Recovering Fine Details for Neural Implicit Surface Reconstruction

Decai Chen[1], Peng Zhang[1,2], Ingo Feldmann[1], Oliver Schreer[1], and Peter Eisert[1,3]

[1]Fraunhofer HHI [2]TU Berlin [3]HU Berlin

Abstract

Recent works on implicit neural representations have made significant strides. Learning implicit neural surfaces using volume rendering has gained popularity in multi-view reconstruction without 3D supervision. However, accurately recovering fine details is still challenging, due to the underlying ambiguity of geometry and appearance representation. In this paper, we present D-NeuS, a volume rendering-base neural implicit surface reconstruction method capable to recover fine geometry details, which extends NeuS by two additional loss functions targeting enhanced reconstruction quality. First, we encourage the rendered surface points from alpha compositing to have zero signed distance values, alleviating the geometry bias arising from transforming SDF to density for volume rendering. Second, we impose multi-view feature consistency on the surface points, derived by interpolating SDF zero-crossings from sampled points along rays. Extensive quantitative and qualitative results demonstrate that our method reconstructs high-accuracy surfaces with details, and outperforms the state of the art. [1]

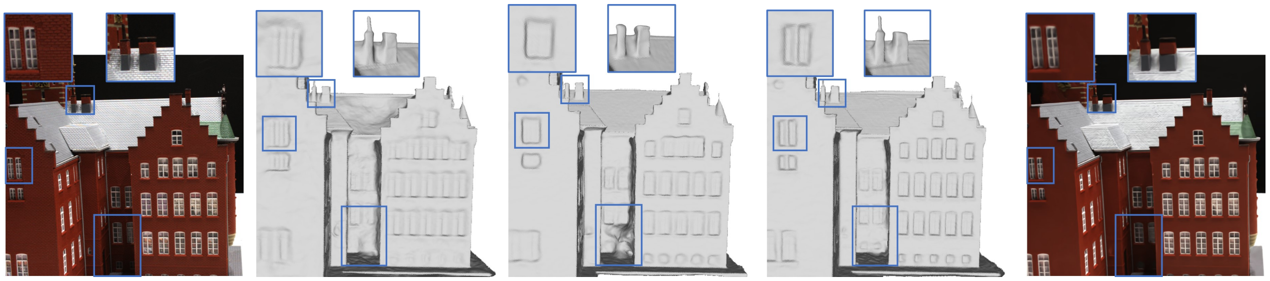
1. Introduction

3D reconstruction from calibrated multi-view images is a long-standing challenge in computer vision and has been explored for decades. Classical approaches such as traditional [9, 24, 27] and learning-based multi-view stereo (MVS) [29, 30, 36, 25, 5] produce depth maps via matching photometric or feature correspondence of pixels or patches across a set of images. However, the classical MVS pipeline involves several steps including depth map prediction, fusion into global space and surface extraction, where errors and artifacts inevitably accumulate. Inspired by the seminal work NeRF [18], neural implicit surface reconstruction approaches [33, 21, 32, 26] have recently emerged as a powerful tool for the 3D reconstruction of geometry and appearance, leveraging coordinate-based Multi-Layer Perceptron (MLP) neural networks to represent the underlying surface. These approaches apply differentiable rendering to optimize jointly the shape and the appearance field by minimizing the gap between rendered images and the input ground truth. While rendering plausible novel views, these methods still struggle to recover high-fidelity geometry details. In this paper, we propose a Details recovering Neural implicit Surface reconstruction method named D-NeuS, with two constraints to guide the SDF field-based volume rendering and thus improve the reconstruction quality. As shown in Fig. 1, our method is able to reconstruct more accurate geometry details than the state of the art [26, 7].

To get rid of geometric errors of the standard volume rendering approaches, NeuS [26] applies a weight function which is occlusion-aware and unbiased in the first-order approximation of SDF. However, we argue that the weight function under non-linearly distributed SDF field causes bias between the geometric surface point (*i.e.* SDF zero crossing) and rendered surface point from alpha compositing. To this end, we propose a novel scheme to mitigate this bias. Specifically, we generate additional distance maps during the volume rendering, back-project the distance into 3D points, and penalize their absolute SDF values predicted by the geometry MLP network. By doing this, we encourage the consistency between volume rendering and underlying surface.

Although current neural implicit surface networks are able to render plausible images on novel views, encoding high-frequency textures in MLP is still challenging [7]. To alleviate this issue, NeuralWarp [7] introduces patch-based photometric consistency, which is computed on all sampled points along rays before merging by alpha compositing. Inspired by MVSDF [37], we instead take the advantage of robust representation performance of convolutional neural networks (CNNs), by employing feature-based consistency on only the surface point along a ray. Instead of finding surface points via ray tracing as in [37], which requires recursively querying the geometry network and thus is computationally expensive, we simply look for the first zero crossing

[1] Code: `https://github.com/fraunhoferhhi/D-NeuS`.



Reference Image NeuS [26] NeuralWarp [7] D-NeuS (Ours) Our Rendered Image

Figure 1: A surface reconstruction example from DTU dataset [12]. Compared to the state-of-the-art methods, D-NeuS recovers higher-fidelity surface details. Besides, we achieve photorealistic view synthesis.

point from the SDF values of the sampled points via locally differentiable linear interpolation. This involves no extra computation because the SDF values of the sampled points are computed already for volume rendering.

To summarize, the main contributions of our work are as follows:

- We provide theoretical analysis of the geometry bias resulting from the unregularized SDF field in volume rendering-based neural implicit surface network, and propose a novel constraint to regularize this bias.
- We apply multi-view feature consistency on surface points from linearly interpolated SDF zero-crossing, for fine local geometric details.
- We evaluate qualitatively and quantitatively the proposed method on DTU [12], BlendedMVS [31] datasets, and show that it outperforms the state of the art, with high-accuracy surface reconstruction, especially on complex scenes.

2. Related Works

Multi-View Stereo. MVS is a classical method for recovering a dense scene representation from overlapping images. Traditional MVS approaches typically leverage pair-wise matching cost of RGB image patches by Normalized Cross-Correlation (NCC), Sum of Squared Distances (SSD) or Sum of Absolute Differences (SAD). In recent years, PatchMatch-based [4] MVS [9, 24, 27] are dominating traditional methods due to highly parallelism and robust performance. Recently, deep learning-based MVS shows superior performance. MVSNet [29] builds cost volumes by warping feature maps from across neighboring views and applies 3D CNNs to regularize the cost volumes. To mitigate the memory consumption of 3D CNNs, R-MVSNet [30] regularizes 2D cost maps sequentially using a gated recurrent network, while other methods [6, 28, 11, 36] integrate coarse-to-fine multi-stage strategies to progressively refine the 3D cost volumes. PatchmatchNet [25] proposes an iterative multiscale PatchMatch strategy in a differentiable MVS architecture. More recently, TransMVSNet [8] introduces Transformer to aggregate long-range context information within and across images. However, matching pixels in low-texture or non-Lambertian areas remains challenging, and errors inevitably accumulate from the following point cloud fusion and surface reconstruction. In this work, we leverage multi-view feature consistency, universally used in learning-based MVS, to constrain the volume rendering for more accurate surface reconstruction.

Implicit Surface Representation and Reconstruction. The success of NeRF [18] in representing a scene by 5D radiance field has recently drawn considerable attention from the community of both computer vision and computer graphics. Implicit neural representation leverage physics-based traditional volume rendering in a differentiable way, enabling photorealistic novel view synthesis without 3D supervision. While NeRF-like approaches [18, 38, 3] achieve impressive rendering quality, their underlying geometry is generally noisy and less favorable. The reason for that is two-fold. First, the geometry and appearance fields are entangled in differentiable rendering, if only learned from 2D image reconstruction consistency. Secondly, representing the geometry field by density is difficult to be constrained and regularized.

To alleviate the above issue, current implicit surface reconstruction methods employ surface indicator functions, mapping continuous spatial coordinates to occupancy [17, 23, 20, 21] and SDF [22, 33, 26, 32], where Marching cubes [16] is commonly applied to extract the implicit surface at any resolution. IDR [33] renders the color of a ray only on the object surface point, and applies differentiable ray tracing to back-propagate the gradients to a local region near the intersection. MVSDF [37] extends this framework with supervision from depth maps and feature consistency,

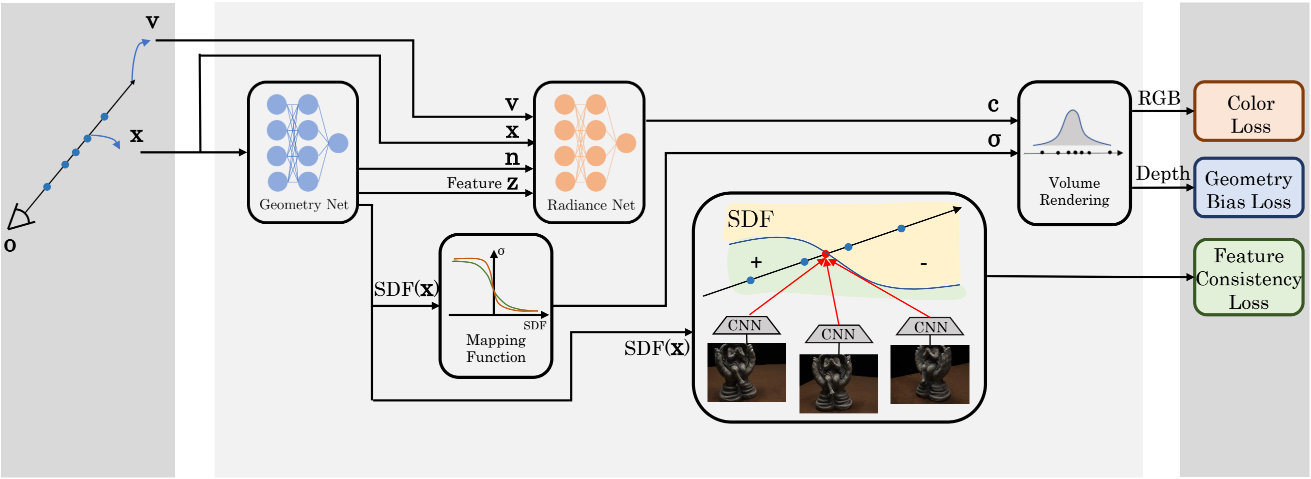


Figure 2: Overview of the proposed method. We build on a neural implicit surface framework [26] and introduce two additional constraints: a geometry bias loss which regularizes SDF-based volume rendering (Sec. 3.2), and multi-view feature consistency loss(Sec. 3.3), to significantly improve the reconstruction quality.

while RegSDF [35] introduces supervision by point clouds and additional geometric regularization to reconstruct unbounded or complex scenes. However, surface rendering-based methods struggle with reconstructing complex objects with sudden depth changes, and thus they usually require additional supervision, such as object masks, depth maps or point clouds.

To combine advantages of surface-based and volume-based rendering techniques, UNISURF [21] proposes a coarse-to-fine strategy for point sampling around the surface represented by an occupancy field. VolSDF [32] trains an implicit surface model using an efficient sampling algorithm, guided by error bound of opacity approximation. NeuralWarp [7] extends VolSDF by adding patch warping from source images to the reference image using homography, and improve the surface geometry by photometric constraint. Since patch warping requires reliable surface normals, NeuralWarp only serves as a post-processing method to fine-tune and optimize a pretrained surface model. Similar to VolSDF, NeuS [26] designs an occlusion-aware transformation function mapping signed distances to weights for volume rending, with a learnable parameter to control the slope of the logistic density function. However, this mapping function is only unbiased in a regularized SDF field which is linearly distributed, so we propose a novel constraint to compensate for the geometry bias. We build our framework on NeuS [26], but we believe our proposed method could be adapted to any volume rendering-based neural implicit surface reconstruction work.

3. Method

Given a set of images with known intrinsic and extrinsic camera parameters, the goal of our method is to reconstruct high-fidelity surface represented by implicit neural networks. Following NeuS [26], we encode surfaces as signed distance fields. The overview of our framework is illustrated in Fig. 2. We will explain our method in four parts: 1) First, we show how SDF-based neural implicit surfaces are learned via volume rendering (Section 3.1). 2) Then, we analyze the geometry bias of volume rendering in unregularized SDF field and propose a novel constraint to mitigate this error (Section 3.2). 3) We demonstrate how to apply feature consistency on surface points from linearly interpolated SDF zero-crossings (Section 3.3). 4) Finally, we present all losses used for optimization (Section 3.4).

3.1. Volume Rendering-based Implicit Surface Reconstruction

In this section, we adopt the notations and review the basics of NeuS [26], an SDF-based neural surface reconstruction method using volume rendering, so that we can better demonstrate our analysis in Section 3.2. In contrast to the density-based geometry representation, the surface S is represented by an implicit field where the signed distance equals zero:

$$\mathcal{S} = \mathbf{x} \in \mathbb{R}^3 \mid f(\mathbf{x}) = 0. \tag{1}$$

where f is a function $f : \mathbb{R}^3 \to \mathbb{R}$ mapping a 3D point $\mathbf{x} \in \mathbb{R}^3$ to its SDF field. In addition to geometry, the view-dependent appearance field is defined by a function $g : \mathbb{R}^3 \times \mathbb{S}^2 \times \mathbb{S}^2 \times \mathbb{R}^m \to \mathbb{R}^3$ that derive the color $\mathbf{c} \in \mathbb{R}^3$

from a spatial point $\mathbf{x}$, its view direction $\mathbf{v} \in \mathbb{S}^2$, its normal $\mathbf{n} \in \mathbb{S}^2$ calculated from automatic differentiation of the SDF (*i.e.*, $\nabla f(\mathbf{x})$), and a feature vector $\mathbf{z} \in \mathbb{R}^m$ from the geometry network f, as shown in Fig. 2. Multi-layer Perception (MLP) neural networks are used to approximate both f and g.

A 3D point on a ray from a camera centered at $\mathbf{o}$ with unit view direction $\mathbf{v}$ can be denoted as:

$$\mathbf{x}(t) = \mathbf{o} + t\mathbf{v} \mid t \geq 0, \tag{2}$$

where t is the distance between $\mathbf{x}$ and $\mathbf{o}$. Colors along a ray are accumulated by volume rendering

$$C(\mathbf{o}, \mathbf{v}) = \int_0^{+\infty} \omega(t) g(\mathbf{x}(t), \mathbf{v}, \mathbf{n}, \mathbf{z}) dt, \tag{3}$$

where $C(\mathbf{o}, \mathbf{v})$ is the rendered color of the pixel associated with the ray, $g(\mathbf{x}(t), \mathbf{v}, \mathbf{n}, \mathbf{z})$ is the color of a point $\mathbf{x}$ along the view direction $\mathbf{v}$, $\omega(t)$ is the weight for volume rendering at the point:

$$\omega(t) = exp\left(-\int_0^t \sigma(u) du\right) \sigma(t), \tag{4}$$

where $\sigma(t)$ is the density of the point $\mathbf{x}$ used in standard volume rendering. After rendering a set of rays, the rendered colors are compared with the input images for network supervision.

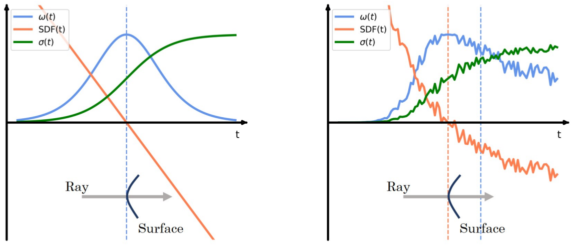
3.2. Constraint on Geometric Bias

Assuming the signed distance field is a linear function near a surface point, which is the first-order approximation of SDF, NeuS [26] proposed an unbiased opaque density function from the SDF f of the scene:

$$\sigma(t) = \max\left(\frac{-\frac{d\Phi_s}{dt}(f(\mathbf{x}(t)))}{\Phi_s(f(\mathbf{x}(t)))}, 0\right), \tag{5}$$

where $\Phi_s(x) = (1 + e^{-sx})^{-1}$ is a sigmoid function and s^{-1} is the trainable standard deviation which approaches 0 as training converges.

Fig. 3 shows how density and weight functions behave under different SDF distributions, in the simple case of a single plane intersection. Assuming the local surface as a plane, the ideal SDF value of a sampling point near the surface is linear along the camera ray [26], i.e., $f(\mathbf{x}(t)) = -|cos(\theta)| \cdot (t - t^*)$, where t^* denotes the distance from camera center to the SDF zero-crossing, and θ is a constant angle of emergence for the local plane. Based on this assumption, NeuS derives the unbiased weight distribution using Eqn. 4, as demonstrated in Fig. 3a. In this case, the point corresponding to the weighted average in volume rendering shares the same position where the SDF value is 0. In other words, the rendered color is consistent with the underlying geometry, so the supervision from input images can optimize the surface geometry precisely.



(a) Ideal SDF (b) Unregularized SDF

Figure 3: Illustration of the density and weight functions under different assumptions of SDF distribution. Non-linear SDF values cause bias between geometric surface (orange dashed line) and volume rendered surface point (blue dashed line).

However, the ideal SDF distribution is not guaranteed by the geometry MLP network, which takes a 3D point $\mathbf{x}$ and outputs its signed distance to the nearest surface. Although the weights of the geometry network are initialized to produce an approximate SDF of a unit sphere [2], the volume rendering-based image supervision itself imposes no explicit regularization on the underlying SDF field. Fig. 3b illustrates an example of unregularized SDF distribution along a camera ray, causing bias between the volume rendering integral and the SDF implicit surface. As a result, the inconsistency between color radiance field and geometry SDF field leads to less desirable surface reconstruction.

At this point, we propose a novel strategy to regularize the SDF field for volume rendering by constraining the mentioned geometry bias. Recalling a 3D point along a ray in Eqn. 2, we can render the distance $t_{rendered}$ between the camera center and the average point for volume rendering via discretizing the volume integration:

$$t_{rendered} = \sum_i^n \frac{\omega_i t_i}{\sum_i^n \omega_i}, \tag{6}$$

where n is the number of sampling points along a ray, ω_i represents the discrete counterpart of the weight in Eqn. 4, and t_i is the distance from a sampling point to the camera center. Then the volume-rendered surface point $\mathbf{x}_{renderd}$ can be formed by back-projection:

$$\mathbf{x}_{rendered} = \mathbf{o} + t_{rendered}\mathbf{v}. \tag{7}$$

Finally, we build a geometry bias loss:

$$\mathcal{L}_{bias} = \frac{1}{|\mathbb{S}|} \sum_{\mathbf{x}_{rendered} \in \mathbb{S}} |f(\mathbf{x}_{rendered})|, \tag{8}$$

where f is the geometry network outputting SDF values, $\mathbb{S}$ is the subset of $\mathbf{x}_{rendered}$ where ray-surface intersection has been found (see Sec. 3.3 for implementation details). By penalizing the absolute value of SDF of the rendered surface points, we encourage the geometry consistency between the implicit SDF field and the radiance field for volume rendering. Intuitively, this constraint regularizes the SDF distribution for unbiased volume rendering, and thus leads to more accurate surface reconstruction. It is also worth noting, that Eikonal loss [10] widely used in neural implicit surface reconstruction regularizes the gradient field of SDF by constraining the gradient norm. Both Eikonal loss and our geometry bias loss support each other, enhancing the reconstruction quality.

3.3. Multi-view Feature Consistency

Guiding geometry reconstruction with multi-view photometric or feature consistency is popular in MVS [24, 27, 29, 36] and recent neural surface reconstructions [37, 7]. Typically, photo-consistency approaches compare the photometric distance across RGB image patches requiring surface normals to compute homography, while feature consistency methods match only a single pixel between the feature maps. Extensive results, *e.g.* benchmarks on Tanks and Temples[1, 15], demonstrate that the deep feature representation shows better performance than the photometric counterpart. Therefore, we apply feature consistency to impose multi-view geometric constraint on the reconstructed object surface.

One key step for applying the multi-view constraint on neural implicit surfaces is to derive the surface point in a differentiable way. In surface rendering-based neural reconstruction [33, 37], differentiable ray tracing is commonly used to find the intersection point between a camera ray and object surface. However, for optimizing volume rendering-based surface reconstruction, applying ray tracing to find the surface point causes extra calculation, as it is not required for color rendering. Instead, NeuralWarp [7] approximates the surface point using alpha composition, *i.e.*, calculating the patch transformation on every sampling point along a ray and merging the results by volume weighted average. However, the volume-rendered surface point can be biased against the real surface, as discussed in Sec. 3.2. To this end, we take the advantage of the SDF values of the sampling points, which are computed already for volume rendering, to directly extract the zero-crossing point using linear interpolation.

Recalling Eqn. 2, we denote a sampled 3D point along a ray as $\mathbf{x}(t_i)$ where $i = 1, ..., N$ is the index and N is the number of sampled points from hierarchical sampling [26]. We search for the first point $\mathbf{x}(t_s)$, satisfying that the SDF value of this point is positive while that of the next sampling point is negative. Specifically, we can define s as:

$$s = \arg\min_i \{t_i \mid f(\mathbf{x}(t_i)) > 0 \; and \; f(\mathbf{x}(t_{i+1})) < 0\}, \quad (9)$$

where f is the geometry network outputting SDF values, and $\mathbf{x}(t_i)$ is the first sampling point right before the object surface. We only consider the first ray-surface intersection because the others are occluded. If none of the sampling points fulfills such requirement, we skip constraints on both the feature consistency and the geometric bias for this ray. Since the hierarchical sampling strategy puts high importance on sampling near the surface point, the distance between $\mathbf{x}(t_i)$ and $\mathbf{x}(t_{i+1})$ is supposed to be small. Therefore, we can approximate the surface point where SDF is zero using differentiable linear interpolation:

$$\hat{\mathbf{x}} = \left\{ \mathbf{x}(\hat{t}) \mid \hat{t} = \frac{f(\mathbf{x}(t_s))t_{s+1} - f(\mathbf{x}(t_{s+1}))t_s}{f(\mathbf{x}(t_s)) - f(\mathbf{x}(t_{s+1}))} \right\}. \quad (10)$$

It is worth noting that IDR [33] employs a similar strategy for ray marching in a surface rendering pattern, using a recursive secant root-finding algorithm in case the sphere tracing method does not converge. In contrast, we reconstruct the surface using volume rendering and directly approximate the ray-surface crossing with only a single iteration of the secant method thanks to the hierarchical sampling strategy.

After deriving the surface point $\hat{\mathbf{x}}$, we compare features of this point across multiple views. Similar to MVSDF [37], we extract feature from RGB images with a convolutional neural network (CNN) that is pre-trained for supervised MVS [36]. Then we constrain the neural implicit surface reconstruction using a multi-view feature consistency loss:

$$\mathcal{L}_{feat.} = \frac{1}{N_c N_v} \sum_{i=1}^{N_v} \left| \mathbf{F_0}(\mathbf{p}_0) - \mathbf{F}_i(\mathbf{K}_i(\mathbf{R}_i \hat{\mathbf{x}} + \mathbf{t}_i)) \right|, \quad (11)$$

where N_v and N_c are the numbers of neighboring source views and feature channels respectively, $\mathbf{F}$ is the extracted feature map, $\mathbf{p}_0$ is the pixel through which the ray casts, $\{\mathbf{K}_i, \mathbf{R}_i, \mathbf{t}_i\}$ are the camera parameters of the i-th source view.

3.4. Training Loss

The overall loss function to train our neural implicit surface reconstruction network is defined as the weighted sum of the following four terms:

$$\mathcal{L} = \mathcal{L}_{color} + \alpha \mathcal{L}_{eik.} + \beta \mathcal{L}_{bias} + \gamma \mathcal{L}_{feat.}. \quad (12)$$

$\mathcal{L}_{color}$ is the difference between the RGB color taken from ground truth input images C and that from volume rendering $\hat{C}$:

$$\mathcal{L}_{color} = \frac{1}{m} \sum_{i=1}^{m} \left| C_i - \hat{C}_i \right|, \quad (13)$$

Scan	24	37	40	55	63	65	69	83	97	105	106	110	114	118	122	means
IDR [33]	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
MVSDF [37]	0.83	1.76	0.88	0.44	1.11	0.90	0.75	1.26	1.02	1.35	0.87	0.84	0.34	0.47	0.46	0.88
NeuS [26]	0.83	0.98	0.56	0.37	1.13	0.59	0.60	1.45	0.95	0.78	0.52	1.43	0.36	0.45	0.45	0.77
RegSDF [35]	0.60	1.41	0.64	0.43	1.34	0.62	0.60	0.90	0.92	1.02	0.60	0.60	0.30	0.41	0.39	0.72
COLMAP [24]	0.81	2.05	0.73	1.22	1.79	1.58	1.02	3.05	1.40	2.05	1.00	1.32	0.49	0.78	1.17	1.36
VolSDF [32]	1.14	1.26	0.81	0.49	1.25	0.70	0.72	1.29	1.18	0.70	0.66	1.08	0.42	0.61	0.55	0.86
NeuS [26]	1.00	1.37	0.93	0.43	1.10	0.65	0.57	1.48	1.09	0.83	0.52	1.20	0.35	0.49	0.54	0.84
NeuralWarp [7]	0.49	**0.71**	0.38	**0.38**	**0.79**	0.81	0.82	**1.20**	1.06	**0.68**	0.66	0.74	0.41	0.63	0.51	0.68
D-NeuS (Ours)	**0.44**	0.79	**0.35**	0.39	0.88	**0.58**	**0.55**	1.35	**0.91**	0.76	**0.40**	**0.72**	**0.31**	**0.39**	**0.39**	**0.61**

Table 1: Quantitative results of the Chamfer distances on DTU dataset (lower values are better). COLMAP results are achieved by trim=0. The upper part of the table are the neural implicit surfaces reconstruction methods that require additional scene-specific prior inputs such as object masks, depth maps or point clouds). The best score for each scan is marked in **bold** and the second-best one is underlined.

where m is the number of pixels trained in a batch. Following previous works [33, 37, 32, 26, 7], we add an Eikonal loss [10] on the sampled points to regularize the gradients of SDF field from the geometry network f:

$$\mathcal{L}_{eik.} = \frac{1}{|\mathbb{P}|} \sum_{\mathbf{x} \in \mathbb{P}} (\|\nabla f(\mathbf{x})\|_2 - 1)^2, \tag{14}$$

where $\mathbb{P}$ is the set of all sampled points in a batch, and $\|\cdot\|_2$ is the L2 norm.

4. Experiments

4.1. Experimental Settings

Datasets. To evaluate our method with full losses described in Section 3.4 on the DTU dataset [12], we follow previous works [33, 32, 26, 7] and select the same 15 models for comparison. Each scene contains 49 or 64 images at 1200×1600 resolution with camera parameters. Objects in DTU dataset have various geometries, appearances and materials, including non-Lambertian surfaces and thin structures. Furthermore, we test on 6 challenging scenes from the BlendedMVS dataset [31]. BlendedMVS dataset provides images at a resolution of 576×768 with more complex backgrounds and various numbers of views vary from 24 to 143. For DTU dataset, the reconstructed surfaces are evaluated quantitatively by metrics of the Chamfer distance in mm, while we demonstrate visual comparison of the reconstruction results on BlendedMVS dataset.

Baselines. We compare the proposed method to a traditional MVS pipeline COLMAP [24], and state-of-the-art learning-based approaches: IDR [33], MVSDF [37], VolSDF [32], NeuS [26], NeuralWarp [7].

Network architecture. Similar to [33, 26], our geometry network f is modeled by an MLP including 8 hidden layers with size of 256 and a skip connection from the input to the output of the 4-th layer. The weights of the geometry network are initialized to approximate the SDF field of a unit sphere [2]. The radiance network c is an MLP consisting of 4 hidden layers MLP with 256 hidden cells. Position encoding is applied to $\mathbf{x}$ with 6 frequencies and $\mathbf{v}$ with 4 frequencies. For volume rendering, we adopt the hierarchical sampling strategy in NeuS [26], sampling 512 rays for each iteration, with 64 coarse and 64 fine sampled points per ray, and additional 32 points outside the unit sphere following NeRF++ [38]. For multi-view features consistency, we compare each reference view with $N_v = 2$ neighboring source views using $N_c = 32$ feature channels.

Training details. To train our networks, we adopt Adam optimizer [14] using the learning rate $5e^{-4}$ with warm-up period of 5k iterations before decaying by cosine to the minimal learning rate of $2.5e^{-5}$. We initialize the trainable standard deviation for the logistic density distribution for the volume rendering weight with 0.3. We train our model for 300k iterations for 19 hours on a single NVIDIA Titan RTX graphics card. In terms of inference, rendering an image of resolution 1200×1600 using standard volume rendering takes approximately 7 minutes. As for the weighting factors of losses in Eqn. 12, we fix the Eikonal weight α as 0.1 for the whole training. In addition, inspired by MVSDF [37], we divide the training in three stages. In the first 50k iterations, we set the geometry bias loss weight β as 0.01. From 50k to 150k iterations, we set β as 0.1 and the feature consistency loss weight γ as 0.5, while in the remaining iterations, β and γ are 0.01 and 0.05, respectively. After optimization, we apply Marching Cubes [16] to extract a mesh from the SDF field f in a predefined bounding box with the volume size of 512^3 voxels, which takes about 57 seconds.

4.2. Comparisons

To evaluate the surface geometry reconstruction quality, we follow previous works and use the official evaluation code to calculate the Chamfer L_1 distance, which is

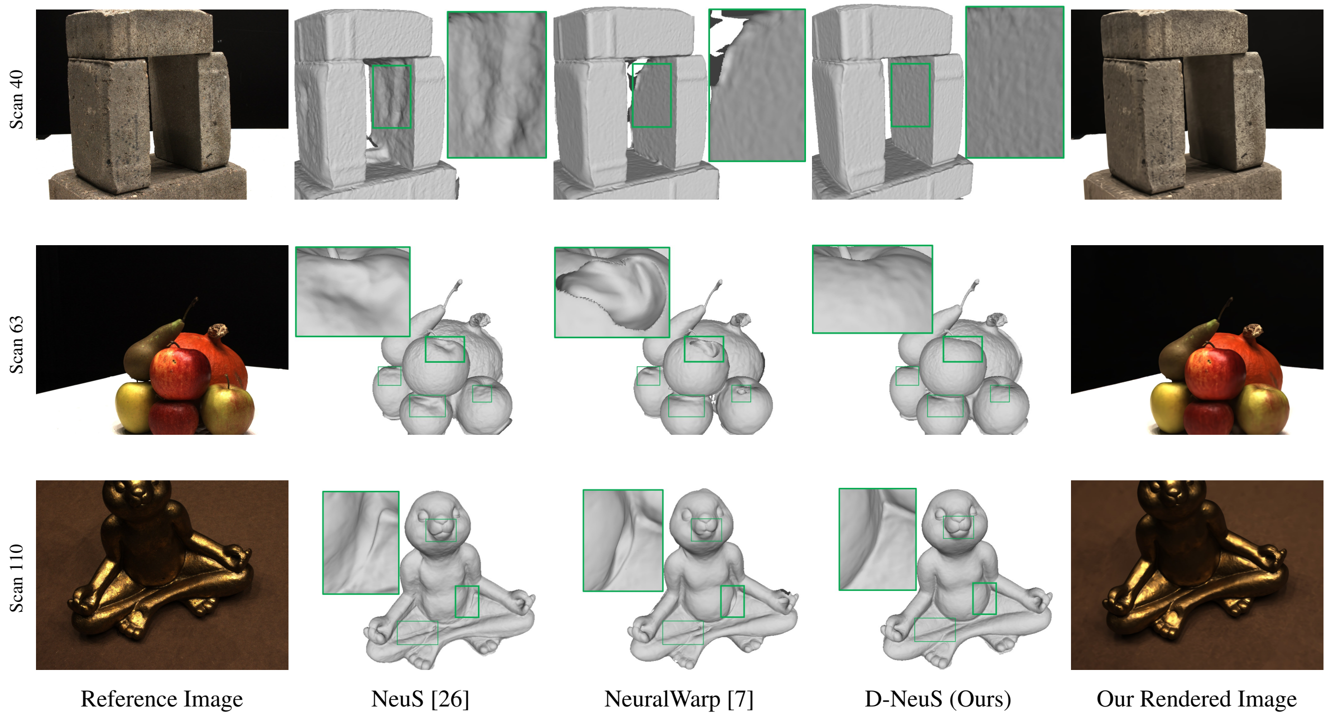


Figure 4: Comparisons on surface reconstruction in DTU dataset.

Scan	24	37	40	55	63	65	69	83	97	105	106	110	114	118	122	means
NeRF [18]	26.24	25.74	26.79	27.57	31.96	31.50	29.58	32.78	28.35	32.08	33.49	31.54	31.00	35.59	35.51	30.65
VolSDF [32]	26.28	25.61	26.55	26.76	31.57	31.50	29.38	33.23	28.03	32.13	33.16	31.49	30.33	34.90	34.75	30.38
NeuS [26]	28.20	27.10	28.13	28.80	32.05	33.75	**30.96**	**34.47**	29.57	32.98	**35.07**	**32.74**	**31.69**	**36.97**	37.07	31.97
D-NeuS (Ours)	**28.98**	**27.58**	**28.40**	**28.87**	**33.71**	**33.94**	30.94	34.08	**30.75**	**33.73**	34.84	32.41	31.42	36.76	**37.17**	**32.22**

Table 2: Qualitative results on DTU dataset in terms of PSNR, evaluating the rendering quality.

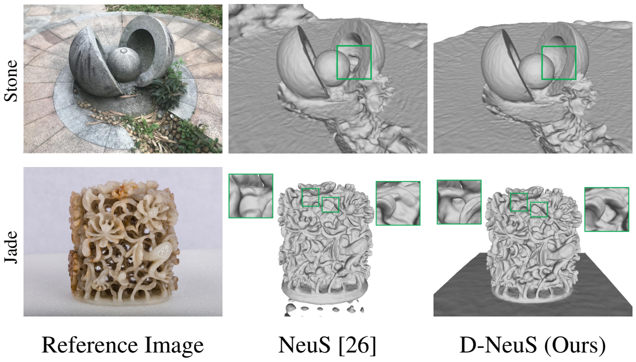
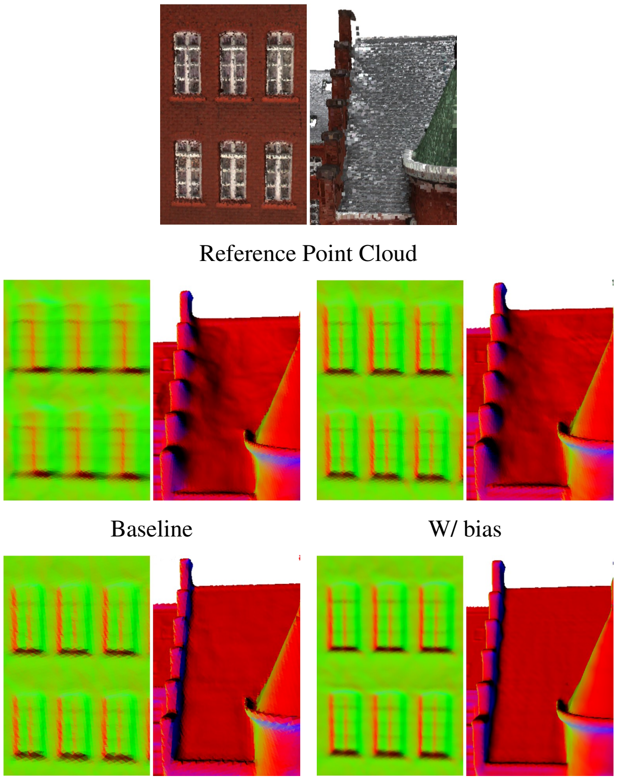


Figure 5: Comparisons on surface reconstruction in BlendedMVS dataset.

the average of accuracy (mean distance from sampled point clouds of the reconstructed surface to the ground truth point cloud) and completeness (mean distance from the ground truth point cloud to the reconstructed counterpart). Similar to previous works [33, 21, 26, 32, 7], we clean the extracted meshes with the object masks dilated by 50 pixels. Table 1 shows the mean Chamfer distances of our work and baselines. Results of the baselines are reported in their original papers, except for COLMAP whose result is taken from [26]. Following previous works, we focus on the comparison of the approaches requiring no additional per-scene prior knowledge including object masks, depth maps or point clouds. As illustrated in Table 1, our method surpasses the baselines by a noticeable margin and achieve the lowest mean Chamfer distance.

Fig. 4 qualitatively compares the reconstructed surface geometry of our method and the baselines on DTU [12] dataset. Surfaces from NeuS [26] are more noisy and bumpy, especially in the plane (Scan 40) or smooth region (Scan 63 and 110), while NeuralWarp struggles to reconstruct the surface boundaries (Scan 40 and 110) and non-Lambertian area (Scan 63). In contrast, our method is ro-


Reference Point Cloud

Baseline

W/ bias

W/ feature

Ours

Figure 6: Qualitative results of ablation study on DTU dataset. To better illustrate the geometry details, we visualize the surface normals of two challenging regions.

bust to these challenging cases, recovering fine geometry details with high accuracy and fidelity. In addition to surface geometry, D-NeuS also achieves photorealistic image rendering. As reported in Table 2, we quantitatively evaluate the PSNR of rendering results from our method, which outperforms other state-of-the-art methods. Following previous works, we only evaluate the PSNR of the pixels inside the object masks provided by IDR [33].

Qualitative results on BlendedMVS dataset [31] are shown in Figure 5. Our method is robust to challenging surfaces, such as the seriously occluded area in Stone, as well as highly complex concave holes in Jade, while NeuS struggles to recover the fine geometric details of these complicated surfaces.

4.3. Ablation Study

We evaluate different components of our method by an ablation study on the DTU dataset. Specifically, we use NeuS [26] as the baseline on which we build D-NeuS, and progressively combine our proposed losses. As demonstrated in Fig. 6, the geometry bias loss successfully recovers the fine geometric details of the windows, while multi-view feature consistency loss faithfully reconstructs the surface boundary: the connecting part between the roof and the facade.

	$\mathcal{L}_{color}$	$\mathcal{L}_{bias}$	$\mathcal{L}_{feature}$	Mean Chamfer
Baseline	✓			0.84
W/ bias	✓	✓		0.76
W/ feature	✓		✓	0.63
Ours	✓	✓	✓	0.61

Table 3: Quantitative results of ablation study on DTU dataset.

Table 3 shows the quantitative result using the mean Chamfer distance. Both contributions of our work can improve the surface reconstruction, and D-NeuS combine their advantages for the best performance.

5. Discussion

Limitations. Similar to other neural implicit surface reconstruction methods, training our model takes some hours for each scene. In addition, the rendered images struggle to recover the high-frequency patterns in the input images. Lastly, a certain number of dense input views are required for high-quality reconstruction.

Future works. One interesting future direction is to represent the scene with a multi-resolution structure, *e.g.*, Instant-ngp [19], for fast optimization as well as high-frequency local details. Moreover, it is promising to generalize the reconstruction to new scenes using learned image priors like [34] or geometry priors like [13], which may also enable surface reconstruction from a sparse set of views.

Conclusions. We introduce D-NeuS, a volume rendering-based neural implicit surface reconstruction method recovering fine-level geometric details. We analyze the cause for geometry bias between the SDF field and the volume rendered color, and propose a novel loss function to constrain the bias. In addition, we apply multi-view feature consistency to surface points derived by interpolating the zero-crossing from sampled SDF values. Extensive experiments on different datasets show that D-NeuS is able to reconstruct high-quality surfaces with fine details and outperforms the state of the art both qualitatively and quantitatively.

Acknowledgement

This work has partly been funded by the H2020 European project Invictus under grant agreement no. 952147 as well as by the Investitionsbank Berlin with financial support by European Regional Development Fund (EFRE) and the government of Berlin in the ProFIT research project KIVI.

References

[1] Tanks and temples benchmark. `https://www.tanksandtemples.org/leaderboard/`.

[2] Matan Atzmon and Yaron 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.

[3] Jonathan T. Barron, Ben Mildenhall, Dor Verbin, Pratul P. Srinivasan, and Peter Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. *CVPR*, 2022.

[4] Michael Bleyer, Christoph Rhemann, and Carsten Rother. Patchmatch stereo-stereo matching with slanted support windows. In *Bmvc*, volume 11, pages 1–11, 2011.

[5] Chenjie Cao, Xinlin Ren, and Yanwei Fu. Mvsformer: Learning robust image representations via transformers and temperature-based depth for multi-view stereo. *arXiv preprint arXiv:2208.02541*, 2022.

[6] Shuo Cheng, Zexiang Xu, Shilin Zhu, Zhuwen Li, Li Erran Li, Ravi Ramamoorthi, and Hao Su. Deep stereo using adaptive thin volume representation with uncertainty awareness. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2524–2534, 2020.

[7] François Darmon, Bénédicte Bascle, Jean-Clément Devaux, Pascal Monasse, and Mathieu Aubry. Improving neural implicit surfaces geometry with patch warping. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6260–6269, 2022.

[8] Yikang Ding, Wentao Yuan, Qingtian Zhu, Haotian Zhang, Xiangyue Liu, Yuanjiang Wang, and Xiao Liu. Transmvsnet: Global context-aware multi-view stereo network with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8585–8594, 2022.

[9] S. Galliani, K. Lasinger, and K. Schindler. Massively parallel multiview stereopsis by surface normal diffusion. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 873–881, 2015.

[10] Amos Gropp, Lior Yariv, Niv Haim, Matan Atzmon, and Yaron Lipman. Implicit geometric regularization for learning shapes. *arXiv preprint arXiv:2002.10099*, 2020.

[11] X. Gu, Z. Fan, S. Zhu, Z. Dai, F. Tan, and P. Tan. Cascade cost volume for high-resolution multi-view stereo and stereo matching. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2492–2501, 2020.

[12] Rasmus Jensen, Anders Dahl, George Vogiatzis, Engil Tola, and Henrik Aanæs. Large scale multi-view stereopsis evaluation. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 406–413. IEEE, 2014.

[13] Mohammad Mahdi Johari, Yann Lepoittevin, and François Fleuret. Geonerf: Generalizing nerf with geometry priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18365–18375, 2022.

[14] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[15] Arno Knapitsch, Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Tanks and temples: Benchmarking large-scale scene reconstruction. *ACM Transactions on Graphics (ToG)*, 36(4):1–13, 2017.

[16] William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3d surface construction algorithm. *ACM siggraph computer graphics*, 21(4):163–169, 1987.

[17] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[18] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *European conference on computer vision*, pages 405–421. Springer, 2020.

[19] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *arXiv preprint arXiv:2201.05989*, 2022.

[20] Michael Niemeyer, Lars Mescheder, Michael Oechsle, and Andreas Geiger. Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3504–3515, 2020.

[21] Michael Oechsle, Songyou Peng, and Andreas 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.

[22] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[23] Songyou Peng, Michael Niemeyer, Lars Mescheder, Marc Pollefeys, and Andreas Geiger. Convolutional occupancy networks. In *European Conference on Computer Vision*, pages 523–540. Springer, 2020.

[24] Johannes Lutz Schönberger, Enliang Zheng, Marc Pollefeys, and Jan-Michael Frahm. Pixelwise view selection for unstructured multi-view stereo. In *European Conference on Computer Vision (ECCV)*, 2016.

[25] Fangjinhua Wang, Silvano Galliani, Christoph Vogel, Pablo Speciale, and Marc Pollefeys. Patchmatchnet: Learned multi-view patchmatch stereo, 2021.

[26] Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. Neus: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. *arXiv preprint arXiv:2106.10689*, 2021.

[27] Q. Xu and W. Tao. Multi-scale geometric consistency guided multi-view stereo. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5478–5487, 2019.

[28] Jiayu Yang, Wei Mao, Jose M Alvarez, and Miaomiao Liu. Cost volume pyramid based depth inference for multi-view

stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4877–4886, 2020.

[29] Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. Mvsnet: Depth inference for unstructured multi-view stereo. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 767–783, 2018.

[30] Yao Yao, Zixin Luo, Shiwei Li, Tianwei Shen, Tian Fang, and Long Quan. Recurrent mvsnet for high-resolution multi-view stereo depth inference. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5525–5534, 2019.

[31] Yao Yao, Zixin Luo, Shiwei Li, Jingyang Zhang, Yufan Ren, Lei Zhou, Tian Fang, and Long Quan. Blendedmvs: A large-scale dataset for generalized multi-view stereo networks. *Computer Vision and Pattern Recognition (CVPR)*, 2020.

[32] Lior Yariv, Jiatao Gu, Yoni Kasten, and Yaron Lipman. Volume rendering of neural implicit surfaces. *Advances in Neural Information Processing Systems*, 34:4805–4815, 2021.

[33] Lior Yariv, Yoni Kasten, Dror Moran, Meirav Galun, Matan Atzmon, Basri Ronen, and Yaron Lipman. Multiview neural surface reconstruction by disentangling geometry and appearance. *Advances in Neural Information Processing Systems*, 33, 2020.

[34] Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelnerf: Neural radiance fields from one or few images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4578–4587, 2021.

[35] Jingyang Zhang, Yao Yao, Shiwei Li, Tian Fang, David McKinnon, Yanghai Tsin, and Long Quan. Critical regularizations for neural surface reconstruction in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6270–6279, 2022.

[36] Jingyang Zhang, Yao Yao, Shiwei Li, Zixin Luo, and Tian Fang. Visibility-aware multi-view stereo network, 2020.

[37] Jingyang Zhang, Yao Yao, and Long Quan. Learning signed distance field for multi-view surface reconstruction. *International Conference on Computer Vision (ICCV)*, 2021.

[38] Kai Zhang, Gernot Riegler, Noah Snavely, and Vladlen Koltun. Nerf++: Analyzing and improving neural radiance fields. *arXiv preprint arXiv:2010.07492*, 2020.