AnchorSplat: Fast and Structure Consistent Detail Synthesis for Gaussian Splatting

Dexu Zhu[1*], Jiangnan Shao[1,2,3*], Xiaofeng Wang[2,4], Junxian Duan[1], Jie Cao[1†], Zheng Zhu[2], and Huaibo Huang[1]

[1] MAIS&NLPR, CASIA, Beijing, China
[2] GigaAI, Beijing, China
[3] ShanghaiTech University, Shanghai, China
[4] Tsinghua University, Beijing, China
dexu.zhu@cripac.ia.ac.cn

Abstract. 3D Gaussian Splatting (3DGS) has emerged as a powerful representation for high-fidelity rendering. However, existing assets often suffer from quality bottlenecks such as missing details and texture noise. Prior attempts to enhance these assets via 2D image processing introduce multi-view inconsistencies and high computational costs. In this paper, we propose a novel 3D-native refinement paradigm named **AnchorSplat**. AnchorSplat is an end-to-end deep network operating directly on 3D structures, avoiding the expensive optimization overhead of traditional 3D-2D-3D pipelines. Crucially, AnchorSplat is a strictly source-free solution requiring no original multi-view images. Central to the proposed method is the Point Anchor Mechanism, which enforces geometric consistency via local offset constraints, mitigating ill-posed mapping and gradient confounding. Furthermore, AnchorSplat replaces iterative densification with a single-pass multiplication mechanism. To facilitate research, we construct 3DGS-SR, the first large-scale benchmark for this task. Experiments demonstrate state-of-the-art results on the 3DGS-SR dataset, with throughput **up to 10^5 times faster** than optimization methods. Notably, AnchorSplat exhibits robust **zero-shot generalization** across diverse data distributions, including generative model outputs and real-world scans. The repository is available at: github

Keywords: 3D Gaussian Splatting · 3D Super-Resolution · 3D Asset Enhancement

1 Introduction

3D Gaussian Splatting (3DGS) [9, 11, 29, 31], despite its success in high-fidelity rendering, suffers from significant quality degradation, including sparse geometry and missing details, when reconstructed from Low-Resolution (LR) inputs. To overcome this limitation, the task of 3D Super-Resolution (3DSR) has

* Equal contribution.
† Corresponding author.

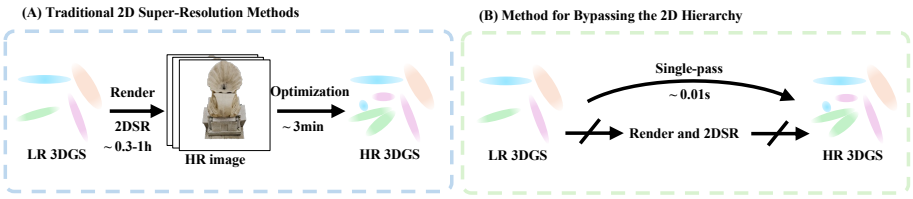

Fig. 1: Comparison of 3DSR paradigms. (A) depicts the conventional 2D-centric pipeline: the process requires rendering the asset to LR images, applying 2DSR, and then performing 3D reconstruction. This multi-step process is computationally expensive. (B) shows our novel 3D native paradigm: by directly processing the 3D input, we bypass the costly intermediate 2DSR step, thus achieving significantly higher throughput and full utilization of 3D geometry.

emerged, which aims to enhance these low-quality assets into High-Resolution (HR), detail-rich counterparts. Addressing this quality degradation is critical, as these deficiencies severely restrict applicability in detail-demanding applications.

Prior works [6,13,27] at 3DSR primarily rely on 2D image-level post-processing. This paradigm does not directly leverage the structured 3D geometric information, leading to critical limitations. First, processing isolated 2D images fails to enforce strict 3D consistency, introducing multi-view artifacts. Second, the required 2D Super-Resolution (2DSR) and subsequent 3D re-optimization incur significant computational overhead. Furthermore, these pipelines are inherently source-dependent, requiring original multi-view images to supervise the re-optimization. This renders them unusable in highly practical source-free scenarios where only the raw 3D asset is available, such as enhancing the unconstrained outputs from recent feed-forward 3D generative models.

The logical alternative to these 2D-centric methods is a 3D-native paradigm. However, while recent works [1, 36, 38, 42, 45] do manipulate 3DGS structures, they focus on different tasks, such as generation or view generalization, and do not address the specific problem of enhancing existing, low-quality assets.

Consequently, a critical research gap persists for a solution that is both 3D native and designed for enhancement. Motivated by this specific gap, we propose AnchorSplat, a framework that rapidly enhances 3DGS asset quality. We pursue a novel high-efficiency technical path: operating directly on the asset structure in the 3D space. This 3D native approach, shown in Fig. 1 (B), is intended to avoid the multi-view inconsistencies and expensive optimization overhead associated with the conventional 2D-centric processing pipelines, which are depicted in Fig. 1 (A). Our architecture is designed as an efficient, single-pass feed-forward network. It first processes the low-quality 3DGS asset as an attribute-rich 3D point cloud. We then employ an efficient point cloud encoder to aggregate local context, extracting a high-dimensional feature descriptor for each individual input Gaussian primitive. This design, which relies on decoding these per-point features, faces two fundamental challenges.

First is the ill-posed mapping problem stemming from the unstructured, point cloud nature of 3DGS. Using a per-point feature to generate an unconstrained new primitive leads to severe gradient confounding, as it can receive conflicting signals from arbitrary views and locations. To resolve this, we introduce the Point Anchor Mechanism. This mechanism leverages the existing coarse geometry of the input as a reliable foundation. It then uses the feature extracted from each anchor to generate new detail primitives only within the immediate vicinity of that anchor, enforcing a strict local constraint that preserves the valid structure and mitigates the mapping ambiguity.

Second, this single-pass encoder design is incompatible with the iteration-based densification of native 3DGS, namely cloning and splitting. To replace this, AnchorSplat uses the same per-point features to drive a controlled multiplicative generation mechanism, named Equivalent Densification Mechanism. This mechanism achieves equivalent density control. It substitutes for cloning by multiplicatively generating new primitives, and substitutes for pruning by learning vanishing opacity.

Experimental results demonstrate that AnchorSplat achieves significant improvements in both performance and efficiency. We evaluate our method on the newly constructed 3DGS-SR benchmark, where it achieves state-of-the-art fidelity while demonstrating a processing throughput increase of up to 10^5 times over conventional optimization methods. Crucially, AnchorSplat exhibits remarkable zero-shot generalization capabilities in highly practical, source-free scenarios. Without any fine-tuning or access to original multi-view images, our model robustly enhances unconstrained assets generated by 3D large generative models as well as noisy real-world object captures. This confirms that AnchorSplat learns dataset-agnostic geometric priors, making it a universal and highly efficient detail enhancer for the broader 3D asset ecosystem.

Overall, our contributions can be summarized as follows:

- We propose AnchorSplat, a novel 3D-native, feed-forward refinement framework that achieves a $10^5\times$ speedup in processing throughput compared to traditional optimization-based enhancement methods.
- We introduce the Point Anchor Mechanism to resolve the gradient confounding inherent in unconstrained 3D prediction, and an Equivalent Densification Mechanism as a single-pass substitute for iterative 3DGS optimization.
- We construct 3DGS-SR, the first large-scale benchmark specifically designed for 3DGS asset enhancement. AnchorSplat achieves state-of-the-art results on this benchmark and demonstrates strong zero-shot generalization to unconstrained AIGC assets and real-world captures under strict source-free conditions.

2 Related Work

2.1 Novel View Synthesis

Novel View Synthesis (NVS) aims to synthesize photorealistic images from unseen viewpoints given a set of input images. Neural Radiance Fields (NeRF) [18,

21] learn continuous implicit scene representations with MLPs and render them via volumetric rendering. In contrast, 3D Gaussian Splatting (3DGS) [11] represents scenes explicitly with anisotropic Gaussians and enables real-time rendering through differentiable rasterization while maintaining high visual quality. Despite these advances, both implicit and explicit representations remain highly dependent on input quality. When reconstructed from sparse or low-resolution images, 3DGS assets often suffer from sparse geometry, missing details, and texture noise.

2.2 3D Super-Resolution

Existing 3D super-resolution methods can be broadly categorized into reconstruction from scratch and asset enhancement. Reconstruction-based methods directly optimize high-resolution 3D representations from low-resolution multi-view inputs [6, 13, 27, 40], often leveraging 2D image priors [10, 12, 15]. For example, SRGS [6] uses SISR models [2,17,24] to generate high-resolution pseudo-labels with a sub-pixel constraint, while SuperGS [25] further incorporates variational residual features. Other methods exploit VSR priors [19,35,37,44]; for instance, Sequence Matters [13] treats multi-view images as video-like sequences [3, 46] to improve temporal and spatial coherence. However, such 2D-to-3D strategies remain limited by the mismatch between 2D enhancement priors and the multi-view consistency required by 3D scenes. Asset enhancement methods instead start from an existing low-quality 3D asset and upgrade it to a higher-fidelity representation. Existing approaches [8, 34] typically follow a 3D-2D-3D pipeline: rendering the coarse 3DGS asset into image sequences, enhancing them with 2D or video super-resolution models, and re-optimizing the 3D representation. Although this pipeline can improve visual details, it introduces additional rendering and reconstruction stages, increasing computational cost and potentially causing multi-view artifacts. In contrast, our method performs enhancement entirely in the 3D domain. It directly takes a low-resolution 3DGS asset as input and outputs a refined Gaussian representation without intermediate 2D conversion or re-optimization. This design avoids 2D-3D domain mismatch, better preserves structural consistency, and enables substantially higher throughput.

2.3 3D Point Cloud

Point clouds [4, 20, 26, 41] serve as the foundational representation for 3D geometry. Early works such as PointNet [22] and PointNet++ [23] introduced the idea of directly processing point clouds via MLP and local aggregation. In recent years, Transformer-based models [5, 32, 33, 43, 47] have achieved state-of-the-art performance across various 3D tasks.

Our work builds upon the profound insights derived from these point cloud networks. At its core, 3DGS is an attribute-rich, unstructured 3D point cloud. PTv3 [7,32], with its simplified architecture and efficient attention mechanism, is highly suitable for modeling 3DGS primitives. While most prior methods focus on tasks like classification or segmentation [7,16,30], our work targets 3DSR. We

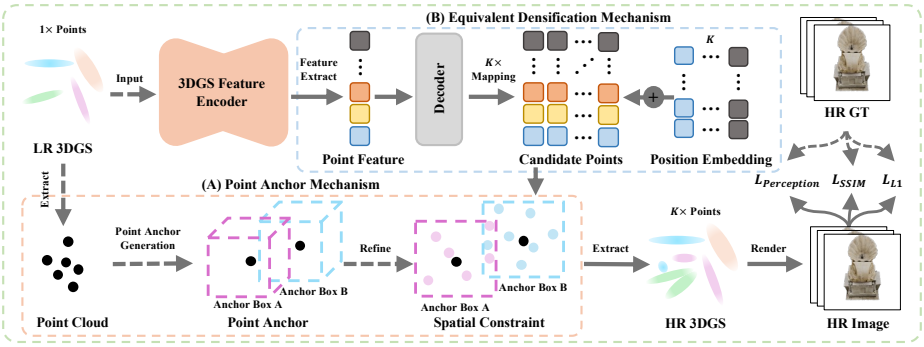

Fig. 2: Framework of AnchorSplat. Our model takes a low-quality 3DGS asset as input and first encodes its non-positional attributes into per-point features. (A) The Point Anchor Mechanism imposes a local geometric constraint by defining an anchor box around each input primitive. (B) The 3DGS Decoder then multiplicatively generates K detail-enhanced primitives strictly within this localized anchor space, utilizing a learnable position embedding. The final high-quality 3DGS asset is rendered and supervised end-to-end against high-resolution ground truth, ensuring strong geometric consistency and fidelity improvement.

utilize PTv3 to effectively extract local and global geometric context in a purely 3D manner, which provides the crucial semantic basis for the Point Anchor Mechanism.

3 Method

3.1 Overview

Existing 3D super-resolution methods primarily achieve high-resolution results indirectly through 2D image enhancement and 3D optimization. This approach struggles to guarantee robust multi-view 3D consistency and cannot directly exploit the explicit geometric structure of 3DGS assets. To overcome these limitations, we propose AnchorSplat.

AnchorSplat is designed to transform a set of low-quality 3DGS assets $\{\mathcal{G}_{\text{low}}\}_{i=1}^{N}$ into detail-rich, high-fidelity assets $\{\mathcal{G}_{\text{high}}\}_{i=1}^{N}$. Our task is formally defined as:

$$\mathcal{G}_{\text{high}} = \mathcal{F}_{\text{AnchorSplat}}(\mathcal{G}_{\text{low}}). \tag{1}$$

The core components of $\mathcal{F}_{\text{AnchorSplat}}$ include: (1) a 3DGS Feature Encoder, (2) the Point Anchor Mechanism, and (3) the Equivalent Densification Mechanism. The overall framework of AnchorSplat is illustrated in Fig. 2.

3.2 3DGS Feature Encoder

We treat the low-quality 3DGS asset $\mathcal{G}_{\text{low}}$ as an attribute-rich 3D point cloud. The input is a set of n Gaussian primitives $\mathcal{G}_{\text{low}} = \{(\boldsymbol{\mu}_j, \mathbf{c}_j, \mathbf{s}_j, \mathbf{R}_j, \alpha_j)\}_{j=1}^{n}$, where

$\boldsymbol{\mu}_j$ denotes the 3D position, and the remaining terms $(\mathbf{c}_j, \mathbf{s}_j, \mathbf{R}_j, \alpha_j)$ represent the rendering attributes. To extract features with local context awareness from this sparse, unstructured representation, we first concatenate all non-positional attributes into a high-dimensional initial attribute feature vector $\mathbf{v}_j \in \mathbb{R}^M$:

$$\mathbf{v}_j = \text{Concat}(\mathbf{c}_j, \text{Vec}(\mathbf{s}_j), \text{Vec}(\mathbf{R}_j), \alpha_j). \tag{2}$$

Subsequently, we input $\boldsymbol{\mu}_j$ and $\mathbf{v}_j$ into the Point Transformer v3 (PTv3) encoder $\mathcal{E}_{\text{PTv3}}$. $\mathcal{E}_{\text{PTv3}}$ efficiently aggregates geometric and attribute information within the neighborhood, transforming the sparse input primitives into a robust high-dimensional feature descriptor $\mathbf{f}_j \in \mathbb{R}^D$:

$$\mathbf{f}_j = \mathcal{E}_{\text{PTv3}}(\boldsymbol{\mu}_j, \mathbf{v}_j). \tag{3}$$

The resulting feature $\mathbf{f}_j$ provides the crucial semantic basis for the Point Anchor Mechanism.

3.3 Point Anchor Mechanism

The optimization objective of AnchorSplat is to minimize the total rendering loss across all views $\mathcal{V}$, defined as $\mathcal{L}_{\text{total}} = \sum_{v \in \mathcal{V}} \mathcal{L}_v$. The core challenge of this process lies in the mapping from the 3D feature $\mathbf{f}_j$, extracted by the encoder as a local descriptor in the vicinity of $\boldsymbol{\mu}_j$, to the global novel primitives $g'_{j,k}$ generated by the decoder $\mathcal{D}_{\text{Gen}}$. An unconstrained decoder $\mathcal{D}_{\text{Gen}}$ results in a fundamental mapping ambiguity.

In practice, such unconstrained training leads to catastrophic convergence failure: while the network learns the coarse envelope of the scene, it ultimately decodes highly blurry geometry with chaotic colors, indicating that the precise optimization signals required for high-frequency details are completely lost.

To investigate the fundamental reason of this convergence barrier, we analyze the global gradient received by $\mathbf{f}_j$. Early in training, the decoder $\mathcal{D}_{\text{Gen}}$ is randomly initialized, and there lacks any prior spatial correspondence between $\mathbf{f}_j$ and the primitives $g'_{j,k}$ it generates. Let $\mathcal{P}_v$ be the set of all pixels in view v, and $\mathbf{C}_p^v$ be the rendered color of pixel p. According to the chain rule, the total gradient with respect to $\mathbf{f}_j$ is the summation of the rendering loss contributions from its K generated primitives $g'_{j,k}$ over all views v and all pixels $p \in \mathcal{P}_v$:

$$\frac{\partial \mathcal{L}_{\text{total}}}{\partial \mathbf{f}_j} = \sum_{v \in \mathcal{V}} \sum_{p \in \mathcal{P}_v} \frac{\partial \mathcal{L}_v}{\partial \mathbf{C}_p^v} \cdot \left(\sum_{k=1}^{K} \frac{\partial \mathbf{C}_p^v}{\partial g'_{j,k}} \cdot \frac{\partial g'_{j,k}}{\partial \mathbf{f}_j} \right). \tag{4}$$

This mathematically reveals why this problem is ill-posed. First, for the mapping term $\frac{\partial g'_{j,k}}{\partial \mathbf{f}_j}$, an unconstrained decoder $\mathcal{D}_{\text{Gen}}$ may map $\mathbf{f}_j$ to a primitive $g'_{j,k}$ at an arbitrary global location. Second, the rendering term $\frac{\partial \mathbf{C}_p^v}{\partial g'_{j,k}}$ implies that this $g'_{j,k}$ can project onto an arbitrary pixel p in an arbitrary view v, thus generating a gradient. Consequently, $\mathbf{f}_j$, despite being a local feature, receives conflicting

and confounded gradient signals from all views and spatial locations. This is the fundamental cause for decoding blurry shapes and chaotic colors.

To resolve this mapping ambiguity and the issue of confounded gradients, we propose the Point Anchor Mechanism. This mechanism enforces the spatial correspondence by imposing an explicit 3D geometric constraint, thereby transforming the ill-posed problem into a well-posed one. We define the position of the new primitive $\boldsymbol{\mu}'_{j,k}$ as a relative offset $\Delta\boldsymbol{\mu}_{j,k}$ with respect to its anchor point $\boldsymbol{\mu}_j$:

$$\boldsymbol{\mu}'_{j,k} = \boldsymbol{\mu}_j + \epsilon \cdot \tanh(\Delta\boldsymbol{\mu}_{j,k}). \tag{5}$$

This constraint compels $\mathbf{f}_j$ to only generate new primitives $g'_{j,k}$ within a local geometric domain $\Omega_{\text{local}}(\boldsymbol{\mu}_j)$ in the vicinity of its 3D anchor $\boldsymbol{\mu}_j$. This constraint mathematically rewrites the gradient flow. Since $g'_{j,k}$ are geometrically constrained to $\Omega_{\text{local}}(\boldsymbol{\mu}_j)$, they can only project onto a local pixel set $\mathcal{P}_{\text{local}}(\boldsymbol{\mu}_j, v)$ in the 2D image of view v. Let $\mathcal{P}^j_{\text{local}} \equiv \mathcal{P}_{\text{local}}(\boldsymbol{\mu}_j, v)$ denote this view-dependent local pixel set for brevity. For pixels $p \notin \mathcal{P}^j_{\text{local}}$ outside this local region, the primitives $g'_{j,k}$ generated from $\mathbf{f}_j$ have no contribution to $\mathbf{C}^v_p$. Consequently, the gradient $\frac{\partial \mathbf{C}^v_p}{\partial g'_{j,k}}$ is mathematically and precisely zero:

$$\text{If } p \notin \mathcal{P}^j_{\text{local}}, \quad \frac{\partial \mathbf{C}^v_p}{\partial g'_{j,k}} = 0. \tag{6}$$

Substituting this zero-gradient condition into Eq. (4), the summation over all pixels $p \in \mathcal{P}_v$ in the global gradient automatically reduces to a summation over only the local pixels $p \in \mathcal{P}_{\text{local}}$:

$$\frac{\partial \mathcal{L}_{\text{total}}}{\partial \mathbf{f}_j} = \sum_{v \in \mathcal{V}} \sum_{p \in \mathcal{P}^j_{\text{local}}} \frac{\partial \mathcal{L}_v}{\partial \mathbf{C}^v_p} \cdot \left(\sum_{k=1}^{K} \frac{\partial \mathbf{C}^v_p}{\partial g'_{j,k}} \cdot \frac{\partial g'_{j,k}}{\partial \mathbf{f}_j} \right). \tag{7}$$

This is not an approximation, but the exact mathematical result of Eq. (4) under the constraint imposed by Eq. (6). This demonstrates that the gradients from all views are constrained to optimize the same 3D local region Ω_{local}. This effectively eliminates the confounding influence of distant primitives, rendering the optimization signal for $\mathbf{f}_j$ clean and 3D-consistent. This enables the network to learn fine-grained geometric and appearance details.

3.4 Equivalent Densification Mechanism

With the spatial localization and gradient confounding issues addressed, the second core challenge is the generation of a sufficient density of geometric primitives, which is essential for achieving high-fidelity detail. The original 3DGS framework achieves densification through an iteration-based, stateful optimization process, relying on runtime signals (e.g., $\nabla_{\mu_j}\mathcal{L}$ and s_j) to trigger cloning or splitting. This mechanism is fundamentally incompatible with our single-pass,

feed-forward architecture, which lacks access to these iteratively accumulated, state-dependent signals.

This necessitates a principled, equivalent substitute that can be executed in a single forward pass, driven only by the per-point feature f_j. We propose Equivalent Densification Mechanism that simultaneously provides an equivalent substitute for both cloning and pruning. For cloning, the decoder $\mathcal{D}_{Gen}$ is trained to predict a fixed set of K new primitives based on the single anchor feature f_j, effectuating a 1-to-K mapping:

$$\{\Delta\mu_{j,k}, c'_{j,k}, s'_{j,k}, R'_{j,k}, \alpha'_{j,k}\}_{k=1}^{K} = \mathcal{D}_{Gen}(f_j). \tag{8}$$

For pruning, the network learns to implicitly remove redundant or incorrectly placed primitives by predicting a vanishing opacity ($\alpha'_{j,k} \to 0$). This achieves a render-equivalent and fully differentiable substitute for the original hard pruning step.

Crucially, the final position $\mu'_{j,k}$ of these new primitives is still determined by the Point Anchor Mechanism, ensuring our new density remains spatially consistent. This design allows the $\mathcal{D}_{Gen}$ module to learn an end-to-end function that directly approximates the ideal density distribution $\mathcal{D}(\mathcal{G}(T))$ in a single pass, efficiently replacing the complex iterative process.

$$\mathcal{D}(\mathcal{G}_{high}) = \mathcal{D}(\mathcal{F}_{\text{AnchorSplat}}(\mathcal{G}_{low})) \approx \mathcal{D}(\mathcal{G}(T)). \tag{9}$$

The final high-quality asset $\mathcal{G}_{high}$ is composed of all newly generated primitives.

3.5 Loss Function

The training of the AnchorSplat model is supervised by high-resolution ground-truth images I_{GT} and their camera poses, utilizing a low-quality 3DGS asset $\{\mathcal{G}_{\text{low}}\}$ as input. During the training process, the model generates $\mathcal{G}_{\text{HR}}$, from which a predicted image $\hat{I}$ is rendered and subsequently compared against I_{GT}. Gradients are back-propagated through the differentiable rendering pipeline to optimize the parameters of the AnchorSplat model.

The total loss function $\mathcal{L}_{\text{total}}$ is defined as a weighted summation of three critical components:

$$\mathcal{L}_{\text{total}} = \lambda_1 \mathcal{L}_{L1} + \lambda_2 \mathcal{L}_{\text{SSIM}} + \lambda_3 \mathcal{L}_{\text{Perception}}, \tag{10}$$

where λ_1, λ_2, λ_3 are balancing coefficients. $\mathcal{L}_{L1}$ is the mean absolute error between $\hat{I}$ and I_{GT}, which is employed to ensure basic pixel-wise similarity. $\mathcal{L}_{\text{SSIM}}$ is defined as $1 - \text{SSIM}(\hat{I}, I_{GT})$, serving as a perceptual metric to guarantee structural fidelity. $\mathcal{L}_{\text{Perception}}$ is a VGG-based perceptual loss that measures distance in the deep feature space, which is essential for maintaining the visual realism of both global appearance and fine-grained details.

4 The 3DGS-SR Dataset

We position our work within the 3DSR domain, focusing on the asset enhancement paradigm. This paradigm is distinct from the traditional reconstruction from scratch approach, which optimizes a 3D representation from low-resolution images $\{I_{\text{LR}}\}$. Instead, asset enhancement aims to take an existing, low-quality 3DGS asset $\mathcal{G}_{\text{low}}$ as its direct input and enhance it into a high-fidelity counterpart. This common real-world task has historically faced a critical research bottleneck: the lack of a standardized, large-scale benchmark. To address this gap, we construct 3DGS-SR, a new benchmark comprising approximately 15k single-object assets sourced from Objaverse. It covers a wide diversity of categories (e.g., vehicles, furniture, household items) and designates 30 representative objects as a held-out test set, with the remainder used for training.

The core contribution of 3DGS-SR lies in its standardized $(\mathcal{G}_{\text{low}}, \{I_{\text{HR}}\})$ data pairs. To generate these, we first render 146 views for each asset from its pristine mesh at two resolutions: HR ground-truth images $\{I_{\text{HR}}\}$ at 1024×1024 and a corresponding LR images $\{I_{\text{LR}}\}$ at 256×256. This direct rendering process avoids the domain gap of 2D downsampling. Next, the task input $\mathcal{G}_{\text{low}}$ is generated by training the standard 3DGS algorithm on the LR images $\{I_{\text{LR}}\}$ for 15,000 iterations. Finally, we implement a PSNR-based screening mechanism, filtering out any asset where $\mathcal{G}_{\text{low}}$ achieves a PSNR below 34 against its own LR training images, ensuring a stable geometric baseline for all inputs. The evaluation protocol follows the original 3DGS methodology, where models enhance $\mathcal{G}_{\text{low}}$ and are then rendered from held-out test views for comparison against the $\{I_{\text{HR}}\}$ ground-truth.

5 Experiments

5.1 Experimental Setting

Experiments are primarily conducted on the newly constructed 3DGS-SR benchmark, with zero-shot generalization evaluated on the NeRF-synthetic dataset. AnchorSplat is compared against representative 3DSR methods including Bicubic Optimization, SRGS [6], SuperGaussian [25], and SequenceMatters [13]. To ensure a fair comparison, all methods utilize identical low-resolution source information from the 3DGS-SR dataset. Specifically, reconstruction-based methods utilize LR images $\{\mathcal{I}_{\text{LR}}\}$, while AnchorSplat takes the corresponding low-quality 3DGS assets $\mathcal{G}_{\text{low}}$ as input.

Standard NVS metrics including PSNR, SSIM, and LPIPS are employed to evaluate rendering quality, with processing time (Time) reported to contrast the efficiency of the feed-forward paradigm against optimization-based methods. The model is implemented in PyTorch and trained via the Adam optimizer. The objective function $\mathcal{L}_{\text{total}}$ is defined as a weighted sum of $\mathcal{L}_{\text{L1}}$, $\mathcal{L}_{\text{SSIM}}$ (formulated as $1 - \text{SSIM}$), and a VGG-based $\mathcal{L}_{\text{Perception}}$. Training was conducted on 8 A800 GPUs.

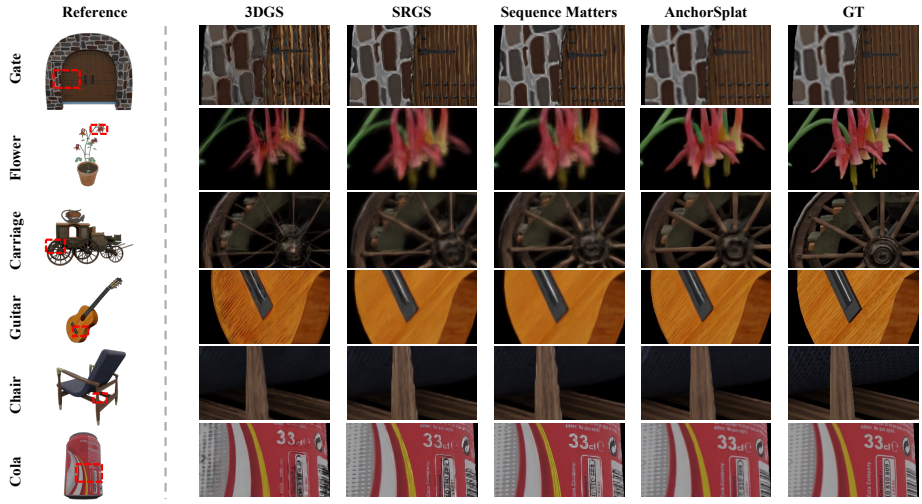

Fig. 3: Qualitative comparison on 3DGS-SR. We select several representative scenes from the 3DGS-SR test set for visual analysis. For image-based methods, the input is the LR image. 2D super-resolution methods yield artifacts and oversmoothing at geometric edges and high-frequency details due to inconsistency; AnchorSplat avoids these phenomena.

5.2 Results and Analysis

Main Results on 3DGS-SR Our primary objective is to enhance an existing low-quality 3DGS asset. Therefore, our most direct baseline is the input to AnchorSplat, represented by the 3DGS row in Tab. 1, which is the low-quality asset reconstructed from LR images.

The quantitative results in Tab. 1 demonstrate that AnchorSplat achieves state-of-the-art performance, significantly surpassing the input baseline and all compared 3DSR paradigms, such as SRGS and SequenceMatters. This empirical evidence validates the two primary limitations of 2D-centric approaches: extremely low throughput and multi-view inconsistency. Tab. 1 clearly quantifies the critical efficiency bottleneck: while traditional optimization-based methods require 0.25 to 1 hour of per-scene optimization, our single-pass, feed-forward network achieves this superior state-of-the-art quality in approximately 0.01 seconds. This realizes a 10^5-fold increase in throughput.

Furthermore, these quantitative results are corroborated by the qualitative comparisons in Fig. 3. This visual evidence confirms that 2D-based methods visibly introduce the artifacts and geometric inconsistencies predicted in our introduction. In contrast, AnchorSplat successfully synthesizes fine, structurally-consistent details. This demonstrates that our 3D native approach is superior in both efficiency and fidelity, as the qualitative advantages are directly reflected in the state-of-the-art quantitative metrics.

Table 1: Quantitative comparison on the 3DGS-SR dataset (1024 × 1024). **Source-Free** indicates methods operating strictly on the 3D asset without accessing original multi-view images. AnchorSplat achieves state-of-the-art fidelity with real-time processing speed.

Method	Source-Free	PSNR↑	SSIM↑	LPIPS↓	Time↓
3DGS [11]	–	31.03	0.917	0.076	–
Bicubic	×	34.36	0.923	0.064	∼ 4m
SRGS [6]	×	35.24	0.941	0.104	∼ 16m
Sequence Matters [13]	×	35.69	0.937	0.074	∼ 34m
SuperGaussian [25]	✓	34.94	0.924	0.097	∼ 41m
AnchorSplat (Ours)	✓	**36.57**	**0.943**	**0.058**	∼ **0.01s**

Table 2: Quantitative comparison on the NeRF-synthetic [21] dataset. **Source-Free** indicates methods operating strictly on the 3D asset without accessing original multi-view images. AnchorSplat achieves the highest PSNR among all source-free methods while being orders of magnitude faster.

Method	Source-Free	PSNR↑	SSIM↑	LPIPS↓	Time↓
3DGS [11]	–	23.30	0.872	0.114	–
Bicubic	×	27.56	0.915	0.104	∼ 4m
DiSR-NeRF [14]	×	26.00	0.889	0.122	∼ 30m
NeRF-SR [28]	×	28.46	0.921	0.076	∼ 24h
Gaussian-SR [39]	×	28.37	0.924	0.087	∼ 15m
SRGS [6]	×	30.83	0.948	0.056	∼ 18m
Sequence Matters [13]	×	**31.41**	**0.952**	**0.054**	∼ 40m
SuperGaussian [25]	✓	28.44	0.945	0.067	∼ 45m
AnchorSplat (Ours)	✓	28.97	0.935	0.077	∼ **0.01s**

Generalization Ability and Efficiency Analysis The generalization capability of AnchorSplat is further evaluated by applying the model directly to the NeRF-synthetic dataset without fine-tuning. As shown in Tab. 2, AnchorSplat delivers substantial quality gains over baseline 3DGS assets, demonstrating robust cross-dataset performance.

This cross-dataset test also highlights the critical trade-off between fidelity and practicality under the Source-Free constraint. In comparison, methods such as Sequence Matters achieve a higher final PSNR; however, they require access to the original multi-view images to execute a complex pipeline involving 2D super-resolution and subsequent 3D re-optimization, costing approximately 40 minutes per scene.

In stark contrast, AnchorSplat operates in a strictly Source-Free manner, achieving the highest PSNR among all methods that do not require original source images. Our method provides this substantial quality improvement in

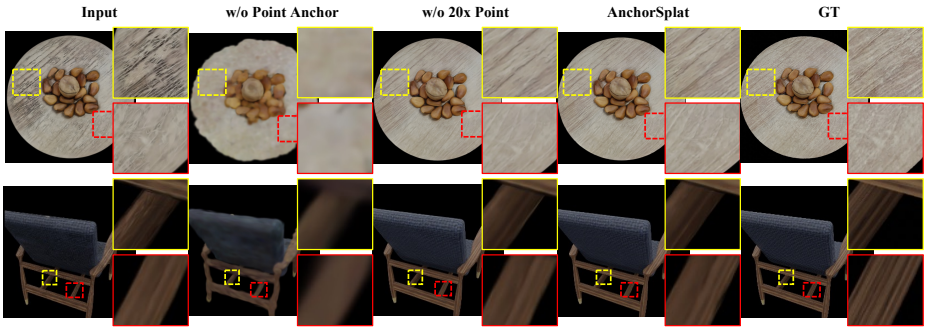

Fig. 4: Ablation study visualization. We perform separate ablations on the Point Anchor Mechanism and the Multiplicative Primitive Factor K. The visual results clearly demonstrate that the Point Anchor Mechanism is crucial for this task, while the generation of a high Multiplicative Primitive Factor ensures smoother and richer texture details.

Table 3: Ablation study on the 3DGS-SR dataset. Bold indicates the best result. The results demonstrate that our model achieves the best performance.

Method	PSNR↑	SSIM↑	LPIPS↓
w/o Point Anchor	26.79	0.886	0.186
w/ 1× Points	36.42	0.943	0.063
w/ 10× Points	36.51	0.944	0.060
Ours (20× Points)	**36.57**	**0.944**	**0.058**

approximately 0.01 seconds, completely eliminating the need for any per-scene optimization or external image dependencies.

These results fully substantiate the powerful generalization of our model and its superior efficiency-accuracy trade-off. By outperforming other Source-Free alternatives like SuperGaussian in both quality and speed, AnchorSplat positions itself as a far more practical solution for high-fidelity deployment than conventional, time-consuming optimization methods.

5.3 Ablation Study

Effectiveness of the Point Anchor Mechanism To validate the necessity of the Point Anchor Mechanism, we ablate this component and retrain the model. As shown in Tab. 3, removing this module leads to a severe collapse in model performance. Quantitatively, all metrics sharply deteriorate: PSNR drops by nearly 10 dB, while SSIM falls significantly and the perceptual metric LPIPS indicates a severe degradation.

Qualitatively, Fig. 4 provides more direct visual evidence. Without the Point Anchor constraint, the rendered geometry and texture become chaotic and dis-

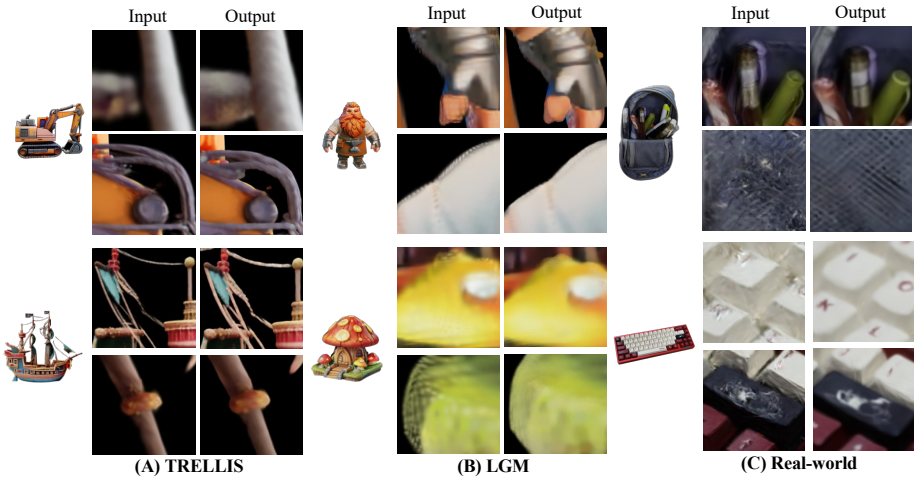

Fig. 5: Zero-shot generalization on diverse 3D sources. Without any fine-tuning, AnchorSplat acts as a plug-and-play enhancer for 3D generative models. It successfully sharpens mechanical boundaries for Trellis outputs (A), enriches complex geometric textures for LGM outputs (B), and robustly enhances details in unconstrained, noisy real-world captures (C).

orderly, with a near-total loss of high-frequency details. This result confirms that the Point Anchor Mechanism is fundamental. Without this local geometric constraint, the model suffers from the gradient confounding and mapping ambiguity issues inherent in direct 3D prediction, leading to ineffective optimization signals that prevent the learning of high-frequency information and result in the observed failure in detail synthesis.

Effectiveness of Equivalent Densification Mechanism Next, we validate the contribution of primitive density, regulated by the Equivalent Densification Mechanism, to model accuracy.

As shown in Tab. 3, we compared the full model using a multiplicative factor of $K = 20$ against variants with lower factors, such as $K = 1$ and $K = 10$. The results show that increasing the multiplicative factor from $K = 1$ to $K = 20$ consistently enhances all metrics, with PSNR improving and LPIPS dropping at each step.

Fig. 4 visually confirms this finding. The $K = 1$ low-density variant renders noticeably blurrier texture details and struggles to produce sharp high-frequency textures compared to the full model with $K = 20$. This experimental observation confirms the direct positive impact of primitive densification on model accuracy, reflecting the necessity of a higher density of geometric primitives for synthesizing high-frequency details.

This validates Equivalent Densification Mechanism as an effective and necessary component for achieving high-fidelity detail synthesis, confirming it serves as a successful substitute for traditional iterative densification.

5.4 Generalization to 3D Generative Models and Real-World Scans

AnchorSplat demonstrates robust zero-shot generalization across diverse data distributions, ranging from feed-forward generative outputs of Trellis and LGM to noisy real-world captures. Although trained exclusively on synthetic datasets, our model consistently refines coarse geometry and restores high-frequency appearance details within a single 0.01s forward pass. This capability directly addresses the textural blurriness and structural ambiguity commonly observed in current large-scale 3D generative models, making AnchorSplat suitable as a lightweight post-processing module for generated 3D assets.

As illustrated in Fig. 5 (A), AnchorSplat significantly enhances the structural definition of Trellis outputs, particularly for mechanical geometries. For the excavator and ship models, our method recovers sharp structural boundaries and clarifies thin, intricate components that were initially rendered with substantial ambiguity. In LGM-generated assets shown in Fig. 5 (B), the Point Anchor Mechanism effectively suppresses generative artifacts and sharpens complex surface textures, such as the detailed armor of the dwarf and the organic patterns of the mushroom house. These results suggest that AnchorSplat does not merely amplify local colors, but learns to reorganize Gaussian attributes and positions in a structure-aware manner.

Crucially, this performance is achieved across a substantial domain gap in appearance modeling. While our model is trained on assets with a Spherical Harmonics (SH) degree of 3, it seamlessly processes the diffuse-only (SH=0) outputs of current generative pipelines without fine-tuning. This capability indicates that the model learns intrinsic geometric and structural priors rather than overfitting to a specific lighting representation. Furthermore, evaluations on unconstrained real-world scans presented in Fig. 5 (C), including objects such as the pencil case and keyboard, show that AnchorSplat remains resilient to capture noise, imperfect reconstruction, and sensor artifacts. By bridging coarse generative outputs and high-fidelity 3D assets, AnchorSplat provides a practical, plug-and-play solution for broader 3D content creation workflows.

6 Conclusion

This paper presents AnchorSplat, a 3D-native feed-forward network addressing low throughput and multi-view inconsistency in 3DGS enhancement. The proposed Point Anchor Mechanism mitigates ill-posed mapping, while the single-pass Equivalent Densification Mechanism replaces iterative optimization. Notably, AnchorSplat is a strictly source-free solution requiring no original multi-view images. We also introduce 3DGS-SR, the first large-scale benchmark for this

task. Experiments demonstrate state-of-the-art fidelity with 10^5 times throughput gains over optimization methods. Furthermore, AnchorSplat exhibits robust zero-shot generalization across generative outputs and real-world scans, establishing the framework as a practical enhancer for high-fidelity 3D synthesis.

Acknowledgements

This work was supported by the New Generation Artificial Intelligence–National Science and Technology Major Project under Grant 2025ZD0123505; the National Natural Science Foundation of China under Grants 62576338, 62576342, 62506362, 62550062, 62425606, and 32341009; and the Beijing Natural Science Foundation under Grants L252145, L257008, and 4252054.

References

1. Chen, Y., Mihajlovic, M., Chen, X., Wang, Y., Prokudin, S., Tang, S.: Splatformer: Point transformer for robust 3d gaussian splatting. arXiv preprint arXiv:2411.06390 (2024)
2. Dong, C., Loy, C.C., He, K., Tang, X.: Image super-resolution using deep convolutional networks. IEEE transactions on pattern analysis and machine intelligence (2015)
3. Duan, J., Liu, S., Hao, Y., Huang, H., He, R.: Dual frequency-guided spatiotemporal feature learning for face forgery detection. IEEE Transactions on Biometrics, Behavior, and Identity Science (2025)
4. Dumic, E., da Silva Cruz, L.A.: Three-dimensional point cloud applications, datasets, and compression methodologies for remote sensing: A meta-survey. Sensors (2025)
5. Fan, Z., Zhang, J., Cong, W., Wang, P., Li, R., Wen, K., Zhou, S., Kadambi, A., Wang, Z., Xu, D., et al.: Large spatial model: End-to-end unposed images to semantic 3d. Advances in neural information processing systems (2024)
6. Feng, X., He, Y., Wang, Y., Yang, Y., Li, W., Chen, Y., Kuang, Z., Fan, J., Jun, Y., et al.: Srgs: Super-resolution 3d gaussian splatting. arXiv preprint arXiv:2404.10318 (2024)
7. Han, X., Tang, Y., Wang, Z., Li, X.: Mamba3d: Enhancing local features for 3d point cloud analysis via state space model. In: Proceedings of the 32nd ACM International Conference on Multimedia (2024)
8. Han, Y., Yu, T., Yu, X., Xu, D., Zheng, B., Dai, Z., Yang, C., Wang, Y., Dai, Q.: Super-nerf: View-consistent detail generation for nerf super-resolution. IEEE Transactions on Visualization and Computer Graphics (2024)
9. Huang, B., Yu, Z., Chen, A., Geiger, A., Gao, S.: 2d gaussian splatting for geometrically accurate radiance fields. In: ACM SIGGRAPH 2024 conference papers (2024)
10. Huang, Y., Miyazaki, T., Liu, X., Omachi, S.: Infrared image super-resolution: A systematic review and future trends. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing (2025)
11. Kerbl, B., Kopanas, G., Leimkühler, T., Drettakis, G.: 3d gaussian splatting for real-time radiance field rendering. ACM Trans. Graph. (2023)

12. Kim, J., Lee, J.K., Lee, K.M.: Deeply-recursive convolutional network for image super-resolution. In: Proceedings of the IEEE conference on computer vision and pattern recognition (2016)
13. Ko, H.k., Park, D., Park, Y., Lee, B., Han, J., Park, E.: Sequence matters: Harnessing video models in 3d super-resolution. In: Proceedings of the AAAI Conference on Artificial Intelligence (2025)
14. Lee, J.L., Li, C., Lee, G.H.: Disr-nerf: Diffusion-guided view-consistent super-resolution nerf. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2024)
15. Li, G., Xing, W., Zhao, L., Lan, Z., Sun, J., Zhang, Z., Zhang, Q., Lin, H., Lin, Z.: Self-reference image super-resolution via pre-trained diffusion large model and window adjustable transformer. In: Proceedings of the 31st ACM International Conference on Multimedia (2023)
16. Liang, D., Zhou, X., Xu, W., Zhu, X., Zou, Z., Ye, X., Tan, X., Bai, X.: Pointmamba: A simple state space model for point cloud analysis. Advances in neural information processing systems (2024)
17. Liang, J., Cao, J., Sun, G., Zhang, K., Van Gool, L., Timofte, R.: Swinir: Image restoration using swin transformer. In: Proceedings of the IEEE/CVF international conference on computer vision (2021)
18. Lin, J.: Dynamic nerf: A review. arXiv preprint arXiv:2405.08609 (2024)
19. Liu, Y., Pan, J., Li, Y., Dong, Q., Zhu, C., Guo, Y., Wang, F.: Ultravsr: Achieving ultra-realistic video super-resolution with efficient one-step diffusion space. In: Proceedings of the 33rd ACM International Conference on Multimedia (2025)
20. Lu, J., Zhu, D., Shi, H., Cai, L., Tang, G., Chen, Y., Cao, J., Tang, D., Zhang, Y., Dai, Y., et al.: Current world models lack a persistent state core. arXiv preprint arXiv:2606.20545 (2026)
21. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: Nerf: Representing scenes as neural radiance fields for view synthesis. Communications of the ACM (2021)
22. Qi, C.R., Su, H., Mo, K., Guibas, L.J.: Pointnet: Deep learning on point sets for 3d classification and segmentation. In: Proceedings of the IEEE conference on computer vision and pattern recognition (2017)
23. Qi, C.R., Yi, L., Su, H., Guibas, L.J.: Pointnet++: Deep hierarchical feature learning on point sets in a metric space. Advances in neural information processing systems (2017)
24. Saharia, C., Ho, J., Chan, W., Salimans, T., Fleet, D.J., Norouzi, M.: Image super-resolution via iterative refinement. IEEE transactions on pattern analysis and machine intelligence (2022)
25. Shen, Y., Ceylan, D., Guerrero, P., Xu, Z., Mitra, N.J., Wang, S., Frühstück, A.: Supergaussian: Repurposing video models for 3d super resolution. In: European Conference on Computer Vision. Springer (2024)
26. Sohail, S.S., Himeur, Y., Kheddar, H., Amira, A., Fadli, F., Atalla, S., Copiaco, A., Mansoor, W.: Advancing 3d point cloud understanding through deep transfer learning: A comprehensive survey. Information Fusion (2025)
27. Wan, Y., Shao, M., Cheng, Y., Zuo, W.: S2gaussian: Sparse-view super-resolution 3d gaussian splatting. In: Proceedings of the Computer Vision and Pattern Recognition Conference (2025)
28. Wang, C., Wu, X., Guo, Y.C., Zhang, S.H., Tai, Y.W., Hu, S.M.: Nerf-sr: High quality neural radiance fields using supersampling. In: Proceedings of the 30th ACM International Conference on Multimedia (2022)

29. Wang, H., Cao, J., Liu, J., Zhou, X., Huang, H., He, R.: Hallo3d: Multi-modal hallucination detection and mitigation for consistent 3d content generation. Advances in Neural Information Processing Systems **37**, 118883–118906 (2024)
30. Wang, X., Li, M., Liu, W., Zhang, H., Hu, S., Zhang, Y., Zhou, Z., Jin, H.: Unlearnable 3d point clouds: Class-wise transformation is all you need. Advances in Neural Information Processing Systems (2024)
31. 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 (2024)
32. Wu, X., Jiang, L., Wang, P.S., Liu, Z., Liu, X., Qiao, Y., Ouyang, W., He, T., Zhao, H.: Point transformer v3: Simpler faster stronger. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (2024)
33. Wu, X., Lao, Y., Jiang, L., Liu, X., Zhao, H.: Point transformer v2: Grouped vector attention and partition-based pooling. Advances in Neural Information Processing Systems (2022)
34. Wu, Z., Wan, Z., Zhang, J., Liao, J., Xu, D.: Rafe: Generative radiance fields restoration. In: European Conference on Computer Vision. Springer (2024)
35. Xiao, Z., Wang, X.: Event-based video super-resolution via state space models. In: Proceedings of the Computer Vision and Pattern Recognition Conference (2025)
36. Xu, H., Barath, D., Geiger, A., Pollefeys, M.: Resplat: Learning recurrent gaussian splats. arXiv preprint arXiv:2510.08575 (2025)
37. Xu, Y., Park, T., Zhang, R., Zhou, Y., Shechtman, E., Liu, F., Huang, J.B., Liu, D.: Videogigagan: Towards detail-rich video super-resolution. In: Proceedings of the Computer Vision and Pattern Recognition Conference (2025)
38. Yan, Z., Li, L., Shao, Y., Chen, S., Wu, Z., Hwang, J.N., Zhao, H., Remondino, F.: 3dsceneeditor: Controllable 3d scene editing with gaussian splatting. arXiv preprint arXiv:2412.01583 (2024)
39. Yu, X., Zhu, H., He, T., Chen, Z.: Gaussiansr: 3d gaussian super-resolution with 2d diffusion priors. arXiv preprint arXiv:2406.10111 (2024)
40. Zeng, H., Bai, Y., Fu, Y.: Arbitrary-scale 3d gaussian super-resolution. arXiv preprint arXiv:2508.16467 (2025)
41. Zhang, T., Liang, Z., Wang, B.: A survey of medical point cloud shape learning: Registration, reconstruction and variation. arXiv preprint arXiv:2508.03057 (2025)
42. Zhang, W., Zhou, J., Geng, H., Zhang, W., Liu, Y.S.: Gap: Gaussianize any point clouds with text guidance. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (2025)
43. Zhao, H., Jiang, L., Jia, J., Torr, P.H., Koltun, V.: Point transformer. In: Proceedings of the IEEE/CVF international conference on computer vision (2021)
44. Zheng, M., Sun, L., Dong, J., Pan, J.: Efficient video super-resolution for real-time rendering with decoupled g-buffer guidance. In: Proceedings of the Computer Vision and Pattern Recognition Conference (2025)
45. Zhou, J., Zhang, W., Liu, Y.S.: Diffgs: Functional gaussian splatting diffusion. Advances in Neural Information Processing Systems (2024)
46. Zhu, D., Cao, J., Shao, J., Zhang, Z., Duan, J., He, R.: Mtsd: Simple yet effective self-distillation for generalizable deepfake detection. In: 2025 IEEE International Conference on Multimedia and Expo (ICME). pp. 1–6. IEEE (2025)
47. Zhu, Q., Fan, L., Weng, N.: Advancements in point cloud data augmentation for deep learning: A survey. Pattern recognition (2024)