NGR: Neural Gradient Rendering for High-Quality 3D Reconstruction from Multi-View Images

Xufan He[1] Dong Du[1*] Yushuang Wu[2] Yunbi Liu[3]
[1]Nanjing University of Science and Technology [2]ByteDance Inc.
[3]Nanjing University of Posts and Telecommunications

Abstract

Recent advances in neural volume techniques have significantly improved 3D shape reconstruction from multi-view images using both signed and unsigned distance fields (SDFs and UDFs). Despite this progress, existing methods still struggle to obtain accurate surfaces and require tedious parameter tuning, especially for non-watertight shapes represented by UDFs. To address this challenge, we observe that most approaches focus on converting distance values into volume density for rendering, but the distance fields change smoothly and almost linearly. In contrast, the gradients of these distance fields change abruptly at surface boundaries, making them more effective for capturing geometric details. Based on this insight, we propose a new volume rendering method based on gradient vectors, which is compatible with both SDF and UDF representations. Additionally, we reconsider the common assumption of unbiased properties in open-surface reconstruction and utilize the softplus function to improve UDF learning. Extensive experiments conducted on the DeepFashion3D, DTU, and BlendedMVS datasets demonstrate the effectiveness and robustness of our method. The code will be publicly available to facilitate future research.

Keywords: Multi-View Reconstruction, Neural Implicit Surface Learning, Gradient-Based Volume Rendering

1. Introduction

Multi-view 3D reconstruction is a fundamental task in computer vision and graphics. Traditional multi-view stereo methods [30, 7, 31] rely heavily on precise image correspondence, which often produces artifacts in complicated scenarios. Recently, the success of neural radiance fields (NeRF) [26] has introduced a new perspective for multi-view reconstruction. By leveraging differentiable neural volume rendering, the underlying shapes can be robustly learned through backpropagation by minimizing the discrepancy between rendered and real multi-view images. Many subsequent works further improve the performance of reconstructed geometry by integrating neural volume rendering with implicit representations based on SDF/UDF (such as NeuS [37], VolSDF [42], NeuralUDF [23], NeUDF [22]) or explicit representations based on 3DGS (such as GOF [46], PGSR [3]).

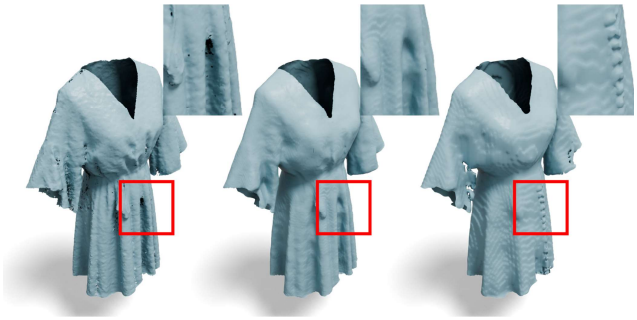


Figure 1: Visual comparisons of surface reconstruction on the DeepFashion3D [49] dataset using our gradient vector-based rendering approach (Ours) and the distance value-based rendering method (NeuralUDF [23]).

However, SDF- and 3DGS-based methods are suitable for reconstructing watertight objects but struggle to accurately represent shapes with open boundaries, often requiring tedious post-processing. In addition, existing neural volume rendering approaches face an intrinsic difficulty: they must convert distance fields into density functions while preserving both unbiasedness (ensuring the volume rendering weight function peaks exactly at the true surface location) and occlusion-awareness (determining whether a point is visible or occluded by a surface) [37, 23]. For example, NeuralUDF [23] uses complex transformation functions and advanced training strategies but still fails to produce consistently reliable results across diverse cases. Similarly, 2S-UDF [6] adopts a two-stage training pipeline to improve reconstruction quality, yet it remains insufficiently robust and frequently yields outputs that deviate substantially from the ground truth.

A key limitation of current methods is their direct conversion of distance fields into density fields, which vary

*Corresponding author: dongdu@njust.edu.cn

almost linearly and therefore struggle to capture fine geometric detail, especially around open surfaces. Existing density-conversion schemes cannot capture the sharp transitions at open boundaries, which leads to blurred details or distorted shapes. By contrast, we find that the gradient vectors of distance fields exhibit abrupt changes near object surfaces, and these gradient discontinuities correlate strongly with sharp geometric features. Motivated by this observation, we propose a new neural volume-rendering approach that leverages gradient vectors.

In this paper, we focus on multi-view reconstruction using neural implicit representations (i.e., SDF and UDF). We propose a novel and unified volume rendering framework applicable to both SDF and UDF. First, we introduce a new rendering formulation based on cumulative gradient differences, which alleviates the smoothing issues of existing approaches as illustrated in Fig. 1. Then, we tackle the harder problem of reconstructing open surfaces and report two key findings: 1) The unbiased property can sometimes misrepresent open surfaces, which actually represent a volumetric band with a certain thickness. For example, unbiased methods will fail when the surface is a single-layer mesh but has different colors on both sides, as shown in our experiments with two-sided colored objects. 2) The activation function used to enforce nonnegative UDF values is critical for open-surface reconstruction. We argue that only monotonic activations are appropriate because the commonly used absolute value induces double-layered artifacts. In this work, we adopt the softplus activation function.

Extensive experiments on various datasets (i.e., DeepFashion3D [49], DTU [12], and BlendedMVS [41]) demonstrate that our method achieves state-of-the-art performance on open surfaces while delivering results comparable to leading methods for closed surfaces, such as NeuS [37]. We also provide thorough ablation studies that validate the effectiveness of our key design choices. The code will be released to facilitate further research.

In summary, our main contributions are as follows:

- We propose a novel neural volume rendering method based on gradient vectors of implicit distance fields, applicable to both SDF and UDF.
- We critically examine the commonly used unbiased property and activation functions in UDF-based volume rendering, providing two key insights to enhance performance.
- We conduct comprehensive experiments on diverse datasets to demonstrate the effectiveness of our method, achieving state-of-the-art results on open object datasets and competitive performance on closed object datasets.

2. Related Work

Classical Multi-View Reconstruction. Traditional multi-view reconstruction methods primarily rely on image correspondence [30]. These approaches first estimate the underlying depth by analyzing feature correspondences between input images, then fuse the depth into a point cloud, and further extract a mesh shape using the ball-pivoting [2] or screened Poisson [16] method. However, feature correspondences between multiple images are sensitive to specular lighting and texture variations, limiting the reconstruction to high-quality images captured by expensive devices. Therefore, many methods based on end-to-end learning paradigms are proposed to improve the performance [13, 14, 15, 33]. Nevertheless, these learning-based methods inherently optimize a volumetric representation, leading to substantial memory and computational overhead, especially for high-resolution reconstruction.

Neural Implicit Representation. Various representations have been proposed for modeling 3D geometric surfaces. The occupancy field [25] represents a scalar field ranging from 0 to 1, indicating whether a given location is inside or outside an object. The signed distance field (SDF) [1] encodes geometry as the signed distance from a point to the nearest surface. However, both representations rely on the inside/outside concept, making them less suitable for reconstructing open surfaces. To address this limitation, approaches such as 3PSDF [4] and HSDF [36] modify SDF to enable open surface representation. Meanwhile, the unsigned distance field (UDF) [5] represents surfaces using unsigned distances, achieving promising results under 3D supervision. However, UDF lacks differentiability at the zero-level set, which hinders optimization. GDF [19] mitigates this issue by defining UDF as the norm of a vector field, ensuring differentiability at the surface while preserving the ability to represent open surfaces. Jäger et al. [11] utilize the gradients of the NeRF density field for edge detection and surface refinement, which focuses on enhancing feature extraction within the NeRF framework. Meanwhile, VF-NeRF [29] and NVF [40] represent objects using a vector field, with changes in vector direction indicating the presence of a surface. Similarly, our NGR fundamentally reformulates the volume rendering equation itself, utilizing the gradient of the SDF/UDF to directly derive the rendering weights, thereby unifying the representation of closed and open surfaces.

Neural Rendering Based Implicit Reconstruction. With the rise of neural implicit representations, multi-view reconstruction has achieved remarkable performance. Typically, implicit representations are optimized through differentiable rendering, either via volume rendering or surface rendering. Surface rendering methods [32, 27, 21, 20, 43,

17, 34] determine the color of a ray based on its intersection with the implicit surface. However, they struggle to reconstruct complex objects. In contrast, volume rendering methods [37, 42, 38, 39] compute the ray color by blending all sampled points along the ray, making optimization more stable and achieving better results. Further improvements incorporate monocular geometric cues [45, 35] and semantic information [10] to enhance reconstruction quality. Despite these advances, most methods rely on SDF or occupancy functions, limiting their ability to reconstruct open surfaces. Recently, UDF-based reconstruction methods [23, 22, 6] have emerged, demonstrating the capability of reconstructing open surfaces, but they still struggle to obtain accurate reconstruction with fine-grained details.

3. Method

Given a set of calibrated images $\{\mathcal{I}_k\}$ of a static object, our goal is to recover its geometric surface, represented as the zero-level set of a distance field (either UDF or SDF). In this section, we will present our gradient vector-based rendering method and describe its main components. Specifically, in Sec. 3.1, we introduce a new volume rendering formulation based on the gradient vectors of distance functions. In Sec. 3.2, we examine the differences between open and closed surfaces in the physical world and highlight why the unbiased property is not suitable for modeling single-layer open surfaces. In Sec. 3.3, we analyze how converting MLP outputs to UDF values affects topological flexibility, which is critical for accurately reconstructing open surfaces.

3.1. Neural Gradient rendering

We use the zero-level set of a distance field to represent the object surface $\mathcal{S} = \{\mathbf{x} \in \mathbb{R}^3 \mid f(\mathbf{x}) = 0\}$. Considering a ray $\mathbf{p}(t) = \mathbf{o} + t\mathbf{r}$ is emitted from a pixel ($t > 0$), $f(t)$ is the distance value at $\mathbf{p}(t)$, then the gradient vector field is defined as

$$\mathbf{v}(\mathbf{x}) = \frac{\partial f(t)}{\partial \mathbf{x}} \cdot \text{sign}(f(t)), \tag{1}$$

where the sign$(f(t))$ equals -1 or 1 for SDF and is always 1 for UDF. This definition ensures a significant change in the vector field near the zero-level set. In our experiments, $\mathbf{v}(\mathbf{x})$ is normalized, and the subsequent normalization of $\mathbf{v}(\mathbf{x})$ will not be elaborated for simplicity. Our goal is to transform the $\mathbf{v}(\mathbf{p}(t))$ into a density field $\sigma(t)$, and then use volume rendering to calculate the color

$$C(\mathbf{r}) = \int_0^{+\infty} T(t)\sigma(t)\mathbf{c}(\mathbf{p}(t), \mathbf{v}(\mathbf{p}(t)))\mathrm{d}t, \tag{2}$$

where $T(t) = \exp\left(-\int_0^t \sigma(u)\mathrm{d}u\right)$ denotes the accumulated transmittance along the ray, $\mathbf{c}$ denotes the color field [26]. The weight function for color blending is denoted as $w(t) = T(t)\sigma(t)$.

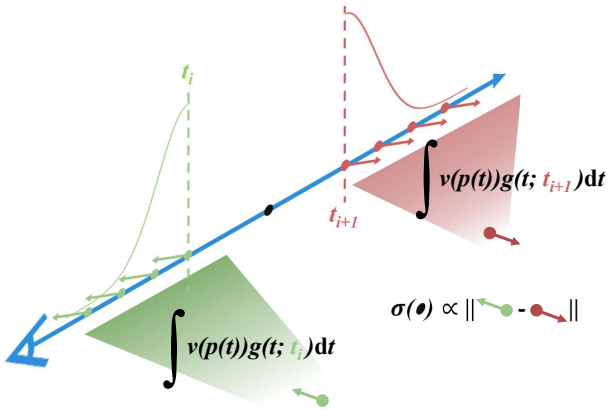


Figure 2: Visualization of our gradient vector-based rendering. We aggregate all the samples before and after the interval using a distribution $g(t; t^*) \sim \mathcal{N}(t^*, \gamma^2)$, where t^* can be either t_i or t_{i+1}. The difference between the aggregated gradients reflects the overall change in the gradient direction.

Naive Rendering Based on Gradient Difference. A naive idea for gradient rendering is to directly convert the difference of $\mathbf{v}(\mathbf{p}(t))$ into a density for neural rendering. That is, given the samples on the ray $\{\mathbf{p}(t_i) | i = 1, 2, \cdots, n\}$,

$$\sigma(t) \propto \frac{||\mathbf{v}(\mathbf{p}(t_{i+1})) - \mathbf{v}(\mathbf{p}(t_i))||_2}{t_{i+1} - t_i}, t \in [t_i, t_{i+1}). \tag{3}$$

However, in our experiments, this method is proven ineffective because it relies heavily on ray sampling, and the limited number of samples makes the gradients used for optimization unreliable.

Robust Rendering Based on Gradient Aggregate. Because the naive rendering scheme that uses only boundary points can yield unreliable gradient estimation from samples taken just before or after an interval, we aggregate all sampled vectors $\mathbf{v}(\mathbf{x})$ from both sides of the interval to evaluate the overall change in the vector direction. Specifically, given a sample $\mathbf{p}(t^*)$ on a ray, we define a normal distribution $g(t) \sim \mathcal{N}(t^*, \gamma^2)$ with a mean of t^* and a standard deviation of γ, then formulate the aggregated gradients before and after t^* as $\mathbf{V}_{left}(t^*)$ and $\mathbf{V}_{right}(t^*)$, i.e.,

$$\mathbf{V}_{left}(t^*) = \int_0^{t^*} \mathbf{v}(\mathbf{p}(t))g(t)\mathrm{d}t, \tag{4}$$

$$\mathbf{V}_{right}(t^*) = \int_{t^*}^{+\infty} \mathbf{v}(\mathbf{p}(t))g(t)\mathrm{d}t. \tag{5}$$

A visualization of the gradient aggregation is illustrated in Fig. 2. In our experiments, we further multiply a learnable coefficient w to enlarge the difference between $\mathbf{V}_{left}$ and $\mathbf{V}_{right}$, and mask out the gradient changes that do not occur

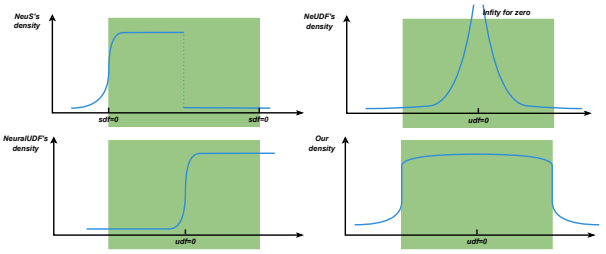


Figure 3: A ray is cast from left to right, passing through a thin volumetric band (highlighted in green). The curves illustrate the volume density profiles for NeuS [37], NeUDF [22], NeuralUDF [23], and our method.

near the zero-level set. The density is formulated as

$$\sigma(t) = w \cdot \mathbf{m}(t) \frac{||\mathbf{V}_{right}(t_{i+1}) - \mathbf{V}_{left}(t_i)||_2}{t_{i+1} - t_i}, t \in [t_i, t_{i+1}), \tag{6}$$

where $\mathbf{m}(t)$ is the corresponding mask. We set $\mathbf{m}(t) = e^{-sf(t)}$ and $\mathbf{m}(t) = \text{sigmoid}(-sf(t))$ for the representation of UDF and SDF, respectively.

3.2. Rethinking Unbiased Property of Weight

The unbiased property is introduced by NeuS [37], that the color blending weight $w(t)$ attains a locally maximum at a surface intersection point $\mathbf{p}(t^*)$, where $f(\mathbf{p}(t^*)) = 0$. It has also been adopted by recent UDF-based open surface reconstruction methods [23, 22, 6]. However, we argue that the property is originally formulated for reconstructing closed objects. In this subsection, we show that our method is universal and can still realize the unbiased property when necessary.

Why does reconstruction of closed surfaces need unbiased property? For shape reconstruction with distance field-based volume rendering, we need to define a function mapping the distance to density, which is applied to determine the weight for color blending. Representing the surface using the zero-level set, a straightforward approach is to define a symmetric distribution with a mean value of zero. However, for the same density values $\sigma_{t_1} = \sigma_{t_2}$, where $t_1 < t_2$, volume rendering inherently assigns more weight to $\mathbf{p}(t_1)$ for color blending. Consequently, the weight is concentrated before the zero-level set, causing the reconstructed mesh to appear shrunken compared to the ground truth.

As shown in Fig. 3, NeuS [37] introduces an asymmetric density function that is monotonically increasing in the neighborhood of the zero-level set. This density function ensures the weights concentrate at the zero-level set unbiasedly, thereby improving the reconstruction quality.

Why is the unbiased property not applicable to the reconstruction of open surfaces? The representation of objects using a zero-thickness level-set exhibits significant differences between open and closed surfaces. For closed surfaces, the zero-level set defines the ideal zero-thickness surface that separates the interior and exterior space. Therefore, achieving the unbiased property is crucial to improving the quality of reconstruction. However, for open surfaces, the zero-level set represents a thin volumetric band (consider a thick sweater), as shown in Fig. 3. This makes the unbiased property inapplicable. For example, enforcing the unbiased property would prevent the method from reconstructing a surface with different colors on each side, as shown in Fig. 6.

Our density near the zero-level set forms a plateau with a high value, similar to NeuS [37]. The thickness of the volumetric band is determined by γ, the deviation of the normal distribution $g(t)$. Large γ will hinder the reconstruction of closed surfaces, as it violates the unbiased rule. Instead of making γ trainable, for closed object reconstruction, we enforce it to gradually decrease during the training process until it reaches 0.001, i.e., $\gamma = 1/(1000x^3 + 100)$, where $x \in [0, 1]$ represents the training progress. In our experiments, we use the same fixed schedule across all datasets without per-scene tuning. The consistent performance across these diverse shapes demonstrates the robustness of this strategy. Besides, an excessively large w increases the weight for intervals with minimal gradient variation, thereby compromising the unbiased property. To mitigate this, a penalty term of w^2 is introduced to preserve the unbiased nature, as validated in Fig. 8.

3.3. Discussing Non-Negativity Property of UDF

The non-negativity of the UDF is often acquired by using a simple function (e.g., ReLU or the absolute function) in many methods. We argue that this makes the optimization of gradient descent methods struggle with changing geometric topology, see Theorem 1. This issue becomes critical when using the sphere initialization [1] and applying an absolute function for UDF, as illustrated in Fig. 7. We believe that this is the underlying reason for the common double-layer phenomenon in the reconstructed results of NeuralUDF [23]. Please refer to the supplementary material for a detailed discussion.

Theorem 1 *Given a scalar output from the MLP, $h(\mathbf{x}) \in \mathbb{R}$, where $\mathbf{x} \in \mathbb{R}^3$ is a point in space, we use the absolute function to ensure the non-negativity of the UDF, i.e., $f(\mathbf{x}) = |h(\mathbf{x})| \geq 0$. A closed zero-level set will exist as long as there exists $\mathbf{x_0} \neq \mathbf{x_1}$ such that $h(\mathbf{x_0})h(\mathbf{x_1}) < 0$.*

To ensure feasible topological optimization, the transformation from the network (e.g., MLP) outputs to the UDFs f should be monotonic everywhere. In this work, we adopt the differentiable ReLU function defined as $\frac{1}{\beta}\log(1+\exp^{\beta x})$ for the reconstruction of open surfaces.

3.4. Training

We follow NeuS [37], use an MLP with 8 hidden layers to model the distance field $f(\mathbf{x})$ and an MLP with 4 layers to model the view-dependent color field $\mathbf{c}(\mathbf{x}, \mathbf{v})$. The loss function is defined as

$$\mathcal{L} = \mathcal{L}_{color} + \lambda_1 \mathcal{L}_{eikonal} + \lambda_2 w^2, \tag{7}$$

where $\mathcal{L}_{color}$ is the L_1 loss between the rendered color and the ground truth, and the Eikonal loss [8] $\mathcal{L}_{eikonal}$ enforces $\left\| \frac{\partial f(\mathbf{x})}{\partial \mathbf{x}} \right\|_2 = 1$. The w^2 term prevents the coefficient from becoming extremely large and hindering the optimization progress. The loss weights are set as $\lambda_1 = 0.05, \lambda_2 = 1e-6$ for DTU [12] dataset and $\lambda_1 = 0.1, \lambda_2 = 1e-5$ for DeepFashion3D [49] dataset in our experiments. We do not perform per-scene tuning because the regularization term is not sensitive to specific object geometries.

For the training, we use the geometric initialization method proposed by SAL [1] to initialize the distance field, then adopt Adam [18] to optimize the MLP with a batch size of 512 for 300k iterations. All experiments are conducted without mask supervision, and the background is modeled using NeRF++[47]. Lastly, we utilize MeshUDF [9] and Marching Cubes [24] to extract the underlying meshes. Detailed information on the implementation and network architecture can be found in the supplementary material.

4. Experiments

4.1. Experiment Setting

Datasets. We mainly conduct experiments on the DTU [12] and DeepFashion3D [49] datasets. The DTU dataset contains scenes with closed objects and multiple images. Following prior works [37], we use the widely adopted 15 scenes from DTU to evaluate our method on closed objects. The DeepFashion3D dataset is a large-scale collection of non-watertight 3D clothing models. We validate our method on 12 instances with multi-view images rendered by NeuralUDF [23]. For ground-truth visualization, we extract meshes from the point clouds of DeepFashion3D using the ball-pivoting algorithm [2]. We also test several scenes from BlendedMVS [41] and NeUDF [22] to assess the robustness of our method further.

Baselines. We compare our method with 1) the classical multi-view reconstruction method COLMAP [30], for which we extract meshes from the generated point clouds using Screened Poisson [16]; 2) neural implicit reconstruction methods based on surface rendering, including IDR [44] and UNISURF [28]; 3) neural implicit reconstruction methods based on volume rendering, i.e., VolSDF [42], NeuS [37], NeuralUDF [23], NeUDF [22], 2S-UDF [6], and UDF-Prior [48].Similar to neural implicit methods (e.g., NeuralUDF and NeUDF), our NGR is trained on a scene-by-scene basis. Therefore, we follow the standard evaluation protocol, reporting per-scene results and their average over each dataset.

Metrics. We follow existing works (e.g., NeuralUDF [23], NeUDF [22] and 2S-UDF [6]) to use the Chamfer distance (CD) for the evaluation of multi-view reconstruction quality, where CD defines the distance between two point sets $\mathcal{S}_1, \mathcal{S}_2 \in \mathbb{R}^3$, i.e.,

$$d_{CD} = \sum_{\mathbf{x} \in \mathcal{S}_1} \min_{\mathbf{y} \in \mathcal{S}_2} ||\mathbf{x} - \mathbf{y}||_2^2 + \sum_{\mathbf{y} \in \mathcal{S}_2} \min_{\mathbf{x} \in \mathcal{S}_1} ||\mathbf{x} - \mathbf{y}||_2^2. \tag{8}$$

4.2. Experimental Results

Comparison Results on Open Objects. To assess the efficacy of our method for reconstructing open objects, we conduct a comparative analysis against baseline methods using the DeepFashion3D dataset. Tab. 1 reports the Chamfer distance between the reconstructed results and the corresponding ground-truth point clouds on the DeepFashion3D dataset. Tab. 1 shows that UDF-based methods outperform the NeuS method by a large margin. This is because the NeuS method relies on SDF to represent open surfaces, which will produce either closed garments with substantial errors or double-layered artifacts. Despite UDF-based methods being naturally suited to open surfaces, our method achieves the lowest CD values on most instances in the DeepFashion3D dataset, demonstrating superior performance in open-surface reconstruction. This is also confirmed in Fig. 4.

Figure 4 shows visual comparisons on DeepFashion3D scans 117 and 448. The zoomed-in regions demonstrate that our method recovers finer garment details than other UDF-based approaches. This improvement comes from using gradient information to more accurately locate the zero-level set, rather than depending only on UDF values. Additionally, our method more flexibly captures garment holes, which are common in DeepFashion3D, because the rendering scheme relaxes strict enforcement of the UDF at the surface during early optimization. Instead of forcing an immediate zero-level set, UDF values decrease gradually. At the start, the underoptimized color field tends to pull the UDF toward a convex hull; with continued optimization, the UDF then converges toward the ground truth. If the zero-level set converges too early, it prevents the color field from further adjusting, which can cause the surface to close prematurely. Although 2S-UDF [6] introduces a sparse loss term, i.e. $\exp(-\text{coef} \cdot \text{udf})$, to lift the UDF values to mitigate this issue, it sometimes results in broken reconstructions (as shown in the bottom row of Fig. 4). More reconstruction results can be found in the supplementary materials.

Comparison Results on Closed Objects. We evaluate our

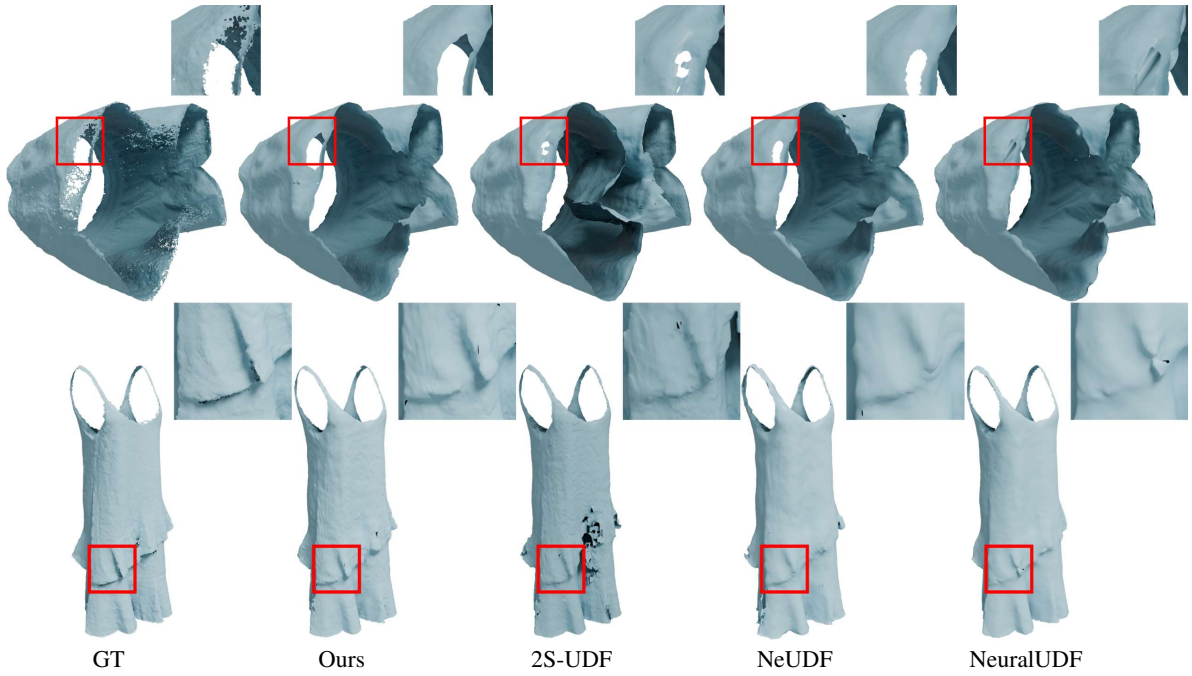


Figure 4: Qualitative comparisons on DeepFashion3D [49] dataset.

Table 1: Quantitative comparisons on Deepfashion3D [49] dataset using the metric of CD ($\times 10^{-3}$). Marker: **1st rank** and 2nd rank.

Methods	30	92	117	133	164	204	300	320	448	522	591	598	Avg.
Colmap [30]	2.95	2.91	3.58	3.06	3.23	3.26	3.09	3.11	2.95	3.16	2.97	2.95	3.10
NeuS [37]	3.18	4.82	4.78	4.99	3.73	5.71	5.89	2.21	5.89	3.60	2.44	5.13	4.36
NeuralUDF [23]	1.92	2.05	2.36	1.58	1.33	4.11	2.47	1.50	1.63	2.47	2.16	2.15	2.14
NeUDF [22]	2.10	2.20	2.04	1.88	1.63	4.35	2.04	1.71	1.61	2.49	2.35	1.98	2.20
2S-UDF [6]	1.92	1.97	2.48	1.59	1.32	2.46	2.17	1.47	2.80	2.14	1.84	1.91	2.01
UDF-Prior [48]	**1.59**	**1.73**	2.06	1.63	1.44	2.07	1.66	1.60	**1.39**	2.14	**1.50**	1.67	1.71
Ours	1.71	2.02	**1.49**	**1.43**	**1.28**	**1.86**	**1.54**	**1.45**	1.40	**1.78**	1.86	**1.61**	**1.62**

method on the DTU [12] dataset to assess its efficacy in reconstructing closed objects. The quantitative results are reported in Tab. 2. As shown in Tab. 2, our method outperforms two competing methods, including COLMAP [30] and UniSurf [26]. In addition, our method achieves quantitative results comparable to the IDR method, even though IDR relies on additional mask supervision during training, whereas our method does not. Note that the Chamfer distance primarily captures the average distance between two point sets and is therefore insensitive to fine geometric details [25]. Please refer to the qualitative results for an assessment of those subtle shape differences.

As shown in the top row of Fig. 5, visual results demonstrate that our method can effectively reconstruct finer details even though it achieves slightly inferior quantitative results compared to NeuS [37] regarding the CD metric. IDR and NeuS methods struggle to differentiate the screw-driver from the brick, whereas our method successfully reconstructs the complex object with well-defined edges, see the bottom row of Fig. 5.

Comparison Results on Open Objects with Dual-Sided Coloring. We mentioned that the unbiased property may limit the ability to reconstruct an open surface with different colors on each side in Sec. 3.2. To validate this, we synthesize data consisting of open objects with dual-sided coloring. Specifically, we render the images under albedo lighting conditions and exclude the view direction and gradient from the color network input to eliminate potential distractions. Details of the synthesized data are provided in the supplementary materials.

As illustrated in Fig. 6, the target object is a cap textured with Van Gogh's The Starry Night, and its brim has different colors on the two faces. Only our method pro-

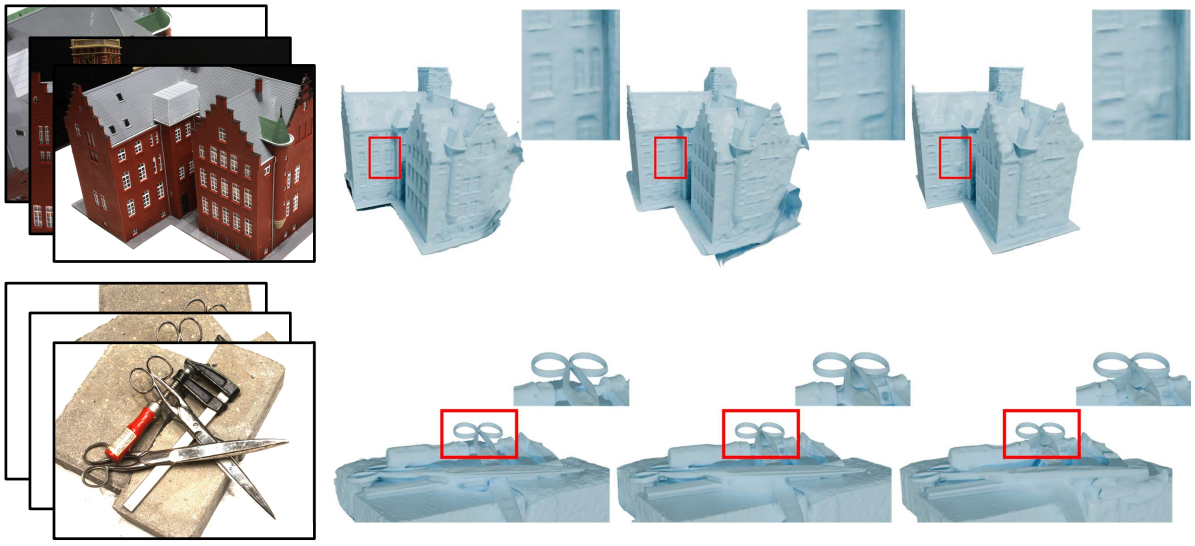


Figure 5: Qualitative comparisons on the DTU [12] dataset. We add a side view in the second example for better visualization.

Table 2: Qualitative comparisons of our gradient-based method with baselines on the DTU [12] dataset using the metric of CD ($\times 10^{-3}$). Note that IDR is trained with additional mask supervision. Marker: **1st rank** and 2nd rank.

Methods	24	37	40	55	63	65	69	83	97	105	106	110	114	118	122	Avg.
COLMAP[30]	**0.81**	2.05	0.73	1.22	1.79	1.58	1.02	3.05	1.40	2.05	1.00	1.34	0.49	0.78	1.17	1.36
IDR [44]	1.63	1.87	**0.63**	0.48	**1.04**	0.79	0.77	1.33	1.16	0.76	0.67	**0.90**	0.42	0.51	0.53	0.90
NeuS[37]	1.37	1.21	0.73	**0.40**	1.20	0.70	0.72	**1.01**	1.16	0.82	0.66	1.69	0.39	**0.49**	0.51	0.87
NeUDF[22]	2.27	2.79	2.11	0.73	2.52	0.97	1.63	1.53	1.63	**0.68**	0.93	3.44	0.59	0.82	0.99	1.58
NeuralUDF[23]	1.51	1.55	1.16	0.49	1.30	0.69	0.96	1.82	1.18	1.02	0.65	1.82	0.47	0.65	0.84	1.07
2S-UDF [6]	2.52	0.89	3.66	0.41	1.07	**0.68**	0.88	3.09	1.15	0.70	0.74	3.41	0.41	0.61	0.51	1.53
UDF-Prior [48]	0.94	**1.17**	0.84	0.47	1.25	**0.68**	**0.64**	1.57	**1.02**	0.96	**0.60**	1.33	**0.35**	**0.49**	**0.50**	**0.85**
Ours	0.97	**0.85**	0.77	0.41	1.07	0.80	0.82	1.48	1.03	1.07	0.61	1.14	0.42	0.62	0.66	**0.85**

Figure 6: Reconstruction results of a cap with a dual-colored brim.

duces a reasonable result. NeuralUDF [23] completely fails to obtain a meaningful reconstruction because the weight always reaches its maximum value at the zero-level set along the ray, regardless of where the ray is cast. As a result, the rendered color remains the same when the ray is cast from different sides of the surface. NeUDF [22] successfully captures the coarse shape of the brim but ends up fragmented due to its unbiased property. Compared to NeuralUDF, NeUDF assigns $+\infty$ to the density of the zero-level set, which may mitigate the issue to some extent.

Comparison Results on Running Time. We also evaluate the running time of our method compared with other implicit neural volume rendering approaches. All experiments are conducted on a single NVIDIA RTX 3090 GPU. As summarized in Tab. 3, our proposed NGR (with gradient-aggregation enabled) consistently requires less training time per iteration than UDF-Prior [48], NeUDF [22], and NeuralUDF [23], while remaining competitive with 2S-UDF [6] and NeuS [37].

4.3. Ablation Study

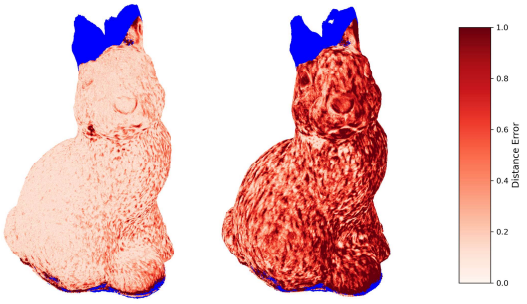
To validate our proposed designs in Sec. 3, we conduct an ablation study: (a) use absolute instead of softplus to ensure the non-negativity of UDF; (b) use the naive rendering scheme instead of the scheme based on gradient aggregation; (c) our full model. The visualization is presented in Fig. 7. The result of (a) is completely null because initializing the sphere SDF with an absolute function prevents the

Table 3: Running time comparisons. All experiments are conducted on a single NVIDIA RTX 3090 GPU.

Methods	Training time (Hours)
Ours - Naive gradient rendering	9.68
Ours - Gradient aggregation rendering	10.32
UDF-Prior [48]	10.80
NeUDF [22]	10.83
2S-UDF [6]	9.20
NeuralUDF [23]	11.25
NeuS [37]	**8.07**

(a) (b) (c)

Figure 7: Ablation study. (a) Use absolute instead of softplus to ensure the non-negativity of UDF. At the beginning of (a), the extracted mesh is closed. During optimization, the mesh collapses to null, and the rendered images turn pure black. (b) Use a naive rendering scheme instead of the scheme based on the aggregation of gradients. (c) Full model.

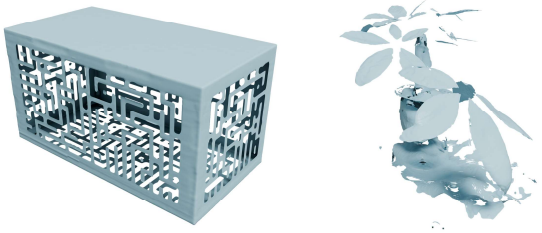


(a) CD: 0.41 (b) CD: 0.86

Figure 8: Visualization of reconstruction error with or without w^2. Redder regions indicate larger geometric errors, while blue areas represent regions outside the calculation range. (a) Without the coefficient loss term, the reconstructed mesh appears shrunken compared to the ground truth, resulting in a larger Chamfer distance while preserving plausible geometry. (b) With the coefficient loss term, the Chamfer distance ($\times 10^{-3}$) is significantly reduced.

optimization from altering the topology. Compared to rendering schemes that rely on uncertain sampling, our gradient aggregation-based approach enables the reconstruction of fine-grained surfaces. The quantitative comparisons are also provided in Tab. 4.

We further investigate the effect of the coefficient loss term w^2. As illustrated in Fig. 8, a large w value compromises the unbiased property, leading to significant reconstruction errors. We achieve a substantial improvement in reconstruction quality through the coefficient loss. A quantitative comparison is also provided in Tab. 5.



(a) rack (b) plant

Figure 9: Results on the dataset released by NeUDF [22]. Specifically, (a) is an extremely complex object, reconstructed from synthetic images, and (b) is a plant reconstructed from images captured in the wild.

Additional Results. We further validate our method on two scenes from NeUDF [22], as shown in Fig.9. The rack in Fig.9 (a) is a highly intricate synthetic open surface that our method successfully reconstructs with remarkable accuracy. Fig. 9 (b) shows a plant captured in the wild, further demonstrating the robustness of our approach. We also evaluate two scenes from BlendedMVS [41] to test the reconstruction of closed objects in diverse scenarios. As shown in Fig. 10, our method consistently produces high-fidelity reconstructions, accurately capturing fine details and complex geometries. The visualization of gradient fields in Fig. 11 provides an additional perspective on the learning stability of our method. More results can be found in our supplementary materials.

5. Conclusions

We introduce a universal gradient-based rendering scheme applicable to both UDF and SDF. Our key insight is that while the gradient direction undergoes sharp changes at the zero-level set of the distance field, the distance function itself remains highly smooth, making it difficult to capture fine details. By leveraging gradient-based rendering, we can effectively capture intricate geometrical details. Experiments demonstrate that our rendering scheme outperforms value-based volume rendering in extracting fine-grained details from multi-view images. Furthermore, we argue that the widely adopted unbiased property is not appropriate for

Table 4: Ablation study on the DeepFashion3D dataset to evaluate the use of softplus activation function and gradient aggregation-based rendering. Marker: **1st rank**.

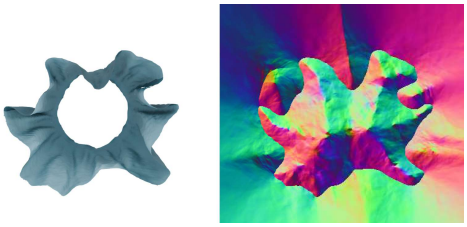
Methods	30	92	117	133	164	204	300	320	448	522	591	598	Avg.
wo/ softplus	N/A	N/A	N/A	21.12	3.81	N/A	12.62	N/A	34.19	N/A	37.00	N/A	21.75
wo/ gradient aggregation	9.14	N/A	N/A	17.57	10.93	10.94	7.93	6.47	21.56	9.06	22.21	10.98	12.68
Ours	**1.71**	**2.02**	**1.49**	**1.43**	**1.28**	**1.86**	**1.54**	**1.45**	**1.40**	**1.78**	**1.86**	**1.61**	**1.62**

Table 5: Ablation study on the DTU dataset to evaluate the use of the w^2 loss term. Marker: **1st rank**.

Methods	24	37	40	55	63	65	69	83	97	105	106	110	114	118	122	Avg.
wo/ w^2 loss	1.94	1.41	1.91	0.86	2.19	1.28	1.31	1.60	1.50	**0.93**	1.07	1.76	1.12	1.49	5.61	1.73
Ours	**0.97**	**0.85**	**0.77**	**0.41**	**1.07**	**0.80**	**0.82**	**1.48**	**1.03**	1.07	**0.61**	**1.14**	**0.42**	**0.62**	**0.66**	**0.85**

(a) dog (b) scupture

Figure 10: Results on the samples of BlendedMVS [41].



(a) (b)

Figure 11: Visualization of the gradient fields. The gradient direction in a cross-sectional plane is encoded as RGB color in (b), and the corresponding mesh cross-section is shown in (a).

reconstructing open surfaces. Extensive experiments further validate the effectiveness and robustness of our method.

Limitations and Future Work. In this paper, we enforce a prior that the surface normal flips by approximately 180° across the surface, which helps guide geometric optimization. However, the true distance function is not differentiable everywhere, while the MLP used to fit it is infinitely differentiable. This mismatch prevents the MLP from exactly matching the ground truth. Besides, because we impose strong constraints to approximate the full distance field rather than only learning the zero-level set position, SDF optimization can sometimes cause the reconstructed zero-level set to expand slightly. Future work will also extend our evaluation framework to include the normal consistency, Hausdorff distance, and F-score for a more robust assessment of geometric accuracy.

Acknowledgements

The work was supported in part by the National Natural Science Foundation of China (No. 62502209 and No. 62401280) and the Fundamental Research Funds for the Central Universities (No. 30925010538). We thank the anonymous reviewers for their constructive comments and suggestions.

References

[1] M. Atzmon and Y. Lipman. Sal: Sign agnostic learning of shapes from raw data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2565–2574, 2020. 2, 4, 5

[2] F. Bernardini, J. Mittleman, H. Rushmeier, C. Silva, and G. Taubin. The ball-pivoting algorithm for surface reconstruction. *IEEE transactions on visualization and computer graphics*, 5(4):349–359, 1999. 2, 5

[3] D. Chen, H. Li, W. Ye, Y. Wang, W. Xie, S. Zhai, N. Wang, H. Liu, H. Bao, and G. Zhang. Pgsr: Planar-based gaussian splatting for efficient and high-fidelity surface reconstruction. 2024. 1

[4] W. Chen, C. Lin, W. Li, and B. Yang. 3psdf: Three-pole signed distance function for learning surfaces with arbitrary topologies. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, June 2022. 2

[5] J. Chibane, A. Mir, and G. Pons-Moll. Neural unsigned distance fields for implicit function learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, December 2020. 2

[6] J. Deng, F. Hou, X. Chen, W. Wang, and Y. He. 2S-UDF: A Novel Two-stage UDF Learning Method for Robust Non-watertight Model Reconstruction from Multi-view Images.

In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5084–5093, 2024. 1, 3, 4, 5, 6, 7, 8

[7] Y. Furukawa and J. Ponce. Accurate, dense, and robust multiview stereopsis. *IEEE transactions on pattern analysis and machine intelligence*, 32(8):1362–1376, 2009. 1

[8] A. Gropp, L. Yariv, N. Haim, M. Atzmon, and Y. Lipman. Implicit geometric regularization for learning shapes. In *Proceedings of Machine Learning and Systems 2020*, pages 3569–3579, 2020. 5

[9] B. Guillard, F. Stella, and P. Fua. Meshudf: Fast and differentiable meshing of unsigned distance field networks. In *European Conference on Computer Vision*, 2022. 5

[10] H. Guo, S. Peng, H. Lin, Q. Wang, G. Zhang, H. Bao, and X. Zhou. Neural 3d scene reconstruction with the manhattan-world assumption. In *CVPR*, 2022. 3

[11] M. Jäger and B. Jutzi. 3d density-gradient based edge detection on neural radiance fields (nerfs) for geometric reconstruction. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 48:71–78, 2023. 2

[12] R. Jensen, A. Dahl, G. Vogiatzis, E. Tola, and H. Aanæs. Large scale multi-view stereopsis evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 406–413, 2014. 2, 5, 6, 7

[13] M. Ji, J. Gall, H. Zheng, Y. Liu, and L. Fang. Surfacenet: An end-to-end 3d neural network for multiview stereopsis. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2307–2315, 2017. 2

[14] M. Ji, J. Zhang, Q. Dai, and L. Fang. Surfacenet+: An end-to-end 3d neural network for very sparse multi-view stereopsis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(11):4078–4093, 2020. 2

[15] A. Kar, C. Häne, and J. Malik. Learning a multi-view stereo machine. *Advances in neural information processing systems*, 30, 2017. 2

[16] M. Kazhdan and H. Hoppe. Screened poisson surface reconstruction. *ACM Transactions on Graphics (ToG)*, 32(3):1–13, 2013. 2, 5

[17] P. Kellnhofer, L. Jebe, A. Jones, R. Spicer, K. Pulli, and G. Wetzstein. Neural lumigraph rendering. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 4287–4297. Computer Vision Foundation / IEEE, 2021. 3

[18] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014. 5

[19] H. Le, F. Stella, B. Guillard, and P. Fua. Gradient distance function, 2024. 2

[20] S. Liu, S. Saito, W. Chen, and H. Li. Learning to infer implicit surfaces without 3d supervision. In H. M. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. B. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 8293–8304, 2019. 3

[21] S. Liu, Y. Zhang, S. Peng, B. Shi, M. Pollefeys, and Z. Cui. DIST: rendering deep implicit signed distance function with differentiable sphere tracing. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 2016–2025. Computer Vision Foundation / IEEE, 2020. 3

[22] Y.-T. Liu, L. Wang, J. Yang, W. Chen, X. Meng, B. Yang, and L. Gao. Neudf: Learning neural unsigned distance fields with volume rendering. In *Computer Vision and Pattern Recognition (CVPR)*, 2023. 1, 3, 4, 5, 6, 7, 8

[23] X. Long, C. Lin, L. Liu, Y. Liu, P. Wang, C. Theobalt, T. Komura, and W. Wang. Neuraludf: Learning unsigned distance fields for multi-view reconstruction of surfaces with arbitrary topologies. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20834–20843, 2023. 1, 3, 4, 5, 6, 7, 8

[24] W. E. Lorensen and H. E. Cline. Marching cubes: A high resolution 3d surface construction algorithm. In *Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '87, page 163–169, New York, NY, USA, 1987. Association for Computing Machinery. 5

[25] L. Mescheder, M. Oechsle, M. Niemeyer, S. Nowozin, and A. Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2, 6

[26] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *European conference on computer vision*, pages 405–421. Springer, 2020. 1, 3, 6

[27] M. Niemeyer, L. M. Mescheder, M. Oechsle, and A. Geiger. Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020*, pages 3501–3512. Computer Vision Foundation / IEEE, 2020. 3

[28] M. Oechsle, S. Peng, and A. Geiger. Unisurf: Unifying neural implicit surfaces and radiance fields for multi-view reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5589–5599, 2021. 5

[29] A. G. Puigjaner, E. M. Rella, E. Sandström, A. Chhatkuli, and L. V. Gool. Vf-nerf: Learning neural vector fields for indoor scene reconstruction, 2024. 2

[30] J. L. Schonberger and J.-M. Frahm. Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4104–4113, 2016. 1, 2, 5, 6, 7

[31] J. L. Schönberger, E. Zheng, J.-M. Frahm, and M. Pollefeys. Pixelwise view selection for unstructured multi-view stereo. In *European Conference on Computer Vision*, pages 501–518. Springer, 2016. 1

[32] V. Sitzmann, M. Zollhöfer, and G. Wetzstein. Scene representation networks: Continuous 3d-structure-aware neural scene representations. In H. M. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. B. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*

32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada, pages 1119–1130, 2019. 3

[33] J. Sun, Y. Xie, L. Chen, X. Zhou, and H. Bao. Neuralrecon: Real-time coherent 3d reconstruction from monocular video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15598–15607, 2021. 2

[34] T. Takikawa, J. Litalien, K. Yin, K. Kreis, C. T. Loop, D. Nowrouzezahrai, A. Jacobson, M. McGuire, and S. Fidler. Neural geometric level of detail: Real-time rendering with implicit 3d shapes. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2021, virtual, June 19-25, 2021*, pages 11358–11367. Computer Vision Foundation / IEEE, 2021. 3

[35] J. Wang, P. Wang, X. Long, C. Theobalt, T. Komura, L. Liu, and W. Wang. Neuris: Neural reconstruction of indoor scenes using normal priors. In *European Conference on Computer Vision*, pages 139–155. Springer, 2022. 3

[36] L. Wang, J. Yang, W.-K. Chen, X.-X. Meng, B. Yang, J.-T. Li, and L. Gao. Hsdf: Hybrid sign and distance field for modeling surfaces with arbitrary topologies. In *Neural Information Processing Systems (NeurIPS)*, 2022. 2

[37] P. Wang, L. Liu, Y. Liu, C. Theobalt, T. Komura, and W. Wang. Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. *Advances in Neural Information Processing Systems*, 34, 2021. 1, 2, 3, 4, 5, 6, 7, 8

[38] Y. Wang, I. Skorokhodov, and P. Wonka. HF-neus: Improved surface reconstruction using high-frequency details. In A. H. Oh, A. Agarwal, D. Belgrave, and K. Cho, editors, *Advances in Neural Information Processing Systems*, 2022. 3

[39] Y. Wang, I. Skorokhodov, and P. Wonka. Pet-neus: Positional encoding triplanes for neural surfaces. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023. 3

[40] X. Yang, G. Lin, Z. Chen, and L. Zhou. Neural vector fields: Implicit representation by explicit learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16727–16738, June 2023. 2

[41] Y. Yao, Z. Luo, S. Li, J. Zhang, Y. Ren, L. Zhou, T. Fang, and L. Quan. Blendedmvs: A large-scale dataset for generalized multi-view stereo networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1790–1799, 2020. 2, 5, 8, 9

[42] L. Yariv, J. Gu, Y. Kasten, and Y. Lipman. Volume rendering of neural implicit surfaces. *Advances in Neural Information Processing Systems*, 34, 2021. 1, 3, 5

[43] L. Yariv, Y. Kasten, D. Moran, M. Galun, M. Atzmon, R. Basri, and Y. Lipman. Multiview neural surface reconstruction by disentangling geometry and appearance. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. 3

[44] L. Yariv, Y. Kasten, D. Moran, M. Galun, M. Atzmon, B. Ronen, and Y. Lipman. Multiview neural surface reconstruction by disentangling geometry and appearance. *Advances in Neural Information Processing Systems*, 33:2492–2502, 2020. 5, 7

[45] Z. Yu, S. Peng, M. Niemeyer, T. Sattler, and A. Geiger. Monosdf: Exploring monocular geometric cues for neural implicit surface reconstruction. *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. 3

[46] Z. Yu, T. Sattler, and A. Geiger. Gaussian opacity fields: Efficient adaptive surface reconstruction in unbounded scenes. *ACM Transactions on Graphics*, 2024. 1

[47] K. Zhang, G. Riegler, N. Snavely, and V. Koltun. Nerf++: Analyzing and improving neural radiance fields. *arXiv preprint arXiv:2010.07492*, 2020. 5

[48] W. Zhang, K. Shi, Y.-S. Liu, and Z. Han. Learning unsigned distance functions from multi-view images with volume rendering priors. *European Conference on Computer Vision*, 2024. 5, 6, 7, 8

[49] H. Zhu, Y. Cao, H. Jin, W. Chen, D. Du, Z. Wang, S. Cui, and X. Han. Deep fashion3d: A dataset and benchmark for 3d garment reconstruction from single images. In *European Conference on Computer Vision*, pages 512–530. Springer, 2020. 1, 2, 5, 6