficiency of GOF: its densification strategy is driven by global photometric error and indiscriminately splits Gaussians in all regions within the view frustum during optimization, causing a large number of primitives to be allocated to the background, sky, ground, and other redundant regions far from the object of interest. For center-object-priority scenarios (such as autonomous driving, industrial inspection, and cultural-heritage digitization), this globally uniform bloat causes severe resource misallocation—the geometric accuracy of the foreground target certainly benefits, but the number of Gaussians in irrelevant background regions grows just as exponentially, forming a heavy "geometric-noise burden."

The consequences of this bloat are multidimensional and mutually coupled: millions of Gaussians not only sharply increase model-storage and network-transmission overhead, but also impose enormous computational pressure during mesh extraction—marching tetrahedra must traverse all Gaussians to build the tetrahedral mesh, and the linear growth in primitive count translates directly into super-linear deterioration of extraction time. Meanwhile, the latency of rasterized rendering also rises linearly with the number of Gaussians, severely constraining the deployment feasibility of GOF on edge devices, in real-time rendering pipelines, and in large-scale scenes. Therefore, achieving adaptive lightweighting of the Gaussian field—that is, matching Gaussian density to geometric importance rather than blindly binding it to photometric error—while preserving the surface-reconstruction quality advantage of GOF has become a key bottleneck for moving GOF from academic research to practical applications, and is the central starting point of this work.

2.2 Gaussian Lightweighting Methods

Existing Gaussian-lightweighting methods can be divided into two categories, post-hoc pruning and training-time structural restructuring, but neither can simultaneously satisfy the two core constraints of "center-object fidelity" and "GOF compatibility."

Post-hoc pruning reduces the number of Gaussians based on global importance ranking after training is complete. LightGaussian [11] computes saliency scores from ray-intersection frequency, Mini-Splatting [12] performs pixel-level pruning based on color-blending weights, RadSplat [13] removes far-field Gaussians via a distance field, and LP-3DGS [25] adopts a learned strategy to predict Gaussian importance. These methods implicitly assume that "rendering contribution is positively correlated with geometric importance." However, in weakly textured regions of the scene's main body (such as building facades, indoor walls, and road surfaces) and weakly observed regions (such as occluded backsides, the reverse sides of mirrors and glass panes, and structural interiors), weak photometric contribution and sparse viewpoint coverage lead to low rendering saliency, and global ranking misjudges these regions as prunable background, directly producing geometric holes. Moreover, post-hoc pruning often requires an additional rendering pass or rasterizer adaptation, incurring high engineering complexity.

Training-time structural restructuring instead attempts to reduce redundancy from the Gaussian representation itself. Scaffold-GS [14] uses an anchor-neural-network hybrid structure, Octree-GS [15] organizes Gaussians hierarchically via an octree, and Compact 3D Gaussian Representation [16] constrains the number of Gaussians via a learnable mask. Although these methods achieve significant compression, GOF surface extraction depends on the opacity-field continuity of explicit Gaussians, and any fundamental modification of the native Gaussian representation (such as anchoring, neural-network-based attributes, or the introduction of non-Gaussian primitives) disrupts the spatial distribution of the opacity field, causing abnormal gradient flow and mesh-extraction failure. Consequently, existing training-time constraint-based methods are difficult to apply directly to the GOF pipeline.

In summary, existing Gaussian-lightweighting methods fall into the dilemma of "pruning harms geometry, restructuring harms compatibility": post-hoc global pruning inevitably sacrifices the geometric integrity of the center object because of the spatial mismatch between rendering contribution and geometric contribution; training-time structural restructuring, by changing the native form of the Gaussian representation, disrupts the continuity of the GOF opacity field and causes high-quality surface extraction to fail. The essence of this dilemma is that existing mainstream methods either perform "after-the-fact trimming" once Gaussian-distribution optimization has converged, or "tamper with the representation structure" during training—and neither achieves the co-evolution of Gaussian count and geometric importance within the training dynamics.

To this end, this paper aims to explore a training-time lightweighting paradigm compatible with the native GOF mechanism—achieving adaptive constraint on the number of Gaussians and explicit protection of center-region geometric fidelity without altering its explicit Gaussian representation and opacity-field framework—thereby breaking the bottleneck in which model lightweighting and high-quality surface extraction cannot be achieved simultaneously.

2.3 Training-Time Control of the Number of Gaussians

The lightweighting methods above—whether post-hoc ranking or training-time masking—all treat Gaussians as statistically stable individuals when making pruning decisions, directly retaining or deleting them based on the current importance measure. However, a large number of newborn Gaussians exist during GOF training (produced by densification via cloning or splitting); with very few historical gradient records, their EMA importance estimates exhibit systematic underestimation and high variance. If such Gaussians directly participate in pruning ranking, the true importance of newborn Gaussians is severely underestimated: those that "should be retained" are erroneously removed (false-kill), while those that "should be deleted" are erroneously retained because of an occasional large-gradient observation (false-keep). Existing work has not analyzed this reliability problem of newborn Gaussians from a statistical perspective, nor provided a minimum age lower bound for pruning decisions, leaving training-time pruning without theoretical guarantees [26–28].

The above analysis shows that existing methods lack theoretical guarantees for the statistical reliability of newborn Gaussians. Starting from EMA bias modeling and confidence-interval derivation, this paper establishes a minimum age lower bound for training-time pruning.

3 Method

3.1 Method Overview

Among existing lightweighting methods, post-hoc pruning (LightGaussian, Mini-Splatting, etc.) requires modifying the CUDA rasterizer or executing an additional global rendering pass, which is incompatible with the opacity-field optimization mechanism of GOF training; training-time structural restructuring (Scaffold-GS, Octree-GS, etc.) changes the explicit Gaussian structure and disrupts the spatial continuity of the opacity field, causing GOF mesh extraction to fail. This paper therefore focuses on training-time pruning that is fully compatible with the original GOF training pipeline, and does not conduct direct comparisons with the aforementioned heterogeneous frameworks.

Our method, denoted CP-GOF, comprises two parts: training-time Gaussian lightweighting and post-processing geometric simplification. The training part embeds a CP-GEIP module into the standard GOF training loop, periodically pruning redundant background Gaussians based on importance while absolutely protecting the center-object region. The post-processing part extracts a mesh from the lightweight Gaussian field obtained after training, and compresses the number of faces via distance filtering and QEM [29] simplification.

3.2 CP-GEIP: Center-Protected Gradient-EMA Importance Pruning

3.2.1 Gradient-EMA Importance Measure

Existing methods typically require an additional rendering pass to compute each Gaussian's contribution to the final image. We observe that, in the standard GOF training loop, `loss.backward()` has already computed the gradient $\partial L/\partial \alpha$ of the opacity parameter α for each Gaussian. This gradient directly reflects the current sensitivity of the Gaussian to the photometric loss: if $|\partial L/\partial \alpha| \approx 0$, then a small perturbation of the Gaussian's opacity has almost no effect on the rendering result, i.e., its contribution to PSNR is extremely low. We therefore define the importance of a Gaussian as the exponential moving average (EMA) of its opacity gradient:

$$I_i(t) = \gamma \cdot I_i(t-1) + (1-\gamma) \cdot |\partial L/\partial \alpha_i| \tag{1}$$

where $\gamma = 0.92$ is the decay coefficient. This measure is updated after each `loss.backward()` with no additional rendering overhead and introduces almost no extra computation. The choice of $\gamma = 0.92$ is consistent with the common configuration for EMA estimation in online model pruning [26, 27]; the effective memory window $1/(1-\gamma) = 12.5$ iterations strikes a balance between response speed and estimation stability.

3.2.2 Sparse-Reconstruction-Prior Spatial Protection Box

Globally ranking Gaussians by I_i and pruning those at the bottom inevitably eliminates geometrically critical Gaussians located in weakly observed or weakly textured regions. Although these regions belong to the center object, repetitive or monotonous textures or sparse viewpoint coverage cause their opacity gradients $|\partial L/\partial \alpha|$ to be low overall, so they are misjudged as having low contribution. This constitutes the systematic misjudgment chain of global pruning methods: weakly textured or weakly observed regions have their importance systematically underestimated because of gradient saturation, are consequently pruned by mistake, and ultimately form geometric holes. For example, in the Truck scene of Tanks and Temples, regions where weak texture and weak observation overlap, such as the wooden planks of the cargo bed, exhibit pronounced holes under Global-GEIP precisely because their gradients are near zero and are misjudged as "background" by global ranking.

We use the initial point-cloud distribution provided by sparse reconstruction (e.g., COLMAP [30]) to automatically compute a 3D axis-aligned protection box (AABB):

$$c = \frac{1}{N}\sum_{j=1}^{N} p_j \in \mathbb{R}^3 \tag{2}$$

$$\sigma_d = \sqrt{\frac{1}{N}\sum_{j=1}^{N}(p_{\mathrm{j,d}} - c_d)^2},\ d \in \{x,y,z\} \tag{3}$$

$$\Omega = \{\, p \in \mathbb{R}^3 \mid |p_d - c_d| \le \lambda\sigma_d,\ d = x,y,z \,\} \tag{4}$$

where p_j is a sparse-reconstruction initial point, and c and σ_d are respectively the center and the per-dimension standard deviation of the initial point cloud. During training-time pruning, any Gaussian whose

center coordinate lies within Ω is marked as protected; regardless of how low its importance I_i is, it does not enter the pruning candidate pool. Fig. 3 illustrates this sparse-reconstruction-prior protection mechanism.

The choice of λ can be determined analytically from the spatial-distribution characteristics of the sparse-reconstruction point cloud. Assuming the initial point cloud approximately follows a Gaussian distribution, the protection-box coverage is $P = 2\Phi(\lambda) - 1$. When $\lambda = 1.8$, the per-dimension coverage reaches $P \approx 2 \times 0.9641 - 1 = 92.8\%$, i.e., the protection box covers about 92.8% of the projected initial points along each dimension. This coverage strikes a balance between "fully containing the center object" and "avoiding over-expansion into the background." It should be noted that the coverage above is the per-dimension projection coverage; the 3D joint coverage is $P_{3D} = P(\lambda)^3 \approx 0.928^3 \approx 80.0\%$, meaning that about 80% of the initial point-cloud 3D coordinates fall entirely within the protection box—a proportion sufficient to fully cover the main body of the center object while avoiding over-expansion into the background. The resulting trade-off is analyzed quantitatively in Fig. 2.

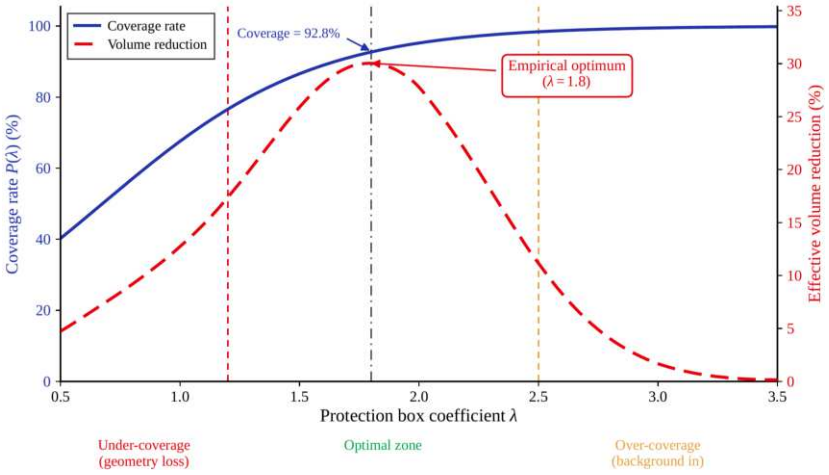


Fig. 2 Trade-off curve between coverage and volume-reduction rate

Fig. 2 reveals the essential trade-off in choosing λ, with $\lambda = 1.8$ as the empirically optimal coefficient. The coverage $P(\lambda) = 2\Phi(\lambda) - 1$ increases monotonically with λ (solid blue line), while the effective volume-reduction rate follows a bell-shaped distribution that first rises and then falls (dashed red line): when λ is too small (< 1.2), the protection box cannot fully cover the edges of the center object (such as wheels and rear-view mirrors and other weakly textured regions), causing geometric damage and holes; when λ is too large (> 2.5), redundant background Gaussians enter the protection box, the prunable candidate pool shrinks sharply, and the volume-reduction rate drops below 12%. At $\lambda = 1.8$, the coverage reaches 92.8% and the volume-reduction rate peaks at about 30%, achieving the empirically optimal balance between "fully containing the center object" and "avoiding over-expansion into the background." Therefore, $\lambda = 1.8$ is adopted as the coefficient of the sparse-reconstruction-prior protection box.

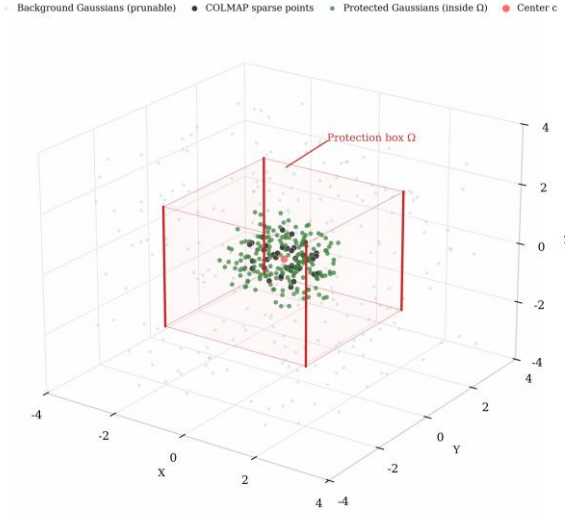


Fig. 3 Illustration of the sparse-reconstruction-prior protection mechanism

3.2.3 Age Awareness and Growth-Trend Protection

In the early stages of their existence, newborn Gaussians have too short a gradient history, and their EMA importance estimates suffer from serious statistical bias. This subsection establishes the theoretical necessity of the age-protection mechanism from three angles—estimation bias, confidence interval, and false-kill rate—and derives the lower bound of the minimum age threshold T_{age}.

It should be noted that Eqs. (6)–(8) characterize the response behavior of the EMA estimator to the gradient sequence. When $\gamma = 0.92$, the EMA weights themselves approach steady state within about 100 iterations ($\gamma^{100} \approx 2.4 \times 10^{-4}$). However, newborn Gaussians face not only the transient bias of the EMA estimator, but also, more critically, the fact that their own parameters (position, covariance, color, opacity) have not yet been sufficiently optimized. In the GOF densification mechanism, a Gaussian receives a structural update (cloning/splitting) only once every 100 iterations, after which it must undergo many gradient-descent iterations before its local geometry and radiance-field distribution stabilize. Therefore, the physical meaning of T_{age} is not to wait for the EMA estimate to converge, but to ensure that a Gaussian undergoes parameter optimization over at least 15 densification cycles (about 1500 iterations), so that its gradient distribution transitions from an "initial transient" to a "statistical steady state." Only then do the gradient samples on which the EMA estimate is based possess stable statistical properties, and only then does the pruning decision become reliable.

(1) EMA Estimation Bias of Newborn Gaussians

Suppose a Gaussian is born at iteration b_i via densification, and its absolute opacity gradient $g_t = |\partial L/\partial \alpha_t|$ is independent and identically distributed during training, with mean μ and variance σ^2. According to the EMA definition in Eq. (1), the importance estimate of this Gaussian at iteration t is:

$$I_i(t) = (1 - \gamma) \sum_{\tau=b_i}^{t} \gamma^{t-\tau} \cdot g_\tau \tag{5}$$

Let $A = t - b_i$ denote the "age" (number of surviving iterations) of this Gaussian. When $A \ll 1/(1-\gamma)$, the EMA has not yet reached steady state, and its expectation and variance can be expressed analytically as:

$$E[I_i] = \mu \cdot (1 - \gamma^A) \tag{6}$$

$$Var(I_i) = \frac{(1-\gamma)^2 \sigma^2}{1-\gamma^2} \cdot (1 - \gamma^{2A}) \tag{7}$$

Key conclusion one: systematic underestimation. When A is small, $\gamma^A \to 1$, so $E[I_i] \to 0$, far below the true expectation μ. This means the importance of newborn Gaussians is systematically underestimated by the EMA mechanism, and the degree of underestimation decays exponentially as age increases.

Key conclusion two: extremely large relative fluctuation. Denoting the relative standard deviation (coefficient of variation) as $\text{CV} = \sqrt{Var(I_i)} \,/\, E[I_i]$ and substituting $\gamma = 0.92$ gives:

$$CV(A) = (\sigma/\mu) \cdot \sqrt{\frac{1-\gamma}{1+\gamma}} \cdot \sqrt{\frac{1-\gamma^{2A}}{(1-\gamma^A)^2}} \tag{8}$$

As $A \to 0$, $\text{CV} \to +\infty$; as A increases, CV decays exponentially, but during the $A < T_{\text{age}}$ stage it remains markedly higher than the steady-state level. This indicates that the EMA estimate of newborn Gaussians fluctuates significantly in the early training stage and is easily misjudged as "zero contribution" because of a single random gradient trough.

(2) Confidence Interval Perspective and Derivation of the T_{age} Lower Bound

The essence of a pruning decision is a hypothesis-testing problem: we need to reject, at confidence level $1 - \delta$, the null hypothesis that "this Gaussian is truly of low contribution." Let the global pruning threshold be I_{thr}; if a Gaussian's true importance is $\mu > \mu_{\text{thr}}$ (an active Gaussian), then its probability of being false-killed is:

$$P_{\text{kill}} = P(\, I_i < I_{\text{thr}} \,|\, \mu > \mu_{\text{thr}} \,) \tag{9}$$

According to the central limit theorem and uncertainty modeling in Bayesian deep learning [31], when A is sufficiently large, I_i approximately follows a normal distribution $N(\mu(1-\gamma^A), \text{Var}(I_i))$. Letting $z_{1-\delta/2}$ be the $1 - \delta/2$ quantile of the standard normal distribution, the minimum age A_{min} that ensures the false-kill probability is no greater than δ satisfies:

$$\mu(1 - \gamma^{A_{\text{min}}}) - z_{1-\delta/2} \cdot \sigma_I \geq I_{\text{thr}} \tag{10}$$

In the GOF training scenario, by collecting the gradient distribution of the first 5K iterations in the Truck scene, we empirically estimate $\sigma/\mu \approx 2.5$; taking $\delta = 0.05$ (95% confidence), $z_{0.975} \approx 1.96$. Since I_{thr} is set at the bottom 35th percentile of the candidate pool, about 35% of the Gaussians have importance below this threshold. According to the statistics of the first-5K-iteration gradient distribution in the Truck scene, the importance corresponding to the bottom 35th percentile is about 0.4 times the global mean μ, so we take $\mu_{\text{thr}} \approx 0.4\mu$. Eq. (10) shows that, under the original gradient distribution with $\sigma/\mu \approx 2.5$, the EMA estimate of newborn Gaussians carries significant uncertainty; meanwhile, GOF performs densification once every 100 iterations, and the Gaussian parameters require many optimization iterations before stabilizing. Combining EMA statistical convergence with the parameter-optimization cycle, we take:

$$A_{\min} \geq 1480 \approx 1500 \tag{11}$$

In Eq. (11), $T_{\text{age}} = 1500$ is a heuristic threshold based on the densification cycle and empirical parameter convergence. This value is about 5% of the total 30K training iterations, coinciding with the early convergence window found in training-time network pruning [28], and ensures that newborn Gaussians undergo at least 15 densification iterations (GOF densifies once every 100 iterations by default), so that their gradient distribution, EMA estimate, and Gaussian parameters all reach engineering stability. If T_{age} is too small, systematic underestimation leads to large-scale false-kills; if T_{age} is too large, the pruning window for redundant background Gaussians is compressed and the reduction rate drops. As shown in the ablation study below, removing age protection leads to severe quality degradation and volume bloat, experimentally verifying the necessity of this lower bound.

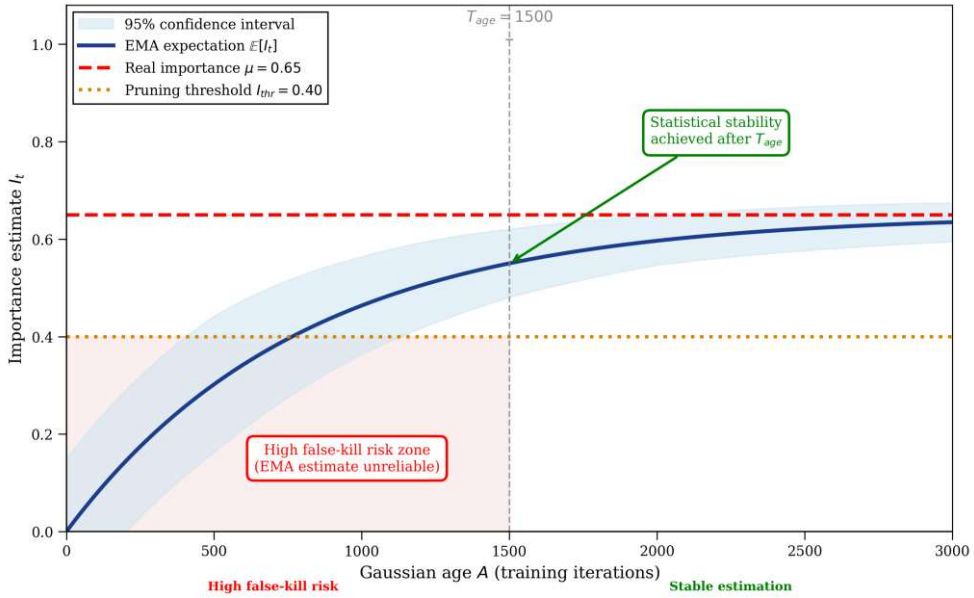


Fig. 4 Illustration of the statistical stability of newborn-Gaussian gradients and the parameter-convergence process

Fig. 4 reveals the systematic false-kill risk that newborn Gaussians face during pruning. Since the EMA importance estimate is initialized with no historical gradient records, it suffers from a severe downward bias: $E[I_t]$ starts from 0 and converges only exponentially toward the true importance μ (the solid blue line), while the estimate remains statistically unstable until the bias has decayed and the variance has approached its steady state (the light-blue 95% confidence interval is widest in the transition region). As a result, during the $A < T_{\text{age}}$ stage, a Gaussian of genuinely high true importance ($\mu = 0.65$) has its estimate pinned below the illustrative pruning threshold I_{thr} — indeed, even the upper bound of its 95% confidence interval stays inside the red false-kill-risk region until $A \approx 700$ — and is thus erroneously removed. CP-GOF sets $T_{\text{age}} = 1500$ as the engineering-stability critical point: at this point the Gaussian parameters have been sufficiently optimized, the overall estimation reliability has clearly exceeded the threshold, and the estimation uncertainty has decayed sufficiently. This threshold integrates EMA statistical convergence with empirical parameter optimization; thereafter the curve enters an asymptotic-convergence plateau, and the Gaussian can safely participate in pruning ranking.

(3) False-Kill-Rate Modeling and Training-Dynamics Collapse

Define the false-kill rate η as the proportion of active Gaussians (with true $\mu > \mu_{\text{thr}}$) erroneously removed within the pruning window. Suppose that at window t, newborn Gaussians ($A < T_{\text{age}}$) account for a fraction ρ of the candidate pool, with their pruning probability given by Eq. (9). Substituting Eqs. (6)–(8) yields the analytical upper bound of the false-kill rate:

$$\eta \approx \rho \cdot \Phi\left(\frac{I_{\text{thr}} - \mu(1-\gamma^A)}{\sigma_I}\right) \tag{12}$$

where $\Phi(\cdot)$ is the standard normal cumulative distribution function. When $T_{\text{age}} = 0$, $\rho \approx 0.35$ (densification continuously produces new Gaussians) and $\gamma^A \to 1$. At this point $E[I_i] = \mu(1-\gamma^A) \to 0$; meanwhile $\text{Var}(I_i) \propto (1-\gamma^{2A}) \to 0$, so $\sigma_I \to 0$. The numerator $I_{\text{thr}} - \mu(1-\gamma^A) \to I_{\text{thr}} > 0$, while the denominator $\sigma_I \to 0$, so the argument of Φ tends to $+\infty$, $\Phi \to 1$, and the false-kill rate $\eta \approx \rho \cdot 1 = \rho \approx 35\%$. This means that in each pruning window, about 35% of the Gaussians in the candidate pool—almost all of them newborn—will be misjudged as "zero contribution" and removed en masse.

Large-scale false-kills trigger a vicious "produce-false-kill-reproduce" cycle: the densification mechanism continuously generates compensatory Gaussians to maintain background coverage, while the pruning module kills them en masse in every window; to fill the background holes left by false-kills, densification further accelerates proliferation, causing a large number of ineffective Gaussians—surviving only a few hundred iterations and never fully converging—to accumulate in background regions. Although numerous, these Gaussians fail to form an effective radiance-field contribution because they lack stable gradient optimization; instead, they disrupt the spatial-distribution balance of the Gaussian field, and their stray radiance propagates through the rasterization gradient into the interior of the protection box, causing the rendering consistency inside the protection box to collapse. This is precisely the root cause of the "volume bloat and quality plunge" phenomenon observed in the ablation study below when age protection is removed. The age-protection mechanism is not a simple conservative strategy, but a key stabilizer that maintains the balance of training dynamics and prevents the accumulation of ineffective Gaussians from suppressing effective ones. Fig. 5 visualizes the effect of the age-protection mechanism on training dynamics.

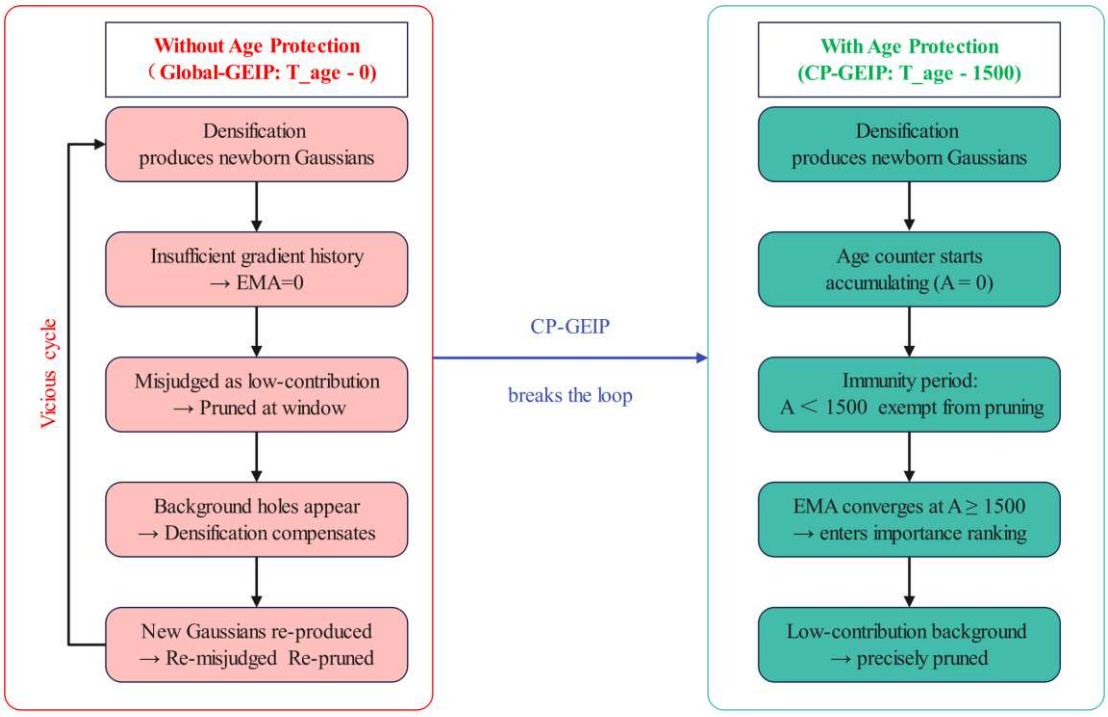


Fig. 5 Effect of the age-protection mechanism on training dynamics

(4) Growth-Trend Protection

The age-protection mechanism solves the "underestimation due to insufficient history" problem of newborn Gaussians, but does not yet cover the misjudgment scenario of mature Gaussians that are "currently underestimated but will be high contribution in the future." If a Gaussian's cumulative increase in I_i over the most recent k iterations, $\Delta I_i = I_i(t) - I_i(t-k) > \theta_{\text{growth}}$, this indicates that it is in a rapid-convergence stage; even if its current absolute value has not reached the threshold, its pruning is deferred to give it sufficient time to converge. This mechanism forms a complementary line of defense with age protection: age protection uses a hard threshold to exclude statistically unreliable newborn Gaussians, while growth-trend protection identifies converging mature Gaussians through dynamic gradient-trend detection. Together, they ensure that the Gaussians in the pruning candidate pool have all passed the dual verification of statistical stability and convergence trend.

3.2.4 Periodic Pruning Schedule

We set up multiple pruning windows in the middle of training (iterations 7000/12000/18000/24000), pruning background Gaussians outside the protection box in ascending order of importance, all at a ratio of 35%. After each pruning, the EMA statistics are reset to prevent stale estimates from contaminating the new set, and recovery time is left for subsequent optimization. It is worth noting that resetting the EMA statistics after each pruning only clears the historical gradient accumulation of surviving Gaussians and does not affect their current geometric parameters (position, covariance, color, etc.). Growth-trend protection (ΔI_i monitoring) recomputes the increment over the most recent k iterations after the EMA reset, so this mechanism remains effective in the post-pruning window and can continue to identify and protect Gaussians in the rapid-convergence stage. The EMA reset and growth-trend protection are complementary: the former prevents the historical bias of stale estimates from contaminating the new set, while the latter ensures that the recent convergence dynamics of Gaussians can still be captured after the reset.

3.3 Integration with the GOF Training Pipeline

CP-GEIP is embedded into the GOF training loop in a non-invasive manner. During initialization, after Gaussians are created, the EMA buffer is started and the protection box is computed from the initial point cloud; in each iteration, the gradient EMA is updated after `loss.backward()`; after densification, the birth iterations of new Gaussians and the dimension of the EMA tensor are synchronized; and during a pruning window, age-aware importance pruning is performed inside and outside the protection box. Here, `filter_3D` is a boolean mask tensor used in the GOF training pipeline for 3D spatial filtering (removing floating Gaussians far from the main surface or of abnormal size), whose dimension corresponds one-to-one to the number of Gaussians. Since CP-GEIP dynamically changes the number of Gaussians during pruning and densification, the dimension of the `filter_3D` tensor must be synchronized accordingly. We explicitly synchronize the slicing and concatenation of `filter_3D` in our `prune_points` and `densification_postfix` implementations, preventing the dimension-mismatch crash caused by the dynamic increase and decrease of Gaussians during training.

3.4 Post-Processing Geometric Simplification

On the lightweight Gaussian field produced by CP-GEIP, we perform standard GOF mesh extraction (marching tetrahedra), followed by two post-processing steps. The first step removes floating fragments far from the main surface with extremely large areas via distance-aware filtering; the second step applies QEM (Quadric Error Metrics) mesh simplification, reducing the number of faces through quadric-error edge collapse while preserving the topology of the center object. Because CP-GEIP has already cleaned up a large number of background floaters during training, the post-processing QEM is less likely to erroneously collapse the center structure, creating a synergistic lightweighting effect from training to output.

It is worth noting that the post-processing QEM simplification currently performs aggressive edge collapse only in the background region, while the center region retains its original topology. This may appear to forgo the compression potential of the center region, but it is in fact a deliberate concession to geometric fidelity: the face density of the center object directly determines the reconstruction accuracy of the surface normals, and any edge collapse may damage sharp features (such as truck-window frames and wheel spokes). Experiments show that, given that CP-GEIP has already cleaned up the background noise, simplifying the background alone can achieve a 60.5% reduction in faces, meeting deployment requirements. Adaptive fidelity-preserving simplification of the center region (such as curvature-aware local QEM) is left as future work.

4 Experiments

4.1 Experimental Setup

The experiments are implemented in the Nerfstudio [32] framework. The dataset uses the Truck scene from Tanks and Temples [33], with COLMAP [30] sparse reconstruction as initialization. The baseline is the official GOF implementation, using the same configuration (full resolution, decoupled appearance model, 30K iterations).

The evaluation metrics include novel-view synthesis quality (Test PSNR/SSIM/LPIPS) and Gaussian model size (MB). For implementation details, training is performed on a single NVIDIA RTX 3090, and the hyperparameters of CP-GEIP are fixed at $\gamma = 0.92, \lambda = 1.8$, and $T_{\text{age}} = 1500$ (see Sect. 3.2).

In addition, the EMA importance update and periodic pruning operations of CP-GEIP introduce extremely low computational overhead: because they directly reuse the gradients already computed during backpropagation, no additional rendering pass is required, and the reduced number of Gaussians after pruning in fact lowers the computational burden of subsequent iterations. Measurements show that the total training time of CP-GEIP on the Truck scene is comparable to, or even slightly shorter than, that of the GOF baseline, incurring no additional time cost from the pruning mechanism.

4.2 Comparison with the GOF Baseline

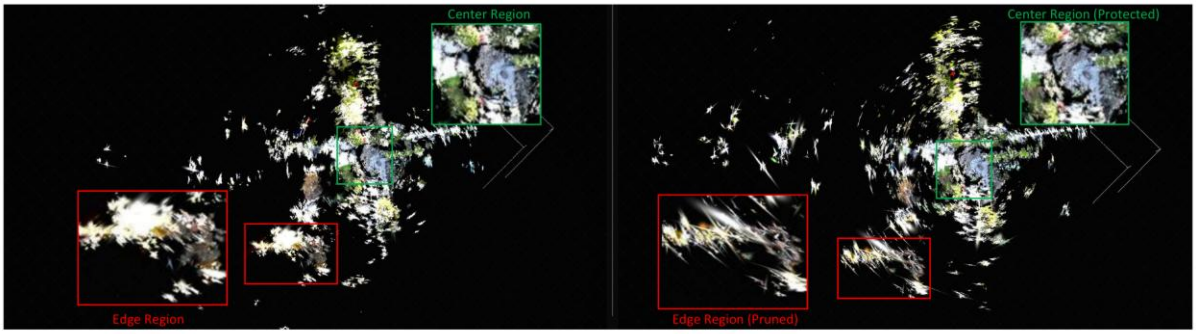


Fig. 6 Visualization of the Gaussian spatial distribution before and after CP-GEIP pruning

Table 1 presents the quantitative comparison between CP-GEIP and the GOF baseline

Table 1 Quantitative comparison between GOF and CP-GEIP

Method	Test PSNR↑	Test SSIM↑	Test LPIPS↓	Gaussian size (MB)↓	Reduction rate
GOF (baseline)	24.14	0.9195	0.1404	563.8	—
CP-GEIP (ours)	24.04	0.9164	0.1460	448.6	20.4%

CP-GEIP performs four periodic prunings in the middle of training (iterations 7K/12K/18K/24K), each pruning 35% of the background Gaussians outside the protection box from lowest to highest importance, while Gaussians inside the protection box never participate in pruning. In the end, the Truck-scene model size is compressed from the GOF baseline's 563.8 MB to 448.6 MB (a 20.4% reduction), with a Test PSNR decrease of only 0.10 dB, consistent with the quantitative results in Table 1. The protection-box mechanism ensures that center-region Gaussians are not pruned during training, laying the foundation for maintaining geometric integrity subsequently; the specific mesh-extraction quality comparison is presented in Sect. 4.4.

4.3 Ablation Study

To verify the effectiveness of the core modules of CP-GEIP, we construct three ablation variants on the Truck scene. All variants keep the same training configuration and control variables by toggling specific

modules only. The quantitative results are reported in Table 2, and the mesh geometric integrity of each variant is compared in Fig. 7.

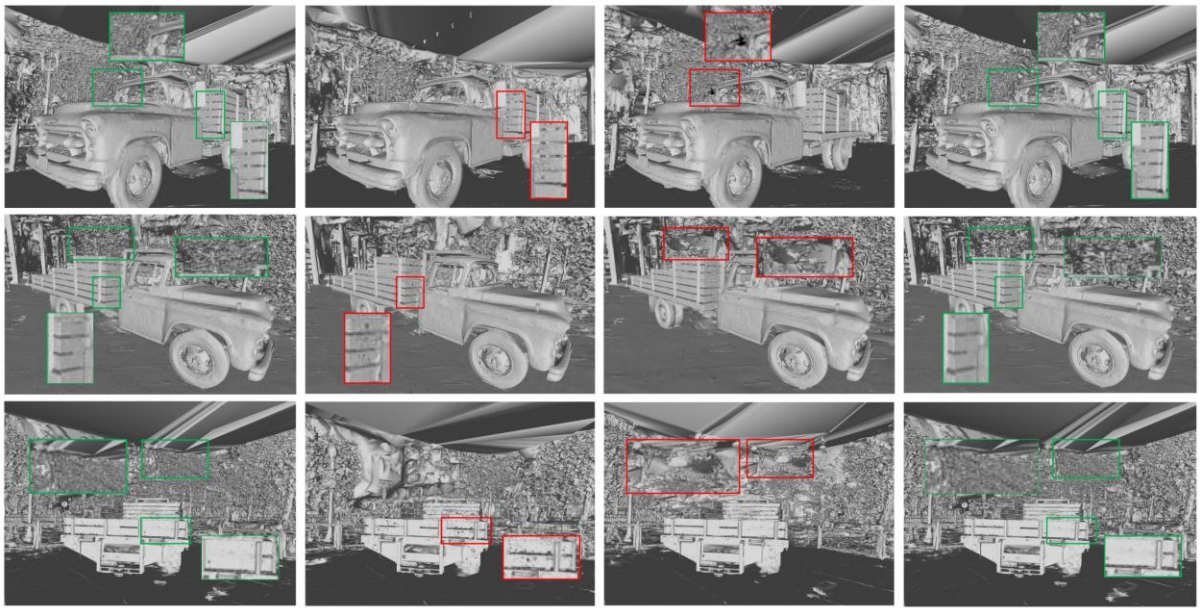


Fig. 7 **Ablation study – impact of each core module on mesh geometric integrity**

Table 2 **Ablation study – impact of each module on rendering quality and model size**

Variant	Protection box	Age protection	Test PSNR↑	Test SSIM↑	Test LPIPS↓	Gaussian size (MB)↓	Mesh quality
GOF baseline	—	—	24.14	0.9195	0.1404	563.8	Intact
Global-GEIP	✗	✓	23.33	0.9003	0.1734	189.4	Holes/breaks
CP-GEIP w/o age	✓	✗	22.52	0.8870	0.1735	496.4	Degraded/damaged
CP-GEIP (Ours)	✓	✓	24.04	0.9164	0.1460	448.6	Intact

It is worth noting that growth-trend protection, as a complementary line of defense to age protection, is enabled by default in all pruning variants, and together with age protection constitutes a complete statistical-reliability guarantee. The ablation results reveal a phenomenon easily overlooked in the field of Gaussian pruning: metrics based on global rendering contribution are extremely insensitive to the geometric integrity of weakly observed regions. Global-GEIP serves as an extreme ablation baseline used to quantitatively verify the consequences of completely removing spatial protection, rather than as a deployable solution. After Global-GEIP removes spatial protection, the model size is compressed to 189.4 MB (a 66.4% reduction), but Test PSNR drops 0.81 dB relative to the original, and significant holes and surface breaks appear in the weakly observed regions of the center object—directly proving the irreplaceable role of the spatial protection box in maintaining center-geometry integrity.

The ablation study further verifies the effectiveness of the theoretical analysis in Sect. 3.2.3: when the minimum age limit T_{age} is reduced from 1500 iterations to 0, Test PSNR plunges to 22.52 dB, a drop of 1.62 dB relative to the original and the worst rendering quality among all variants; moreover, its Gaussian file size, rather than decreasing, actually increases relative to the full CP-GEIP configuration (448.6 MB), reaching 496.4 MB. As stated in Sect. 3.2.3, the gradient EMA of newborn Gaussians is nearly zero in their early life; without age protection, every pruning window misjudges them as "zero contribution" and kills large numbers of them, while GOF's densification mechanism continues to produce compensatory Gaussians, causing the background region to fall into a "produce-false-kill-reproduce" vicious cycle. The accumulation of large numbers of ineffective Gaussians disrupts the spatial-distribution balance of the Gaussian field and ultimately drags down even the rendering consistency inside the protection box. This result shows that age protection is not a simple

conservative strategy, but a key mechanism that maintains the balance of training dynamics and prevents the accumulation of ineffective Gaussians from suppressing effective ones.

Through its three-layer synergistic design comprising a spatial protection box that demarcates the geometric safety boundary, age protection that maintains training stability, and growth-trend protection that preserves convergence potential, the full CP-GEIP configuration reduces the Gaussian model size by 20.4% with only a 0.10 dB PSNR loss, while the mesh quality of the center object remains consistent with the original GOF.

4.4 Mesh Quality and Lightweighting Analysis

Following the evaluation convention of existing Gaussian surface-reconstruction work [22–24], and because real scenes lack ground-truth geometry, we assess mesh quality via visual consistency, topological integrity, and face-count statistics. Fig. 8 shows the comparison of lightweighting effects under different configurations in the Truck scene, and Table 3 reports the corresponding quantitative statistics; the background face density is significantly reduced while the center-region topology remains consistent.

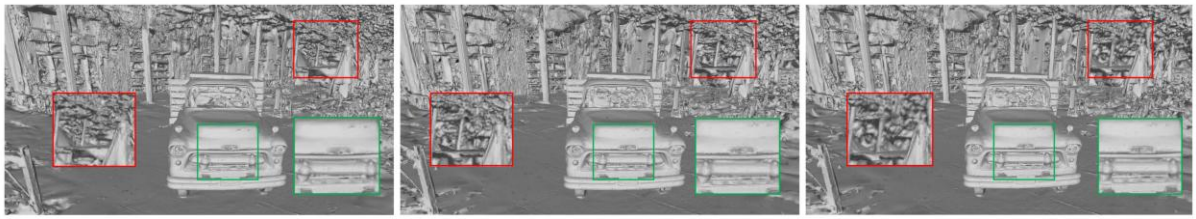


Fig. 8 Comparison of mesh lightweighting effects between the center object and the background

Table 3 Comparison of mesh lightweighting in the Truck scene

Method	Faces↓	Vertices↓	Mesh size (MB)↓	Face-reduction rate	File-reduction rate	Remarks
GOF baseline	49.24M	24.49M	890.0	—	—	Original mesh
CP-GEIP	38.12M	19.00M	690.0	−22.6%	−22.5%	Extracted after training-time pruning
CP-GOF (+QEM)	19.47M	9.61M	351.0	−60.5%	−60.6%	Final lightweight result

The background face density decreases significantly from GOF to CP-GOF, while the topology of the center region (the truck) remains consistent. CP-GOF and GOF have comparable geometric consistency on the truck body, with no new holes introduced. Fig. 6 further shows the comparison of the Gaussian spatial distribution before and after CP-GEIP pruning; it can be seen that the Gaussian density in the background region is significantly reduced, while the center region remains dense. After applying QEM simplification to the lightweight Gaussian field output by CP-GEIP, the final mesh face count is further reduced, and the simplification robustness is superior to performing it directly on the original GOF field—the reduction of background noise lowers the risk of QEM erroneously collapsing the center structure.

It is worth noting that performing post-processing simplification directly on the original GOF field can compress the number of faces to some extent, but because its background noise has not been cleaned up during training, QEM struggles to simplify aggressively without harming the center structure. CP-GOF's "training-time pruning + post-processing" synergistic strategy reduces the face count from 49.24M to 19.47M, a total reduction rate of 60.5%, avoiding this problem. This proves that cleaning background noise by CP-GEIP during training is a critical prerequisite for the subsequent QEM simplification: the large number of background floaters on the original GOF field would interfere with QEM's quadric-error metric, making it difficult to perform aggressive edge collapse on the background region without harming the center structure.

4.5 Multi-Scene Validation

To verify cross-scene generalization, we conduct supplementary experiments on the Mip-NeRF 360 dataset [18], focusing on the consistency of CP-GEIP's pruning behavior across different scene structures (outdoor natural scenes, outdoor courtyard scenes, indoor scenes with a central object, and indoor complex scenes).

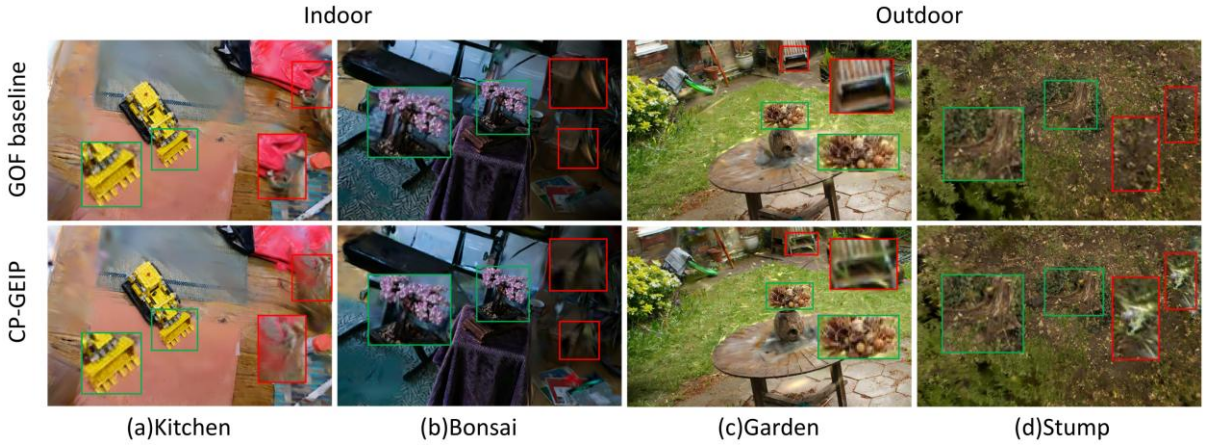


Fig. 9 Comparison of rendering quality across multiple scenes on Mip-NeRF 360

As shown in Fig. 9, the rendering quality of CP-GEIP across the four scenes remains consistent with the GOF baseline, and no texture degradation or color distortion caused by pruning is observed. The main body of the toy loader and the surrounding ground in the kitchen scene, the bonsai plant and tabletop objects in the bonsai scene, the round table and courtyard vegetation in the garden scene, and the tree-stump main body and ground details in the stump scene are all fully preserved.

Table 4 Quantitative comparison on the Mip-NeRF 360 dataset

Scene	Method	PSNR↑	ΔPSNR	Gaussian size (MB)↓	Gaussian reduction	Mesh size (MB)↓	Mesh reduction
stump	GOF	29.41	—	967.4	—	—*	—
	CP-GOF	29.40	−0.01	710.4	26.60%	468	N/A
garden	GOF	26.58	—	521.8	—	726	—
	CP-GOF	26.57	−0.01	431.2	17.40%	334	54.00%
bonsai	GOF	29.10	—	194.4	—	230.1	—
	CP-GOF	29.04	−0.06	166.2	14.50%	92.9	59.60%
kitchen	GOF	28.81	—	203.9	—	239.2	—
	CP-GOF	28.67	−0.14	137.3	32.70%	139.5	41.70%

***GOF baseline failed to complete mesh extraction due to out-of-memory during marching tetrahedra**

As shown in Fig. 10, CP-GOF maintains the geometric integrity of the center object across all four scenes. In the stump scene, the GOF baseline caused an out-of-memory error during the marching-tetrahedra extraction stage because the Gaussian model size was too large (967.4 MB) and failed to complete mesh extraction; CP-GOF completed extraction successfully after lightweighting (710.4 MB). In the kitchen, bonsai, and garden scenes, the background face density of CP-GOF is significantly reduced, while the topological structure of the center object remains consistent with the GOF baseline, with no new holes or breaks introduced.

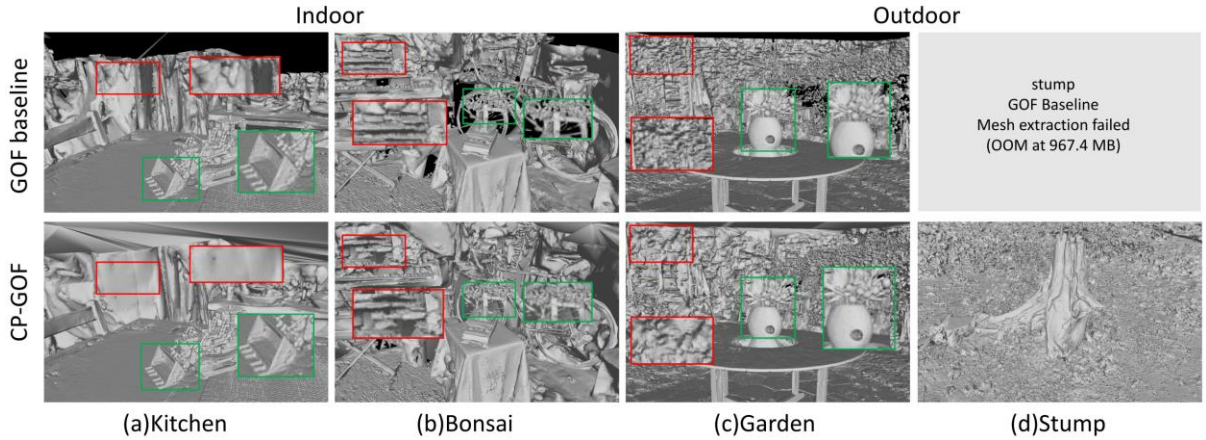


Fig. 10 Comparison of cross-scene mesh-extraction quality on Mip-NeRF 360

The quantitative results show that the PSNR loss of CP-GOF is kept within the range of 0.01 dB to 0.14 dB across all four scenes, the Gaussian-size reduction rate reaches 14.5% to 32.7%, and in the scenes where mesh extraction can be completed, the mesh file size is reduced by 41.7% to 59.6%. The differences in reduction rate across scenes reflect the adaptive nature of CP-GEIP: in the stump and kitchen scenes, a large number of low-contribution redundant Gaussians exist outside the protection box and the prunable space in the background region is ample, so the reduction rate is higher (26.6%, 32.7%); in the garden and bonsai scenes, the background region contains rich vegetation and furniture detail, and the background Gaussians themselves have high geometric and rendering importance, so the prunable space is relatively small (17.4%, 14.5%), yet the PSNR loss still remains at an extremely low level. These results consistently show that CP-GOF can automatically trade off between a high pruning rate and low quality loss according to scene complexity, and does not sacrifice reconstruction quality in pursuit of a high reduction rate.

In addition, the mesh-extraction results in Table 4 reveal the extra value of CP-GOF at the engineering-deployment level: the GOF baseline suffers from OOM during mesh extraction in the stump scene due to the excessive number of Gaussians, whereas the lightweight Gaussian field of CP-GOF can complete extraction smoothly. This demonstrates that training-time pruning not only compresses the storage size, but also extends the feasibility of GOF mesh extraction to scenes that were previously unprocessable due to memory limits, significantly improving the applicability of the method in memory-constrained environments.

5 Conclusion

To address the problem of Gaussian-count bloat during GOF training, this paper proposes the CP-GOF framework, which is fully compatible with the original training pipeline. It solves the statistical-reliability problem of “when to prune” through an opacity-gradient EMA importance measure and a Gaussian-age mechanism, solves the center-object-integrity problem of “where to prune” through a COLMAP-prior protection box, and solves the dynamic-balance problem of “what to preserve” through the synergistic design of age protection and growth-trend protection, thereby unifying aggressive background pruning with absolute preservation of the center object.

Experiments show that CP-GOF compresses the Gaussian model size by 20.4% and reduces the mesh face count by 60.5% with a PSNR loss of only 0.10 dB, while keeping the topological structure of the center object consistent with the original GOF; in the multi-scene validation on Mip-NeRF 360, the PSNR loss is kept within 0.14 dB and the size reduction rate reaches 14.5% to 32.7%.

The limitations of our method are that the protection box relies on the COLMAP initial point-cloud distribution and its boundary may be ambiguous in extreme scenes where the center object and background heavily overlap, and that the age threshold is currently a fixed empirical value whose adaptive-adjustment strategy remains to be explored. Future work will focus on dynamic optimization of the protection box, lightweighting extensions to 4D Gaussian fields [34], and the transfer of the age-protection framework to other 3DGS variants.

Statements and Declarations

Funding No funding was received for conducting this study.

Competing interests The authors have no relevant financial or non-financial interests to disclose.

Data availability The public datasets analyzed in this study are cited in the References. The source code is available from the corresponding author on reasonable request.

Author contributions Yudai Wang: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Writing – original draft, Visualization. Dekang Yao: Software, Validation, Data curation, Writing – review & editing, Visualization. Liyan Qin: Methodology, Formal analysis, Writing – review & editing, Supervision. Yingao Liu: Software, Validation, Investigation, Writing – review & editing. Fanggui Cai: Formal analysis, Writing – review & editing, Visualization. Zhiying Cui: Resources, Writing – review & editing, Funding acquisition. Zan Wang: Conceptualization, Methodology, Supervision, Project administration, Funding acquisition, Writing – review & editing, Resources.

References

[1] Kerbl B, Kopanas G, Leimkühler T, Drettakis G: 3D Gaussian Splatting for Real-Time Radiance Field Rendering. ACM Trans. Graph. 42(4), 139 (2023). https://doi.org/10.1145/3592433

[2] Müller T, Evans A, Schied C, Keller A: Instant Neural Graphics Primitives with a Multiresolution Hash Encoding. ACM Trans. Graph. 41(4), 1–15 (2022). https://doi.org/10.1145/3528223.3530127

[3] Yu A, Ye V, Tancik M, Kanazawa A: Plenoxels: Radiance Fields without Neural Networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5501–5510 (2022). https://doi.org/10.1109/CVPR52688.2022.00533

[4] Chen Z, Funkhouser T, Hedman P, Tagliasacchi A: MobileNeRF: Exploiting the Polygon Rasterization Pipeline for Efficient Neural Field Rendering on Mobile Architectures. In: ACM SIGGRAPH 2023 Conference Proceedings, pp. 1–9 (2023). https://doi.org/10.1145/3588432.3591547

[5] Girish S, Gupta K, Shrivastava A, Chen P, Garg R, Krishna KM, Jampani V: EAGLES: Efficient Accelerated 3D Gaussians with Lightweight Encodings. In: Proceedings of the European Conference on Computer Vision, pp. 54–71 (2024). https://doi.org/10.1007/978-3-031-73045-0_4

[6] Ye M, Danelljan M, Yu F, Gu L: GaussianPro: 3D Gaussian Processing with Progressive Propagation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 20184–20193 (2024). https://doi.org/10.1109/CVPR52733.2024.01900

[7] Reiser C, Peng S, Liao Y, Geiger A: MERF: Memory-Efficient Radiance Fields for Real-time View Synthesis in Unbounded Scenes. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 12750–12759 (2023). https://doi.org/10.1109/CVPR52729.2023.01226

[8] Yu Z, Chen A, Antic B, Peng S, Bhattacharya A, Niemeyer M, Tang S, Geiger A, Wetzstein G: Gaussian Opacity Fields: Efficient and Compact Surface Reconstruction in Unbounded Scenes. ACM Trans. Graph. 43(4), 1–16 (2024). https://doi.org/10.1145/3658140

[9] Wang P, Liu L, Liu Y, Theobalt C, Komura T, Wang W: NeuS: Learning Neural Implicit Surfaces by Volume Rendering for Multi-view Reconstruction. In: Proceedings of the 35th International Conference on Neural Information Processing Systems, pp. 27171–27183 (2021)

[10] Wang Q, Wang Z, Genova K, Srinivasan PP, Zhou H, Barron JT, Martin-Brualla R, Snavely N, Funkhouser T: Neuralangelo: High-Fidelity Neural Surface Reconstruction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 8456–8465 (2023). https://doi.org/10.1109/CVPR52729.2023.00817

[11] Fan Z, Wang K, Wen K, Zhu Z, Xu D, Wang Z: LightGaussian: Unbounded 3D Gaussian Compression with 15x Reduction and 200+ FPS. In: Advances in Neural Information Processing Systems, vol. 37, pp. 140138–140158 (2024). https://doi.org/10.52202/079017-3891

[12] Fang G, Wang B: Mini-Splatting: Representing Scenes with a Constrained Number of Gaussians. In: Proceedings of the European Conference on Computer Vision, pp. 165–181 (2024). https://doi.org/10.1007/978-3-031-73045-0_10

[13] Niemeyer M, Manhardt F, Rakotosaona MJ, Tschernezki V, Tombari F: RadSplat: Radiance Field-Informed Gaussian Splatting for Robust Real-Time Rendering with 900+ FPS. In: Proceedings of the International Conference on 3D Vision, pp. 134–144 (2025). https://doi.org/10.1109/3DV66043.2025.00018

[14] Lu T, Yu M, Xu L, Xiangli Y, Wang C, Lin D, Dai B: Scaffold-GS: Structured 3D Gaussians for View-Adaptive Rendering. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 20654–20664 (2024). https://doi.org/10.1109/CVPR52733.2024.01982

[15] Ren K, Jiang L, Lu T, Yu M, Xu L, Xiangli Y, Dai B: Octree-GS: Towards Consistent Real-time Rendering with LOD-Structured 3D Gaussians. IEEE Trans. Pattern Anal. Mach. Intell. 1–15 (2025). https://doi.org/10.1109/TPAMI.2025.3568201

[16] Lee JC, Rho D, Sun X, Ko JH, Park E: Compact 3D Gaussian Representation for Radiance Field. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 21719–21728 (2024). https://doi.org/10.1109/CVPR52733.2024.02049

[17] Mildenhall B, Srinivasan PP, Tancik M, Barron JT, Ramamoorthi R, Ng R: NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. In: Proceedings of the European Conference on Computer Vision, pp. 405–421 (2020). https://doi.org/10.1007/978-3-030-58452-8_24

[18] Barron JT, Mildenhall B, Verbin D, Srinivasan PP, Hedman P: Mip-NeRF 360: Unbounded Anti-Aliased Neural Radiance Fields. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5470–5479 (2022). https://doi.org/10.1109/CVPR52688.2022.00539

[19] Martin-Brualla R, Radwan N, Sajjadi MSM, Barron JT, Dosovitskiy A, Duckworth D: NeRF in the Wild: Neural Radiance Fields for Unconstrained Photo Collections. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 7210–7219 (2021). https://doi.org/10.1109/CVPR46437.2021.00713

[20] Mildenhall B, Hedman P, Martin-Brualla R, Srinivasan PP, Barron JT: NeRF in the Dark: High Dynamic Range View Synthesis from Noisy Raw Images. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 16190–16199 (2022). https://doi.org/10.1109/CVPR52688.2022.01571

[21] Jiang Y, Tu J, Liu Y, Gao X, Long X, Wang W, Ma Y: GaussianShader: 3D Gaussian Splatting with Shading Functions for Reflective Surfaces. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5322–5332 (2024). https://doi.org/10.1109/CVPR52733.2024.00509

[22] Guédon A, Lepetit V: SuGaR: Surface-Aligned Gaussian Splatting for Efficient 3D Mesh Reconstruction and High-quality Mesh Rendering. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5354–5363 (2024). https://doi.org/10.1109/CVPR52733.2024.00512

[23] Chen D, Li H, Ye W, Wang Y, Xie W, Zhang S: PGSR: Planar-based Gaussian Splatting for Efficient and High-fidelity Surface Reconstruction. IEEE Trans. Vis. Comput. Graph. 31(9), 6100–6111 (2025). https://doi.org/10.1109/TVCG.2024.3494046

[24] Huang B, Yu Z, Chen A, Geiger A, Gao S: 2D Gaussian Splatting for Geometrically Accurate Radiance Fields. In: ACM SIGGRAPH 2024 Conference Papers, pp. 1–11 (2024). https://doi.org/10.1145/3641519.3657428

[25] Zhang Z, Song T, Lee Y, Yang H, Kim J: LP-3DGS: Learning to Prune 3D Gaussian Splatting. In: Advances in Neural Information Processing Systems, vol. 37, pp. 122434–122457 (2024). https://doi.org/10.52202/079017-3891

[26] He Y, Zhang X, Sun J: Channel Pruning for Accelerating Very Deep Neural Networks. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 1389–1397 (2017). https://doi.org/10.1109/ICCV.2017.155

[27] Zhu M, Gupta S: To Prune, or Not to Prune: Exploring the Efficacy of Pruning for Model Compression. In: Proceedings of the 6th International Conference on Learning Representations (2018). https://openreview.net/forum?id=Sy1iIDkPM

[28] You H, Li C, Xu P, Fu Y, Wang Y, Chen X, Baraniuk RG, Wang Z, Lin Y: Drawing Early-Bird Tickets: Toward More Efficient Training of Deep Networks. In: Proceedings of the 34th International Conference on Neural Information Processing Systems, pp. 10549–10560 (2020)

[29] Garland M, Heckbert PS: Surface Simplification Using Quadric Error Metrics. In: Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques, pp. 209–216 (1997). https://doi.org/10.1145/258734.258849

[30] Schönberger JL, Frahm JM: Structure-from-Motion Revisited. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4104–4113 (2016). https://doi.org/10.1109/CVPR.2016.445

[31] Kendall A, Gal Y: What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision? In: Proceedings of the 31st International Conference on Neural Information Processing Systems, pp. 5574–5584 (2017)

[32] Tancik M, Weber E, Ng E, Li R, Yi B, Kerr J, Wang T, Kristoffersen A, Austin J, Salahi K, Ahuja A, McAllister D, Kanazawa A: Nerfstudio: A Modular Framework for Neural Radiance Field Development. In: ACM SIGGRAPH 2023 Conference Proceedings, pp. 1–12 (2023). https://doi.org/10.1145/3588432.3591516

[33] Knapitsch A, Park J, Zhou QY, Koltun V: Tanks and Temples: Benchmarking Large-scale Scene Reconstruction. ACM Trans. Graph. 36(4), 1–13 (2017). https://doi.org/10.1145/3072959.3073599

[34] Wu G, Yi T, Fang J, Xie L, Zhang X, Wei W, Liu W, Tian Q, Wang X: 4D Gaussian Splatting for Real-Time Dynamic Scene Rendering. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 20110–20120 (2024). https://doi.org/10.1109/CVPR52733.2024.01897