Gaussian Volumetric Representation for Efficient Shear-Warp Visualization

Mayuri Mathur and Ojaswa Sharma

Indraprastha Institute of Information Technology Delhi, New Delhi, India
mayurim@iiitd.ac.in, ojaswa@iiitd.ac.in
https://graphics-research-group.github.io/Gaussian-Volumetric-Representation/

Abstract. Medical image visualization requires volumetric rendering algorithms that preserve anatomical fidelity while maintaining high rendering speeds. To address the high computational cost of large volumetric datasets, we propose a Gaussian-based volumetric representation for efficient visualization of dense medical volumes without compromising structural and radiometric details. We optimize the proposed representation using Monte Carlo volumetric estimation, which enables training on a highly sparse subset of voxels while maintaining consistency with the dense volumetric objective. In addition, we introduce a curriculum learning strategy that progressively incorporates structured slice-based sampling during training. Sparse voxel samples provide an early global coverage of the volume, while slice samples capture spatially correlated regions that aid geometric structure and texture continuity. This combination enables the Gaussian representation to learn anatomical details of various structures and corresponding textures from sparse supervision while significantly reducing the computational cost associated with dense voxel processing. The learned representation supports slice-based rendering methods such as shear–warp volume rendering, enabling efficient visualization of multimodal medical datasets including MRI and Cryosection volumes while preserving anatomical structures. Using sparse supervision, our method achieves up to 43.86 FPS rendering with a compression ratio of 11.31:1.

Keywords: Gaussian representation · Volume representation · Monte Carlo estimation · Curriculum Learning

1 Introduction

Medical volumetric data like MRI, CT, and Cryosection data are high-dimensional datasets and resource-hungry during real-time visualization. The ray marching [8, 12] volumetric visualization technique renders each pixel by sampling and interpolating along a camera ray through the volume, making it computationally expensive for high-resolution views and large volumetric datasets.

Current methods encompassing the domain of volumetric representation and compression comprise conventional [1, 14, 22] and learning based techniques [24,

29, 35]. Conventional methods such as tensor decomposition [14], discrete cosine transform [1], and octree-based [22] compression aim to reduce storage requirements of volumetric data by approximating the voxel grid using compact representations. While these methods provide efficient compression, they primarily operate on voxel-level signal approximations and do not explicitly model the volumetric structure required for rendering operations. As a result, these approaches are not well suited for applications that require flexible reconstruction of volumetric signals under varying rendering configurations, particularly for real-time visualization of large medical volumes.

Learning-based techniques such as feedforward networks [24], convolutional neural networks (CNNs) [4], and Transformers [36] have been used to represent volumetric data through learned embeddings. Although these approaches can capture fine details, modeling large volumes often requires a large number of parameters, making them computationally expensive. Optimization-based rendering techniques [9, 26] have also been proposed for real-time reconstruction; however, most of these methods are surface-based and focus on reconstructing object boundaries rather than modeling the full volumetric interior. As a result, they fail to capture the fine cross-sectional structures present in dense volumetric datasets such as MRI and Cryosection.

Efficient learning of volumetric representations therefore, requires training strategies that approximate the dense voxel objective while minimizing computational cost. Gaussian splatting [10, 33, 39] is primarily optimized for surface-like scene representations, and therefore struggles to directly model voxel-based medical volumes where meaningful information is distributed across internal cross-sectional structures rather than only on visible surfaces. To address these limitations, we propose a Gaussian-based volumetric representation that learns a compact approximation of dense medical volumes. Inspired by the recent success of Gaussian splatting in real-time rendering, we extend this representation to volumetric medical data for efficient reconstruction and visualization. We optimize the proposed representation using a curriculum learning strategy that combines sparse voxel sampling and structured slice-based sampling during training. Sparse voxels provide global coverage of the volume, while slice samples capture spatially correlated regions that encode geometric structure and texture continuity. This combination enables the Gaussian representation to efficiently learn both global volumetric consistency and local anatomical details. Since shear-warp rendering operates on stacks of 2D slices, incorporating slice-based supervision during training naturally aligns the learned representation with the rendering process.

The proposed framework is applicable to multimodal volumetric datasets including MRI and Cryosection volumes. By learning a compact Gaussian representation of organ structures, the method enables efficient reconstruction and supports real-time selective visualization of individual organs. Our contributions can be summarized as follows:

– A Gaussian-based volumetric representation that approximates a dense medical volume using a sparse set of learnable Gaussian kernels.

- A training strategy that utilizes Monte Carlo voxel sampling for efficient optimization of the volumetric representation from sparse supervision.
- A curriculum learning-based approach to training Gaussians using a mix of volumetric and planar slice-based sampling strategies. Incorporating slice samples provides spatially correlated supervision that improves reconstruction of anatomical structures compared to isolated voxel samples.
- A fast GPU-based Gaussian volume renderer based on the Shear-warp algorithm for real-time visualization.

Furthermore, the learned representation naturally supports slice-based rendering techniques such as shear–warp volume rendering, enabling high frame-rate visualization of large volumetric datasets.

2 Related Work

Learning-based methods in volumetric representation: Learning-based approaches have been explored for representing volumetric data using continuous implicit functions. Neural Radiance Fields (NeRF) [7, 11, 25, 30, 34, 40] approximate a volumetric function that maps 3D spatial coordinates and viewing directions to color and density values using a multilayer perceptron (MLP). Rendering a single image requires evaluating the network multiple times along each camera ray, making the rendering process computationally expensive due to repeated MLP inference and dense ray sampling.

Although NeRF can capture detailed volumetric signals, the use of large neural networks and dense sampling along rays leads to high computational overhead during both training and rendering. For high-resolution volumetric datasets such as medical scans, the number of network evaluations required to reconstruct fine spatial structures can become prohibitively expensive.

CNNs [8,12,21,38] have also been used to learn volumetric representations by operating directly on voxel grids. While CNN-based models can capture spatial context through convolutional filters, they often require dense volumetric inputs and large memory footprints when applied to high-resolution volumes.

More recently, Transformer [2, 3, 6, 17, 19, 32, 35, 41] models have been explored for volumetric representation learning by modeling long-range spatial dependencies across volumetric data. Although transformers can capture global context effectively, their computational complexity grows rapidly with the size of the input volume, making them challenging to scale for large medical datasets. GNT-MOVE [6] extends NeRF-based novel-view synthesis using Transformer-based view aggregation. CVT-xRF [41] is a NeRF-based technique that uses Transformers to build a more consistent 3D radiance field from multiple input views. It applies cross-view reasoning, where the Transformer exchanges and aggregates features across images of the same scene, assigning higher importance to the views that provide more useful information for reconstruction. These limitations motivate the need for compressed volumetric representations that can efficiently capture volumetric structure while reducing the computational cost associated with dense neural representations.

Surface rendering with Gaussians: Surface-oriented Gaussian Splatting methods have been widely explored for novel-view synthesis and visualization [5,10,13,16,18,33,39]. Kerbl et al. [10] introduced 3D Gaussian Splatting, where a scene is represented as a set of anisotropic Gaussian primitives parameterized by position, covariance, color, and opacity. These Gaussians are projected onto the image plane using a visibility-aware rasterization procedure, enabling real-time novel-view synthesis with high visual fidelity. Liang et al. [18] proposed Analytic Splatting to reduce aliasing artifacts observed in Gaussian Splatting. Their method analytically integrates the Gaussian contribution across the pixel footprint, producing smoother renderings and preserving fine structural details. FreeSplatter [39] is a Gaussian Splatting framework that predicts 3D Gaussian primitives and camera parameters from uncalibrated sparse view images using a Transformer-based multi-view feature exchange technique. The Transformer exchanges and aggregates information across multiple input views, helping infer consistent 3D structures from sparse observations. Tang et al. [33] introduced iVR-GS for interactive visualization of volumetric medical data. The method groups Gaussian models associated with different transfer functions to enable interactive exploration and editing of volumetric structures. Kleinbeck et al. [13] proposed a layered Gaussian representation for immersive visualization of anatomical structures. Each Gaussian layer corresponds to a specific anatomical component and is trained sequentially to reduce overlap between structures. ClipGS [16] extends Gaussian splatting for interactive visualization of medical scans by introducing clipping operations that allow users to explore internal structures through dynamically adjustable slicing planes. Recent work [5] reformulates Gaussian primitives as volumetric density functions and performs ray-based integration through the Gaussian field to solve the radiative transfer equation. This enables physically consistent volumetric rendering but introduces additional computational cost compared to rasterized splatting methods.In contrast to surface-oriented Gaussian splatting techniques, our work focuses on learning a Gaussian representation that models the interior volumetric structure of medical datasets while enabling efficient reconstruction and visualization.

3 Method

As shown in Fig. 1, our method represents volumetric data using a set of Gaussian kernels that approximate the underlying continuous volumetric signal. The Gaussian representation enables reconstruction of the entire voxel grid while significantly reducing the number of primitives required for its representation. The reconstructed color at any spatial location is obtained by evaluating the Gaussian mixture. To efficiently optimize this representation, we initialize Gaussians from a sparse set of voxels and update their parameters using sampled supervision. Instead of computing the reconstruction loss over the full voxel grid, we use a Monte Carlo volumetric estimator [20] to approximate the dense objective using a subset of voxels while preserving consistency with the full-volume optimization. We further improve the learning of anatomical structures by com-

bining sparse voxel sampling with slice-based sampling by progressively training these sampling strategies using Curriculum learning [37]. In each training iteration, a plane with arbitrary orientation is sampled, and sparse pixels are selected on the plane. Each pixel corresponds to a three-dimensional voxel coordinate, allowing the Gaussian representation to learn spatial continuity across neighboring regions. This sampling strategy aligns with shear–warp volume rendering, which generates images from stacks of two-dimensional slices extracted from a three-dimensional volume. In the following sections, we first present our Gaussian representation designed for the shear–warp renderer, followed by an analysis of the performance improvements achieved by the proposed method.

3.1 Gaussian representation

Consider representing a given voxel grid with a set of Gaussian kernels [10] estimated from m voxel samples. We initialize one Gaussian kernel centered at each sampled voxel position and define the set $\mathcal{G} = \{G_1, G_2, \ldots, G_m\}$ of m Gaussians. Each Gaussian has associated learnable parameters: mean $\boldsymbol{\mu} \in \mathbb{R}^3$, covariance matrix $\boldsymbol{\Sigma} \in \mathbb{R}^{3\times3}$, RGB color $\mathbf{C} \in \mathbb{R}^3$, and opacity $\alpha \in \mathbb{R}$ that are estimated during optimization. The Gaussian mean $\boldsymbol{\mu}$ determines the spatial location of the colored blob, while the covariance matrix $\boldsymbol{\Sigma}$ controls the spatial spread and orientation of the color contribution. Although each Gaussian has a constant color parameter $\mathbf{C}$, the Gaussian weighting function formulates its spatial influence, producing a continuous volumetric color field when multiple Gaussians are combined. This Gaussian representation defines a continuous volumetric field $\mathbf{f}$ of colors computed as the weighted contribution of all Gaussian kernels in the mixture

$$\mathbf{f}(\mathbf{x}) = \sum_{i=1}^{m} \alpha_i \mathbf{C}_i \exp\left(-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu}_i)^\top \boldsymbol{\Sigma}_i^{-1}(\mathbf{x}-\boldsymbol{\mu}_i)\right). \tag{1}$$

3.2 Monte Carlo volumetric estimation

Let Ω denote the set of all n voxels in the volume and $\mathbf{x}$ be the 3D coordinate of a voxel. Our reconstruction objective is to minimize the voxel-wise mean squared error over the entire volume V

$$L_{\text{dense}} = \frac{1}{n}\sum_{\mathbf{x}\in\Omega} \ell(\mathbf{x};\Theta), \qquad \ell(\mathbf{x};\Theta) = \|V_\Theta(\mathbf{x}) - V(\mathbf{x})\|_2^2, \tag{2}$$

where $V(\mathbf{x})$ is the ground-truth voxel color at coordinate $\mathbf{x}$ and V_Θ is the Gaussian-based continuous volume representation parameterized by Θ. Optimizing the Gaussian volumetric representation using all voxels of a volume is computationally expensive for high-resolution grids, since every iteration requires computing the predicted color and gradient for each voxel that incurs significant memory and computational overhead. Therefore, directly minimizing

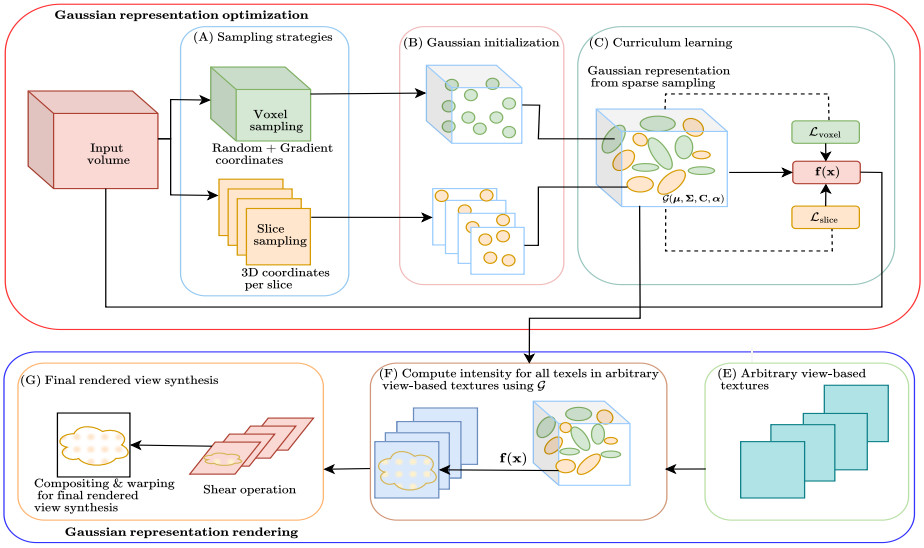

Fig. 1: Overview of the proposed Gaussian-based volumetric representation for real-time rendering. Sparse voxel and slice samples are used under a curriculum sampling strategy to optimize a continuous Gaussian volumetric representation, which is subsequently used to generate arbitrary view-based slice textures for shear–warp volume rendering.

the dense loss L_{dense} is impractical for large volumetric datasets where grid sizes of 512^3 are common.

A common alternative is to compute the loss using a subset of voxels sampled from the volume. One naive approach is to randomly sample a set of voxel positions in each epoch and compute the reconstruction loss only on these samples. That is, randomly sample $m \ll n$ voxel positions from a sampling distribution $q(\mathbf{x})$ defined over Ω. In practice, this sampled voxel set corresponds to the sparse supervision set $\mathcal{M}$ introduced in Section 3.2. The resulting loss can then be estimated as

$$\hat{L}_{\text{naive}} = \frac{1}{m} \sum_{i=1}^{m} \ell(\mathbf{x}_i; \Theta), \qquad \mathbf{x}_i \sim q(\mathbf{x}). \tag{3}$$

This estimator matches the dense objective when the sampling distribution is uniform, i.e., $q(\mathbf{x}) = \frac{1}{n}$. While uniform sampling produces an unbiased estimate, it is often inefficient for volumetric data since large regions may contain redundant or low-information voxels. In practice, it is beneficial to sample informative voxels more frequently, such as those with large reconstruction error or high spatial gradients. Therefore, such random subsampling does not guarantee that the estimated loss corresponds to the dense loss, particularly when the sampling distribution is non-uniform or when informative regions of the volume are sampled more frequently than others. In these cases, the optimization effectively

minimizes a modified objective that depends on the sampling distribution rather than the actual volumetric reconstruction error.

Monte Carlo estimation [20] provides a principled framework for approximating the dense volumetric objective using a sparse subset of samples while preserving consistency with the original objective. By incorporating appropriate importance weights based on the sampling distribution, the Monte Carlo estimator produces an unbiased estimate of the dense loss. Consequently, the expected value of the estimated loss and its gradients match those of the dense loss, allowing stochastic optimization to converge to the same optimum while significantly reducing computational cost. A naive subsampling from a non-uniform distribution leads to a biased estimator of the dense objective. To correct this, we use importance-weighted Monte Carlo estimation. The Monte Carlo estimator is defined as

$$\hat{L}_{\mathrm{MC}} = \frac{1}{m}\sum_{i=1}^{m} \frac{1}{n\, p(\mathbf{x}_i)} \ell(\mathbf{x}_i; \Theta), \qquad \mathbf{x}_i \sim p(\mathbf{x}). \tag{4}$$

Taking expectation with respect to $p(\mathbf{x})$,

$$\mathbb{E}[\hat{L}_{\mathrm{MC}}] = \frac{1}{n}\sum_{\mathbf{x}\in\Omega} \ell(\mathbf{x}; \Theta) = L_{\mathrm{dense}}(\Theta), \tag{5}$$

showing that the estimator is unbiased. The corresponding gradient estimator is

$$\nabla_\Theta \hat{L}_{\mathrm{MC}} = \frac{1}{m}\sum_{i=1}^{m} \frac{1}{n\, p(\mathbf{x}_i)} \nabla_\Theta \ell(\mathbf{x}_i; \Theta), \tag{6}$$

that satisfies $\mathbb{E}\big[\nabla_\Theta \hat{L}_{\mathrm{MC}}\big] = \nabla_\Theta L_{\mathrm{dense}}(\Theta)$.

Thus, stochastic gradient descent using this estimator converges to the same optimum as dense training under standard assumptions. In practice, informative voxels often correspond to regions with high reconstruction error or strong spatial gradients. Sampling such voxels more frequently improves optimization efficiency, which naturally leads to an importance sampling formulation where the sampling probability $p(\mathbf{x})$ is proportional to a measure of voxel information.

$$p(\mathbf{x}) = \lambda \frac{1}{n} + (1-\lambda)\, p_{\mathrm{importance}}(\mathbf{x}), \tag{7}$$

where $\lambda \in [0, 1]$ balances uniform coverage and importance-driven sampling. The importance distribution is estimated from the reconstruction error

$$p_{\mathrm{importance}}(\mathbf{x}) \propto \ell(\mathbf{x}; \Theta) + \epsilon, \tag{8}$$

where $\epsilon > 0$ ensures non-zero probability.

This strategy prioritizes high-error regions while maintaining unbiased estimation and global coverage of the volume. Reconstruction error directly depends

on the correctness of the boundary of various features in the volume and thus can be represented by high-gradient samples in the volume. A subset of voxels with high gradient magnitude are selected based on their reconstruction error, while another subset of voxels is sampled randomly from the volume. Both types of samples are periodically refreshed every few training iterations. The details of our sampling and the corresponding probability estimates are given in the supplementary material.

3.3 Curriculum sampling strategy for structured supervision

Our approach targets efficient shear-warp volume rendering, which composites stacks of 2D slices sampled from a 3D volume. To learn the Gaussian volumetric field while keeping training efficient, we use two complementary supervision techniques which include sparse voxel samples drawn for Monte Carlo volumetric estimation (Section 3.2) and pixel samples on randomly oriented slice planes. We optimize the Gaussian field parameters Θ using sparse voxel coordinates $\mathbf{x} \in \Omega$ sampled from the Monte Carlo distribution $p(\mathbf{x})$ (Section 3.2). For slice-based supervision, instead of computing the loss over the entire image grid, we evaluate it only on a subset of sampled locations on the slice. In each iteration, we additionally sample a plane with arbitrary orientation, and select sparse 2D pixel locations on that plane. Each sampled pixel corresponds to a 3D coordinate $\mathbf{x}$ in the volume, and the set of samples from the same plane provides supervision on multiple points that share the same plane. Compared to isolated voxel samples, these planar samples introduce structured constraints that encourage spatial coherence within each slice plane, and thus directly improves the fidelity of the generated slices later used by shear-warp compositing. Both voxel and slice supervision update the same Gaussian volumetric function, the difference lies only in how the supervised 3D coordinates $\mathbf{x}$ are sampled and paired with ground-truth color $\mathbf{y}(\mathbf{x})$. This approach of sampling of coordinates from the slices enables the Gaussians to learn better spatial correlation while maintaining the sparsity of the input, which further makes this technique highly memory efficient.

Rather than switching supervision abruptly, we blend the two losses with weights that vary over training progress. The weights $\lambda_v(t)$ and $\lambda_s(t)$ are scheduled over the training iteration t to gradually introduce slice supervision while maintaining strong voxel supervision during early optimization. To gradually introduce planar supervision during training, we define a smooth activation schedule for the slice loss

$$\lambda_s(t) = \lambda_{s,\max}\sigma\left(\frac{t-\gamma_s}{\beta_s}\right), \qquad \lambda_v(t) = 1 - \lambda_s(t), \tag{9}$$

where $\sigma(\cdot)$ denotes the sigmoid function for curriculum learning [37], $\lambda_{s,\max}$ is the maximum weight assigned to the slice loss, γ_s represents the iteration at which slice supervision becomes significant, and β_s controls the smoothness of

the transition. The voxel loss weight $\lambda_v(t)$ can be defined similarly or gradually decreased during training.

$$\mathcal{L}(t) = \lambda_v(t)\,\hat{L}_{\mathrm{MC}} + \lambda_s(t)\,L_{\mathrm{slice}}, \tag{10}$$

where $\mathcal{L}(t)$ denotes the total loss at iteration t. The term L_{slice} represents the slice supervision loss computed on pixel coordinates sampled from slices. For a set of sampled slice pixels $\mathcal{S}$, the slice loss is defined as

$$L_{\mathrm{slice}} = \frac{1}{|\mathcal{S}|} \sum_{\mathbf{x}\in\mathcal{S}} \|\mathbf{f}(\mathbf{x}) - \mathbf{y}(\mathbf{x})\|_2^2,$$

where $\mathbf{y}(\mathbf{x})$ denotes the ground-truth voxel color at $\mathbf{x}$.

Although pixels are sampled from 2D slice coordinates, each pixel corresponds to a 3D voxel location within the volume. Therefore, both voxel and slice supervision update the same Gaussian parameters through gradients of the predicted volumetric function $\mathbf{f}_\Theta$. This curriculum strategy combines sparse voxel supervision for global volume coverage with structured planar sampling that provides spatially correlated supervision across slices. Together, these complementary sampling strategies improve the learning of spatial coherence and geometric structures within the Gaussian volumetric representation.

3.4 Shear–warp volume rendering using Gaussian representation

To enable efficient real-time visualization of the learned Gaussian volumetric representation, we adopt a shear–warp volume rendering [15] strategy. Shear–warp rendering is chosen due to its computational efficiency for slice-based compositing and its suitability for structured 2D slice extraction from 3D volumes. Conventional shear–warp rendering takes as input a voxel grid and extracts three stacks of slices aligned with the principal axes of the volume. For an arbitrary viewing direction, a shear transformation is applied to align the viewing rays perpendicular to the selected principal stack, followed by slice-by-slice compositing. The ray marching [8, 12] algorithm is a prevalent real-time rendering technique that involves sampling along every ray passing through the volume, and each ray corresponds to a pixel position on the rendered image from a camera viewing direction. The sampling operation is followed by interpolation for all the samples along all the rays for a single rendered view. These operations are computationally expensive for higher-resolution rendered views and larger volumetric data. While efficient, this approach assumes a voxel representation which is dense by nature and requires maintaining axis-aligned slice stacks. In contrast, our approach integrates shear–warp rendering directly with the Gaussian volumetric representation. Our GPU-based renderer makes use of the fact that the Gaussians are not oriented along principal axes (unlike voxels) and therefore does not need to maintain three stacks of parallel slices but just one. We sample slice textures from the Gaussian representation along the current viewing direction,

shear and combine these using the standard front-to-back alpha compositing scheme to generate an image. The stack is reused for subsequent frames as long as the camera orientation remains within a specified angular threshold. When the camera rotates beyond this threshold, the slice stack is refreshed and re-oriented to match the new viewing direction. This adaptive stacking strategy reduces memory consumption by avoiding multiple precomputed stacks while maintaining rendering efficiency.

The existing Gaussian Splatting-based renderers are mainly surface-oriented, while our goal is internal volume rendering. We therefore use shear-warp rendering, which supports slice-based compositing of volumetric interiors and is computationally cheaper than ray marching. Its slice-based memory access and incremental compositing strategy enable high rendering throughput. On the Cryosection dataset, our method achieves an average rendering speed of 43.86 FPS, compared with 15.05 FPS for ray marching achieving a 2.91× speedup. Table 1 reports organ-wise frame rates and volume compression. These results show that the Gaussian-based shear-warp formulation enables efficient real-time rendering while preserving structural fidelity and maintaining a low memory footprint.

4 Implementation details, results and analysis

We evaluate our approach on volumetric datasets from multiple imaging modalities, including MRI and Cryosection volumes. For grayscale MRI data, we evaluate performance on the T2 BraTS dataset [23] and the MRI dataset of embryonic and neonatal mouse [28]. The color dataset used in our experiments is the Cryosection dataset from the Visible Korean Project [27]. We further evaluate our approach on multiple organs from this dataset, including lungs, liver, heart, and brain, where the organ-level volumes are extracted using the provided segmentation masks. The ground-truth supervision consists of a sparse voxel set sampled from the dense voxel grid using a hybrid strategy based on gradient magnitude and random sampling. We select the sparse voxel set such that the total memory footprint of the Gaussian representations remains within 25% of the original volumetric data size. Within this sparse set, 60% of voxels are selected from high gradient magnitude regions, while the remaining 40% are randomly sampled across the volume. Optimization is performed using the Adam optimizer with a learning rate of 5×10^{-4}, and the Gaussian parameters are trained for 120 epochs. We evaluate reconstruction accuracy by comparing the Gaussian-reconstructed volume with the original dense voxel grid. Rendering quality is evaluated by comparing shear–warp rendered images generated from the Gaussian representation with rendered images from the dense voxel grid. The reconstruction accuracy and rendering quality are analyzed for both the reconstructed voxel grid and arbitrary rendered views obtained from the Gaussian representation.

4.1 Comparisons

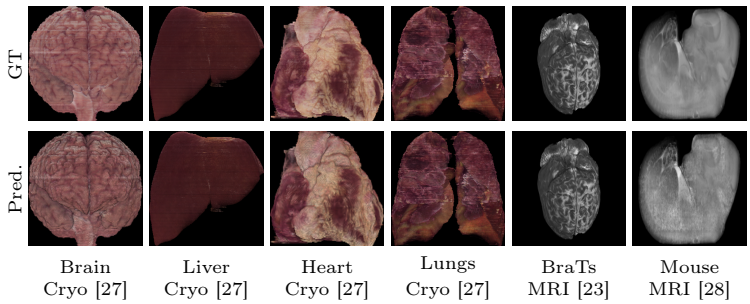

Fig. 2: Visual comparison of rendered views for different organs, including brain, liver, heart, and lungs from Cryosection data, BraTS MRI data, and mouse neonatal MRI data.

Datasets The evaluation metrics are computed for grayscale MRI modalities, including the BraTS [23] dataset and the embryonic–neonatal mouse MRI dataset. The RGB modality corresponds to the Cryosection dataset, along with organ-level volumes including lungs, liver, heart, and brain. The qualitative analysis of our approach on the BraTS dataset is shown in Fig. 2. The visual results illustrate the dense volume reconstructed from the sparse Gaussian representation. It can be observed that the proposed technique successfully generates rendered views using the shear–warp rendering algorithm from the Gaussian-based volumetric representation. Furthermore, the method efficiently preserves structural details in both the rendered views and volumetric cross-sectional slices across all modalities. Performance is quantified using Mean Squared Error (MSE) [31], Peak Signal-to-Noise Ratio (PSNR) [31], Structural Similarity Index (SSIM) [42], and Multi-Scale SSIM (MS-SSIM) [42] as shown in Table 1. The results demonstrate that our technique consistently achieves low MSE values and high PSNR, SSIM, and MS-SSIM scores across all modalities, indicating that the sparse Gaussian representation effectively captures the structural and textural characteristics of the target volumes.

Sparse representations We compare our approach with several high-capacity deep learning models used for learning complex data patterns, including MLPs [24], Transformers [36], and UNet [4]. These models rely on large numbers of learnable parameters and are evaluated for reconstructing dense voxel grids from sparse voxel observations. As shown in Fig. 3, we evaluate MLP-based representations using two supervision strategies: voxel-based and slice-based supervision. In voxel supervision, the MLP maps sparse voxel coordinates sampled from the dense volume to their corresponding color intensities in the Cryosection data. In slice supervision, sparse voxel coordinates are sampled from cross-sectional planes and mapped to the color intensities of those planes. In both settings, the MLP fails to preserve fine structural details and struggles to reconstruct high-quality rendered views. For the Transformer-based [36] representation, we train

Table 1: Quantitative analysis on different datasets.

Dataset	PSNR	SSIM	MS-SSIM	FPS	Compression ratio
Brain [27]	32.10	0.943	0.991	45.5	9.34:1
Liver [27]	43.16	0.981	0.996	50.18	9.45:1
Heart [27]	32.19	0.958	0.990	48.57	10.09:1
Lungs [27]	34.93	0.952	0.989	39.77	4.73:1
BraTS [23]	40.88	0.988	0.998	44.82	11.31:1
Mouse Neonatal [28]	33.48	0.956	0.987	46.56	9.94:1

the model using sparse $8 \times 8 \times 8$ 3D patches sampled from the volume. However, the Transformer struggles to preserve fine anatomical structures and introduces synthetic artifacts in regions with weak anatomical signals. For the UNet-based approach [4], we use a local 3D voxel neighborhood of size $8 \times 8 \times 8$ and predict voxel intensities using 3D convolutional layers. While this captures local context, 3D convolutions increase computational overhead, and the model underperforms in capturing complex color variations and anatomical structures. We also include GNT-MOVE [6], CVT-xRF [41], and FreeSplatter [39] as recent Transformer, NeRF, and Gaussian Splatting-based reconstruction methods. These approaches are representative of modern high-capacity rendering architectures that use attention mechanisms, radiance-field modeling, or Gaussian-based scene representations to reconstruct complex visual data. Since these methods were not originally designed for sparse volumetric reconstruction of dense Cryosection data, we adapt their input representations, training hyperparameters, and minor architectural settings to fit our problem while keeping their core model designs unchanged. As shown in Fig. 3, they do not consistently preserve fine internal anatomical structures or dense color variations in the Cryosection volume, whereas our approach directly fits a compact volumetric representation for internal 3D visualization. We further evaluate reconstruction and rendering accuracy using PSNR, MSE, SSIM, and MS-SSIM, as shown in Table 2. Our Gaussian-based representation consistently achieves lower MSE and higher PSNR, SSIM, and MS-SSIM than the deep learning baselines, indicating better preservation of structural and textural details. The table also reports efficiency in terms of training time, memory usage, FLOPs, and FPS, which are critical for real-time volumetric visualization. Compared with neural baselines, our method achieves high visual fidelity with substantially lower computational cost, demonstrating that the sparse Gaussian representation supports efficient reconstruction and real-time rendering. †FPS is not reported for GNT-MOVE [6] because its rendering pipeline requires view-dependent neural inference with a large number of parameters, making real-time rendering infeasible under our evaluation setup.

4.2 Ablation study

We perform an ablation study to analyze the contribution of the proposed sampling strategy and curriculum learning–based supervision and evaluate reconstruction quality using PSNR, MSE, SSIM, and MS-SSIM.

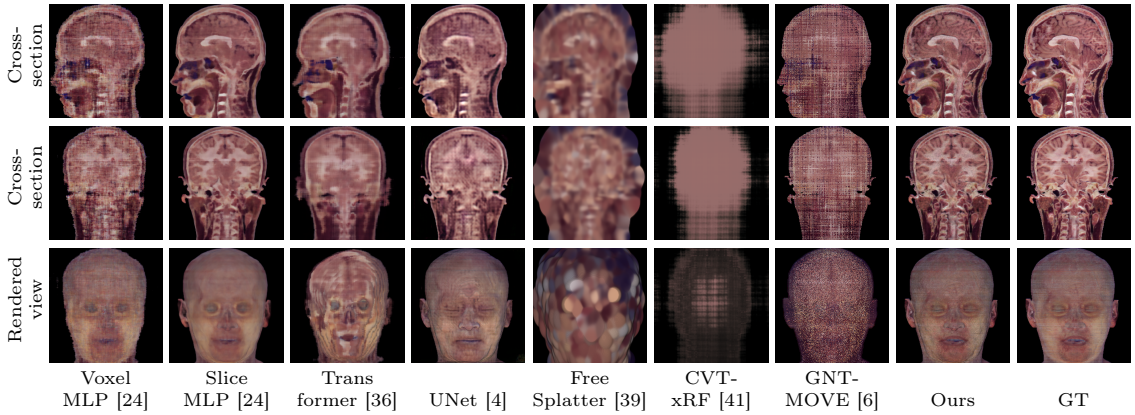

Fig. 3: Comparisons with learning-based volumetric representation methods.

Table 2: Quantitative comparison with baselines. Top results are highlighted as **1st**, 2nd, and 3rd.

Method	Reconstruction				Rendering				Efficiency				
	PSNR↑	MSE↓	SSIM↑	MS-SSIM↑	PSNR↑	MSE↓	SSIM↑	MS-SSIM↑	Epochs	Time (hours)	Mem (GB)	FLOPs	FPS
Voxel MLP [24]	23.63	0.000434	0.772	0.871	46.610	0.000176	0.968	0.980	200	6.7	15	9.50×10^{12}	18.99
Slice MLP [24]	30.02	0.000995	0.920	0.969	36.32	0.000248	0.973	0.983	300	35.0	14	1.88×10^{13}	14.27
Transformer [36]	9.60	0.052000	0.631	0.404	29.73	0.014900	0.873	0.481	200	33.3	30	1.91×10^{15}	0.11
UNet [4]	26.70	0.002130	0.798	0.943	42.10	0.000520	0.982	0.985	200	16.7	36	2.05×10^{12}	15.00
CVT-xRF [41]	18.98	0.012657	0.591	0.560	15.09	0.033922	0.536	0.551	500	19.3	22	2.84×10^{15}	30.17
FreeSplatter [39]	14.74	0.033608	0.483	0.515	16.24	0.023822	0.569	0.426	700	**0.13**	**11**	1.27×10^{12}	36.01
GNT-MOVE [6]	11.28	0.074417	0.046	0.317	15.11	0.030866	0.084	0.413	300	39.2	12	1.05×10^{15}	N/A†
Ours	**34.81**	**0.000398**	**0.956**	**0.993**	**54.87**	**0.000021**	**0.995**	**0.993**	**120**	**6.0**	28	9.17×10^{9}	**43.86**

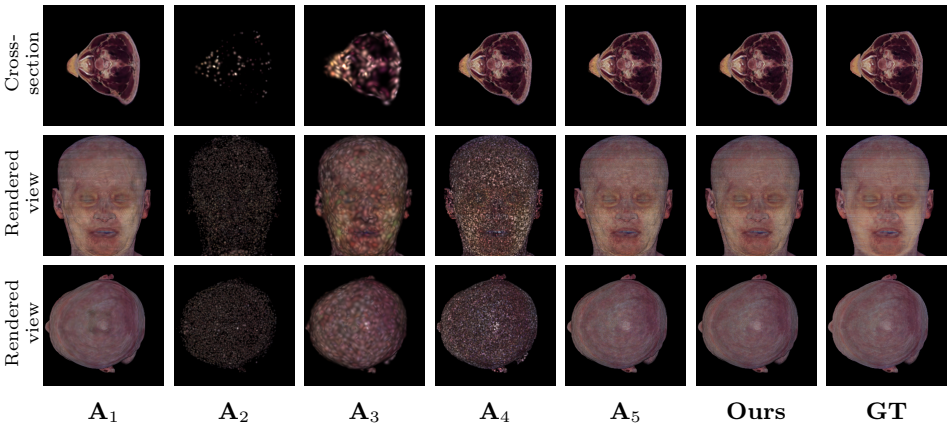

Fig. 4: Visual comparisons of the ablation study on cross sections and rendered views from the shear-warp renderer.

Table 3: Quantitative results of ablation on reconstruction and rendering quality.

Method	Reconstruction quality			Rendering quality		
	PSNR↑	SSIM↑	MS-SSIM↑	PSNR↑	SSIM↑	MS-SSIM↑
$\mathbf{A}_1$	33.46	0.956	0.990	50.7	0.962	0.964
$\mathbf{A}_2$	12.16	0.060	0.214	27.10	0.826	0.789
$\mathbf{A}_3$	22.64	0.727	0.850	40.59	0.932	0.937
$\mathbf{A}_4$	14.350	0.646	0.512	33.18	0.878	0.887
$\mathbf{A}_5$	32.40	0.943	0.988	52.34	0.968	0.991
Ours	**34.81**	**0.956**	**0.993**	**54.870**	**0.995**	**0.993**

$\mathbf{A}_1$(Random voxel supervision): In this experiment, we optimize the Gaussian representation using randomly sampled voxel coordinates together with high-gradient-magnitude voxels. At each iteration, a new set of voxel samples is selected, and the reconstruction loss is computed against the dense volume. In contrast, our approach initializes Gaussians over a sparse voxel set and approximates the dense volume using MCE. Our method captures finer structural details in both reconstructed volumes and rendered views. The evaluation metrics further confirm improved reconstruction under sparse voxel supervision.

$\mathbf{A}_2$(Curriculum learning with random voxel then random slices): This setting introduces slice supervision together with voxel supervision under random sampling during Gaussian optimization. While slice supervision progressively incorporates finer structural details through curriculum learning, random voxel sampling leads to unstable global approximation and further destabilizes the learning of Gaussian parameters.

$\mathbf{A}_3$ (Curriculum learning random slices then random voxel): In this ablation, slice supervision is applied first, followed by voxel supervision under random sampling with dense approximation. Since slice supervision is introduced before stable global voxel constraints are established, the Gaussian parameters are initially optimized using limited planar information. This results in unstable learning of the volumetric structure, and the later introduction of random voxels does not sufficiently correct the global approximation.

$\mathbf{A}_4$ (Only slice supervision): This configuration uses only slice supervision without voxel supervision. Although slices provide structured planar constraints and capture local spatial continuity, they cover limited regions of the volume at each iteration. Without sparse voxel supervision across the volume, the Gaussian representation lacks global spatial guidance. This limits accurate reconstruction of the full volumetric structure.

$\mathbf{A}_5$ (MCE voxel based supervision): In this setting, MCE is used for voxel sampling while optimizing Gaussian parameters using voxel supervision only. Importance-weighted sampling improves coverage of informative regions while remaining consistent with the dense reconstruction objective. This updates Gaussian parameters using more representative voxel samples and stabilizes optimization. However, without slice supervision, the model lacks structured planar constraints and shows lower performance in rendered views than our method.

5 Limitations

The proposed representation has limitations due to the smooth nature of Gaussian primitives. Since each primitive models the signal using a smooth spatial kernel, the reconstructed volumetric field may exhibit slight blurring near sharp boundaries or high-frequency details. This behavior is inherent to Gaussian-based representations that approximate volumetric signals using smooth basis functions. Additionally, small bright dot-like artifacts in certain regions of the reconstructed volume appear occasionally. These artifacts likely arise from the highly sparse supervision used during optimization, where only a limited subset of voxel samples guides the Gaussian parameter updates. Although the current framework produces visually consistent reconstructions even under strong sparsity constraints (as shown in the supplementary material), improved sampling strategies or regularization could further reduce these artifacts in future work.

6 Conclusion

In this work, we present a Gaussian-based volumetric representation for efficient reconstruction and visualization of high-resolution medical volumes. The volume is modeled as a continuous Gaussian field that can be evaluated at arbitrary spatial coordinates, enabling dense reconstruction from sparse voxel supervision. To optimize this representation, we use Monte Carlo volumetric estimation to approximate the dense reconstruction objective using a sparse voxel subset while remaining consistent with the full objective.We further introduce a curriculum learning–based sampling strategy that combines sparse voxel samples with planar slice samples during training. While voxel samples provide global volumetric coverage, slice supervision introduces structured spatial constraints that improve the learning of anatomical structures. The learned Gaussian representation supports efficient shear–warp volume rendering for high-quality visualization of volumetric medical data across modalities such as MRI and Cryosection. The effectiveness of our approach is demonstrated across multiple modalities, compared with learning-based volumetric reconstruction methods, and validated through comprehensive ablation studies.

Acknowledgements

This research was supported by the Science and Engineering Research Board (SERB) of the Department of Science and Technology (DST) of India (Grant No. CRG/2020/005792). The Visible Korean Human dataset for our research was provided by the Korea Institute of Science and Technology Information (KISTI), South Korea.

References

1. Ahmed, N., Natarajan, T., Rao, K.R.: Discrete cosine transform. IEEE transactions on Computers **100**(1), 90–93 (1974)

2. Chang, Z., Jin, H., Song, Y., Sun, Y., Yu, H.: GAT-NeRF: Geometry-aware-transformer enhanced neural radiance fields for high-fidelity 4d facial avatars. ACM Transactions on Multimedia Computing, Communications and Applications (2026)
3. Chen, Y., Wang, X.: Transformers as meta-learners for implicit neural representations. In: European Conference on Computer Vision. pp. 170–187. Springer (2022)
4. Çiçek, Ö., Abdulkadir, A., Lienkamp, S.S., Brox, T., Ronneberger, O.: 3D U-Net: learning dense volumetric segmentation from sparse annotation. In: International conference on medical image computing and computer-assisted intervention. pp. 424–432. Springer (2016)
5. Condor, J., Speierer, S., Bode, L., Bozic, A., Green, S., Didyk, P., Jarabo, A.: Don't Splat your Gaussians: Volumetric Ray-Traced Primitives for Modeling and Rendering Scattering and Emissive Media. ACM Transactions on Graphics **44**(1), 1–17 (2025)
6. Cong, W., Liang, H., Wang, P., Fan, Z., Chen, T., Varma, M., Wang, Y., Wang, Z.: Enhancing NeRF akin to Enhancing LLMs: Generalizable NeRF Transformer with Mixture-of-View-Experts. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 3193–3204 (2023)
7. Corona-Figueroa, A., Frawley, J., Bond-Taylor, S., Bethapudi, S., Shum, H.P., Willcocks, C.G.: Mednerf: Medical neural radiance fields for reconstructing 3d-aware ct-projections from a single X-ray. In: 2022 44th annual international conference of the IEEE engineering in medicine & Biology society (EMBC). pp. 3843–3848. IEEE (2022)
8. Devkota, S., Pattanaik, S.: Deep learning based super-resolution for medical volume visualization with direct volume rendering. In: International Symposium on Visual Computing. pp. 103–114. Springer (2022)
9. Duan, Y., Zhu, H., Wang, H., Yi, L., Nevatia, R., Guibas, L.J.: Curriculum deepsdf. In: European Conference on Computer Vision. pp. 51–67. Springer (2020)
10. Kerbl, B., Kopanas, G., Leimkühler, T., Drettakis, G.: 3D Gaussian splatting for real-time radiance field rendering (2023), `https://arxiv.org/abs/2308.04079`
11. Khojasteh, S.B., Fuentes-Jimenez, D., Pizarro, D., Espinel, Y., Bartoli, A.: MIS-NeRF: neural radiance fields in minimally-invasive surgery. International Journal of Computer Assisted Radiology and Surgery **20**(7), 1481–1490 (2025)
12. Kim, S., Jang, Y., Kim, S.E.: Image-based TF colorization with CNN for direct volume rendering. IEEE Access **9**, 124281–124294 (2021)
13. Kleinbeck, C., Schieber, H., Engel, K., Gutjahr, R., Roth, D.: Multi-layer gaussian splatting for immersive anatomy visualization. IEEE Transactions on Visualization and Computer Graphics (2025)
14. Kolda, T.G., Bader, B.W.: Tensor decompositions and applications. SIAM review **51**(3), 455–500 (2009)
15. Lacroute, P., Levoy, M.: Fast volume rendering using a shear-warp factorization of the viewing transformation. In: Proceedings of the 21st annual conference on Computer graphics and interactive techniques. pp. 451–458 (1994)
16. Li, C., Tong, Y., Chen, K., Yang, Z., Li, R., Qiu, S., Chan, J.Y.K., Heng, P.A., Dou, Q.: Clipgs: Clippable gaussian splatting for interactive cinematic visualization of volumetric medical data. arXiv preprint arXiv:2507.06647 (2025)
17. Liang, Y., He, H., Chen, Y.: Retr: Modeling rendering via transformer for generalizable neural surface reconstruction. Advances in neural information processing systems **36**, 62332–62351 (2023)
18. Liang, Z., Zhang, Q., Hu, W., Zhu, L., Feng, Y., Jia, K.: Analytic-splatting: Anti-aliased 3D Gaussian splatting via analytic integration. In: European conference on computer vision. pp. 281–297. Springer (2024)

19. Lin, K.E., Lin, Y.C., Lai, W.S., Lin, T.Y., Shih, Y.C., Ramamoorthi, R.: Vision transformer for nerf-based view synthesis from a single input image. In: Proceedings of the IEEE/CVF winter conference on applications of computer vision. pp. 806–815 (2023)
20. Luengo, D., Martino, L., Bugallo, M., Elvira, V., Särkkä, S.: A survey of Monte Carlo methods for parameter estimation. EURASIP Journal on Advances in Signal Processing **2020**(1), 25 (2020)
21. Lyu, J., Ling, S.H., Banerjee, S., Zheng, J., Lai, K.L., Yang, D., Zheng, Y.P., Bi, X., Su, S., Chamoli, U.: Ultrasound volume projection image quality selection by ranking from convolutional RankNet. Computerized Medical Imaging and Graphics **89**, 101847 (2021)
22. Meagher, D.: Geometric modeling using octree encoding. Computer graphics and image processing **19**(2), 129–147 (1982)
23. Menze, B.H., Jakab, A., Bauer, S., Kalpathy-Cramer, J., Farahani, K., Kirby, J., Burren, Y., Porz, N., Slotboom, J., Wiest, R., Lanczi, L., Gerstner, E., Weber, M.A., Arbel, T., Avants, B.B., Ayache, N., Buendia, P., Collins, D.L., Cordier, N., Corso, J.J., Criminisi, A., Das, T., Delingette, H., Demiralp, c., Durst, C.R., Dojat, M., Doyle, S., Festa, J., Forbes, F., Geremia, E., Glocker, B., Golland, P., Guo, X., Hamamci, A., Iftekharuddin, K.M., Jena, R., John, N.M., Konukoglu, E., Lashkari, D., Mariz, J.A., Meier, R., Pereira, S., Precup, D., Price, S.J., Raviv, T.R., Reza, S.M.S., Ryan, M., Sarikaya, D., Schwartz, L., Shin, H.C., Shotton, J., Silva, C.A., Sousa, N., Subbanna, N.K., Szekely, G., Taylor, T.J., Thomas, O.M., Tustison, N.J., Unal, G., Vasseur, F., Wintermark, M., Ye, D.H., Zhao, L., Zhao, B., Zikic, D., Prastawa, M., Reyes, M., Van Leemput, K.: The Multimodal Brain Tumor Image Segmentation Benchmark (BRATS). IEEE Transactions on Medical Imaging **34**(10), 1993–2024 (2015). https://doi.org/10.1109/TMI.2014.2377694
24. 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 **65**(1), 99–106 (2021)
25. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: NeRF: representing scenes as neural radiance fields for view synthesis. Commun. ACM **65**(1), 99–106 (Dec 2021). https://doi.org/10.1145/3503250, https://doi.org/10.1145/3503250
26. Müller, T., Mcwilliams, B., Rousselle, F., Gross, M., Novák, J.: Neural importance sampling. ACM Trans. Graph. **38**(5) (Oct 2019). https://doi.org/10.1145/3341156, https://doi.org/10.1145/3341156
27. Park, J.S., Chung, M.S., Hwang, S.B., Lee, Y.S., Har, D.H., Park, H.S.: Visible Korean human: improved serially sectioned images of the entire body. IEEE transactions on medical imaging **24**(3), 352–360 (2005)
28. Petiet, A.E., Kaufman, M.H., Goddeeris, M.M., Brandenburg, J., Elmore, S.A., Johnson, G.A.: High-resolution magnetic resonance histology of the embryonic and neonatal mouse: A 4D atlas and morphologic database. Proceedings of the National Academy of Sciences **105**(34), 12331–12336 (2008). https://doi.org/10.1073/pnas.0805747105, https://www.pnas.org/doi/abs/10.1073/pnas.0805747105
29. Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015. pp. 234–241. Springer International Publishing, Cham (2015)
30. Rückert, D., Wang, Y., Li, R., Idoughi, R., Heidrich, W.: Neat: Neural adaptive tomography. ACM Transactions on Graphics (TOG) **41**(4), 1–13 (2022)

31. Sara, U., Akter, M., Uddin, M.S., et al.: Image quality assessment through FSIM, SSIM, MSE and PSNR—a comparative study. Journal of Computer and Communications **7**(3), 8–18 (2019)
32. Su, Y., Wang, S., Wang, H.: Dt-nerf: Decomposed triplane-hash neural radiance fields for high-fidelity talking portrait synthesis. In: ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 3975–3979. IEEE (2024)
33. Tang, K., Yao, S., Wang, C.: iVR-GS: Inverse volume rendering for explorable visualization via editable 3D Gaussian splatting. IEEE Transactions on Visualization and Computer Graphics (2025)
34. Tang, Z.J., Cham, T.J., Zhao, H.: Able-nerf: Attention-based rendering with learnable embeddings for neural radiance field. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 16559–16568 (2023)
35. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017)
36. Wang, D., Cui, X., Chen, X., Zou, Z., Shi, T., Salcudean, S., Wang, Z.J., Ward, R.: Multi-View 3D Reconstruction With Transformers. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 5722–5731 (2021)
37. Wang, Y., Gan, W., Yang, J., Wu, W., Yan, J.: Dynamic curriculum learning for imbalanced data classification. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 5017–5026 (2019)
38. Wen, Y., Zhang, K., Li, Z., Qiao, Y.: A discriminative feature learning approach for deep face recognition. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) Computer Vision – ECCV 2016. pp. 499–515. Springer International Publishing, Cham (2016)
39. Xu, J., Gao, S., Shan, Y.: FreeSplatter: Pose-Free Gaussian Splatting for Sparse-View 3D Reconstruction. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) (2025)
40. Zha, R., Zhang, Y., Li, H.: NAF: neural attenuation fields for sparse-view cbct reconstruction. In: International Conference on Medical Image Computing and Computer-Assisted Intervention. pp. 442–452. Springer (2022)
41. Zhong, Y., Hong, L., Li, Z., Xu, D.: CVT-xRF: Contrastive In-Voxel Transformer for 3D Consistent Radiance Fields from Sparse Inputs. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2024)
42. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: Proceedings of the IEEE international conference on computer vision. pp. 2223–2232 (2017)