Towards Unbiased Volume Rendering of Neural Implicit Surfaces with Geometry Priors

Yongqiang Zhang[1], Zhipeng Hu[1], Haoqian Wu[1], Minda Zhao[1], Lincheng Li[1]*, Zhengxia Zou[2], Changjie Fan[1]
[1]NetEase Fuxi AI Lab, [2]Beihang University
{zhangyongqiang02,zphu,wuhaoqian,zhaominda01,lilincheng}@corp.netease.com,
zhengxiazou@buaa.edu.cn, fanchangjie@corp.netease.com

Abstract

Learning surface by neural implicit rendering has been a promising way for multi-view reconstruction in recent years. Existing neural surface reconstruction methods, such as NeuS [24] and VolSDF [32], can produce reliable meshes from multi-view posed images. Although they build a bridge between volume rendering and Signed Distance Function (SDF), the accuracy is still limited. In this paper, we argue that this limited accuracy is due to the bias of their volume rendering strategies, especially when the viewing direction is close to be tangent to the surface. We revise and provide an additional condition for the unbiased volume rendering. Following this analysis, we propose a new rendering method by scaling the SDF field with the angle between the viewing direction and the surface normal vector. Experiments on simulated data indicate that our rendering method reduces the bias of SDF-based volume rendering. Moreover, there still exists non-negligible bias when the learnable standard deviation of SDF is large at early stage, which means that it is hard to supervise the rendered depth with depth priors. Alternatively we supervise zero-level set with surface points obtained from a pre-trained Multi-View Stereo network. We evaluate our method on the DTU dataset and show that it outperforms the state-of-the-arts neural implicit surface methods without mask supervision.

1. Introduction

3D reconstruction is an important task in 3D games and AR/VR applications. As a key technique in computer vision and graphics, recovering surfaces and textures from Multi-View calibrated RGB images has been widely studied in recent decades. Early unsupervised Multi-View Stereo (MVS) approaches [14, 20] provide solutions through a certain multistage pipeline, including grouping the related views, depth prediction, filtering with photometric consistency and geometry consistency, fusion of points from different views, meshing the dense points by off-the-shelf methods such as screened Poisson Surface Reconstruction [8], and texture mapping finally.

Later MVS networks [5,20,27,36] are developed rapidly benefiting from the available large-scale 3D datasets. This kind of MVS networks use Convolutional Neural Network (CNN) to predict depth maps effectively, then follow the traditional pipeline to fuse a global dense point cloud and mesh it. However, MVS networks suffer from texture-less regions and sudden depth changes, so there usually exist many holes in the recovered meshes.

Recently, neural implicit surface and differentiable rendering methods present a promising way to improve and simplify the progress of the Multi-View 3D reconstruction. The surfaces are represented as Signed Distance Functions (SDF) [18, 24, 32, 33] or occupancy field [16, 17]. At the same time, neural radiance field [13, 35] are proposed with different volume rendering. The neural surface-based rendering method can recover reliable and smooth surfaces, but it is hard to train without mask supervision. On the contrary, the different volume rendering can achieve good 2D views without mask supervision, but the quality of 3D geometry is rather coarse.

Is there some connections between the SDF field and occupancy field? NeuS [24] and VolSDF [32] point that the connection can be conducted with a certain Cumulative Distribution Function (CDF). Thanks to this significant progress, it is able to learn 3D surfaces effectively from neural implicit surface with the self-supervised volume rendering. The necessary input can only be well-posed 2D images. Masks could be removed, because it is hard to obtain accurate masks for many complex objects in the real world.

Although these great methods have made big progress on 3D reconstruction from calibrated multi-view images,

*Corresponding author

the accuracy of meshes is still limited compared with those methods [5, 14, 20, 20, 27, 36] in a classical pipeline. We find that there exist unexpected convex or concave surfaces in regions with poor texture or strong highlight. In addition, the learned surface and rendered color tend to be smooth, and the high frequency details can not be captured well.

Based on NeuS [24] and VolSDF [32], we further analyze the precision of the bridge between the SDF field and density field, and we found that there exist a bias between real depth and rendered depth by SDF-based volume rendering. We argue that there are two factors leading to the bias: (1) The angle between the view direction and the normal vector; (2) The learnable standard deviation of the SDF field. The bias increases with the growth of the angle. It decreases with the descent of the learnable standard deviation, but it is still relatively large when the deviation is not small enough at early training stage. More details of the analysis are described in Section 3. In order to reduce the bias caused by the various angles, we modify the transformation between the SDF field and the density field. Furthermore, we adopt dense point clouds predicted by an accessible MVS network to reduce the bias further. Finally we evaluate our method and compare it with other SDF-based volume rendering methods on the public benchmark.

To summarise, we provides the following three contributions: a) We amend the conditions of unbiased SDF-based volume rendering, and analyze the bias of VolSDF [32] and NeuS [24]. b) We propose a new transformation between the SDF field and the density field, which does not require a plane assumption and outperforms VolSDF [32] and NeuS [24] even without geometry priors. In particular, we scale the SDF field by the inverse of the angle between the view direction and the normal vector. c) Geometry priors from a pre-trained MVS network are used with the annealing sampling to further reduce the bias effectively at early training stage and boost the reconstruction quality.

2. Related Work

Multi-view Stereo: Traditional Multi-View Stereo methods reconstruct 3D point clouds from predicted depth maps [5, 20, 27, 36]. As a typical PatchMatch-based MVS method, COLMAP [20] optimizes depth and normal maps with a graph model. Then a depth fusion step is processed to get a dense point cloud, finally a meshing algorithm like screened Poisson Surface Reconstruction [8] is used to reconstruct surface. Benefiting from deep learning, supervised MVS methods have also become popular [6, 23, 28–30]. These methods show impressive performance on multiple benchmarks [7, 9], but they have to be trained on specific datasets [7, 31]. CasMVSNet [6] apply the cascade cost volume to the representative MVS-Net [29]. In this work, the single cost volume is decomposed into a cascade formulation of multiple stages, thus depth range and total number of hypothesis planes at each stage can be reduced. This coarse-to-fine structure remarkably decreases time cost and GPU memory consumption, and it also achieves the state-of-the-art performance on multiple benchmarks [7, 9].

Neural implicit surface: Recently, with the development of differentiable rendering, neural implicit surface has been introduced, which represent a surface as SDF [18] or an occupancy field by a neural network [12, 19]. Moreover, these representations are combined with surface rendering [16, 22, 33, 34]. Different with data-driven MVS networks, Implicit Differentiable Renderer (IDR) [33] is an end-to-end self-supervised neural system. Although IDR [33] can learn 3D surface, appearance, and cameras from posed images and noisy poses, it is heavily dependent on silhouette masks. Drift of boundaries may result in either inaccurate surfaces. Based on IDR [33], geometry constraints are introduced in MVSDF [34] to improve the mesh quality and relax the requirement of masks. They take advantage of the knowledge of stereo matching and feature consistency to optimize the implicit surface representation. Surface-rendering methods assume that the color of a ray only relies on the color of the first intersection of the ray with the surface, which makes the gradient only backpropagated to a local region near the intersection [24]. Thus such methods are hard to handle with severe self-occlusions and sudden depth changes.

Neural volume rendering: As a famous neural volume rendering method, NeRF [3, 13, 15, 35] combines classical volumetric rendering with implicit function to render high-quality 2D images, but it can not export high-quality geometry. Recent works improve the geometric network and build connections between density-based representation and surface-based representation [2, 17, 21, 24, 32], which can extract more accurate and smooth surfaces. VolSDF [32] further models the volume density as Laplace Cumulative Distribution Function applied to a SDF representation. Benefiting from this simple but effective density representation, a bound on the opacity approximation error can be deduced, leading to more accurate sampling of the volume rendering integral. Moreover, the background is modeled using an additional NeRF network [35] to predict the point density and radiance field outside of the focused object. Thus it can reconstruct surfaces even without silhouette masks. Similar to VolSDF [32], NeuS [24] transform the SDF field to the accumulated transmittance for volume rendering with the Logistic Cumulative Distribution Function, and it is pointed that the rendered weight should reach maximum at the first intersection point from outside to inside. However, it is limited by the first-order approximation of local plane, and the peak of weights declines with the growth of the angle between the view direction and the normal vector. Due to these limitations, it tends to learn a smooth surface with less

high frequency details, and unexpected concave-convex regions occur on the recovered meshes. NeuralWarp [2] further use image warping in combination with volumetric rendering to improve the performance of VolSDF [32]. Firstly, the original VolSDF [32] is trained for 50k iterations with batches of 1024 pixel as initialization. Then the geometry and radiance networks are finetuned by minimizing the sum of the volumetric rendering loss and the patch warping loss. Although the patch warping loss works very well, it particularly depends on the initialized VolSDF [32].

More recently, there are some concurrent methods proposed for implicit surface reconstruction. Some methods focus on improving geometric details, such as Geo-Neus [4] and HF-NeuS [25]. Others aim to enhance training efficiency via voxel-based representation, such as Voxurf [26] and Vox-Surf [10]

3. Method

In this section, we introduce the conditions of the unbiased SDF-based volume rendering, and we analyze the bias of the density as transformed SDF with a certain Cumulative Distribution Function. Then we provide a novel transformation from SDF to volume density, which satisfies the unbiased conditions above. Moreover, we find that the bias is still non-negligible when the learnable scale is not small enough at early training stage. Therefore we supervise the zero-level set with dense point clouds predicted by a pre-trained MVS network. Finally, we present our full optimization.

3.1. Unbiased SDF-based Volume Rendering

The Signed Distance Function $f(\mathbf{p})$ means the minimal signed distance between a point $\mathbf{p}$ and the surface. The surface of the object can be is represented by the zero-level set of its SDF, that is,

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

Consistent with previous works, SDF is encoded by Multilayer Perceptrons (MLP). We also use MLP to encode the color related to a point $\mathbf{p}$ and a unit viewing direction $\mathbf{v}$. The ray emitted from the camera center $\mathbf{o}$ at the viewing direction $\mathbf{v}$ is denoted as $\{\mathbf{p}(t) = \mathbf{o} + t\mathbf{v} | t \geq 0\}$.

In the meanwhile, a classical volumetric rendering of radiance field can be represented as the accumulation of colors along the ray, that is

$$\begin{array}{ll} T(t) = \exp(-\int_0^t \sigma(u)\mathrm{d}u), & w(t) = T(t)\sigma(t) \\ \hat{C} = \int_0^{+\infty} w(t)c(\mathbf{p}(t), \mathbf{v})\mathrm{d}t, & \hat{t} = \int_0^{+\infty} w(t)t\mathrm{d}t' \end{array}, \tag{2}$$

where $\sigma(t)$ is the volume density, $T(t)$ is the accumulated transmittance along the ray, and $w(t)$ can be regarded as the weight function along the ray. $\hat{C}$ and $\hat{t}$ denote the rendered color and depth along the ray, respectively.

Conditions of the unbiased SDF-based volume rendering. It is very important to build an accurate transformation from the SDF field to the radiance field. Here we focus on opaque objects, so the rendered depth $\hat{t}$ should be equal to the distance t^* between the first intersection point and the camera center along the ray, which is formulated as

$$\hat{t} - t^* = 0, \ where \ t^* = \min\{t | f(\mathbf{p}(t)) = 0\} \tag{3}$$

It is noticed that this ideal condition is hard to be solved directly. A possible solution is modeling the weight function $w(t)$ as the Dirac delta function around t^*. In other words, $w(t)$ tends to be infinite when t gets close to t^*, and tends to be zero when t moves away from t^*. Inspired by NeuS [24], relaxed conditions are given: (a) The derivative of $w(t)$ respect to t is equal to zero at the intersection point when the ray going from outside to inside. (b) It is greater than zero when the ray gets close to the surface from outside to inside. (c) It is less than zero when the ray goes in the surface or goes from inside to outside. That is

$$\begin{cases} \frac{\mathrm{d}w}{\mathrm{d}t}(t) = 0, & \text{if } f(\mathbf{p}(t)) = 0 \text{ and } f'(\mathbf{p}(t)) < 0 \\ \frac{\mathrm{d}w}{\mathrm{d}t}(t) > 0, & \text{if } f(\mathbf{p}(t)) > 0 \text{ and } f'(\mathbf{p}(t)) < 0 \\ \frac{\mathrm{d}w}{\mathrm{d}t}(t) < 0, & \text{if } f(\mathbf{p}(t)) < 0 \text{ or } f'(\mathbf{p}(t)) > 0 \end{cases}, \tag{4}$$

where $f'(\mathbf{p}(t)) = \nabla f(\mathbf{p}(t)) \cdot \mathbf{v} = \mathbf{n}(t) \cdot \mathbf{v}$, which means the cosine of the angle between the normal vector $\mathbf{n}(t)$ and the viewing direction $\mathbf{v}$. $f'(\mathbf{p}(t))$ is negative when the ray goes from outside to inside, positive on the contrary. It is zero when the ray is tangent to the surface.

If these conditions are met, $w(t)$ will reach local maximum at each intersection point from outside to inside. Benefiting from the property of the volume rendering, it will reach global maximum at the first intersection point.

3.2. Transformation from SDF to density

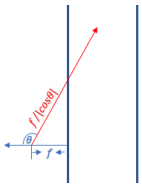
Following the conditions above, we analyze the bias of the existing SDF-based volume rendering methods, and then we present a new rendering method by scaling the SDF field with the angle between the viewing direction and the surface normal vector.

According to VolSDF [32], the density can be modeled by SDF with a scaled Cumulative Distribution Function, such as Laplace distribution or Logistic distribution. It is formulated as

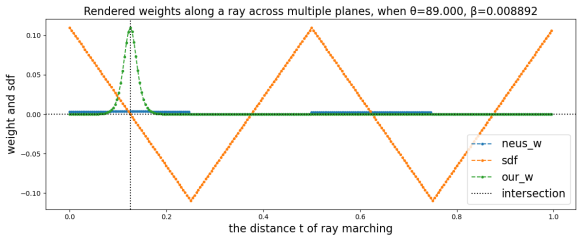
$$\sigma(\mathbf{p}(t)) = \alpha \Psi_\beta(-f(\mathbf{p}(t))), \tag{5}$$

where Ψ_β denotes the CDF, $\beta > 0$ is a learnable deviation, and $\alpha = 1/\beta$.

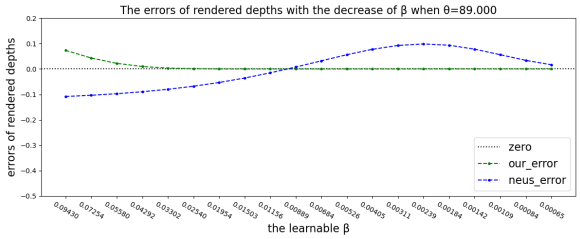
Considering any intersection point $\bar{t} \in \{t | f(\mathbf{p}(t)) = 0\}$,



(a) A ray across multiple planes with a large angle.

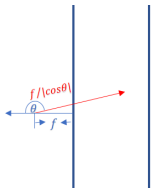


(b) Weights rendered by NeuS and Our model with a specific β.

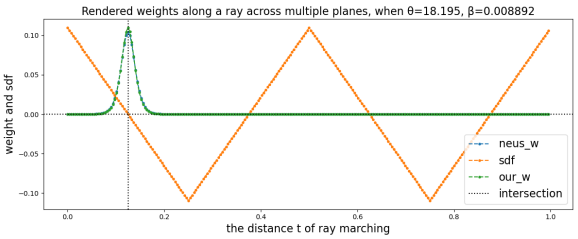


(c) Biases of $\hat{t}$ rendered by NeuS and Our model, with the descent of β.

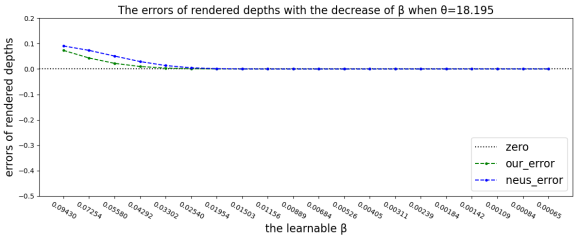
Figure 1. Simulation of SDF-based rendering when a ray goes across multiple planes with a large angle. In subfigures (b) and (c), green and blue lines represent our method and NeuS, respectively. It is noted that the weights of NeuS (blue) are close to 0 in this case.



(a) A ray across multiple planes with a small angle.



(b) Weights rendered by NeuS and Our model with a specific β.



(c) Biases of $\hat{t}$ rendered by NeuS and Our model, with the descent of β.

Figure 2. Simulation of SDF-based rendering when a ray goes across multiple planes. It is noted that the weights of our (green) and NeuS (blue) are very close in this case.

the derivative of $w(\bar{t})$ respect to $\bar{t}$ is

$$\begin{aligned}\frac{\mathrm{d}w}{\mathrm{d}t}(\bar{t}) &= T(\bar{t})\left(\sigma'(\mathbf{p}(\bar{t})) - \sigma(\mathbf{p}(\bar{t}))^2\right) \\ &= T(\bar{t})\left(-\alpha\Psi_\beta'(0)f'(\mathbf{p}(\bar{t})) - \alpha^2\Psi_\beta(0)^2\right) \\ &= \begin{cases} T(\bar{t})\left(-\frac{1}{2\beta^2}f'(\mathbf{p}(\bar{t})) - \frac{1}{4\beta^2}\right), & if\ \text{Laplace} \\ T(\bar{t})\left(-\frac{1}{4\beta^2}f'(\mathbf{p}(\bar{t})) - \frac{1}{4\beta^2}\right), & if\ \text{Logistic} \end{cases}\end{aligned} \tag{6}$$

We can find that the derivative of $w(\bar{t})$ is zero only when $f'(\mathbf{p}(\bar{t})) = -0.5$ for the Laplace CDF, or $f'(\mathbf{p}(\bar{t})) = -1$ for the Logistic CDF. Since $T(\bar{t}) > 0$, the derivative of $w(\bar{t})$ is less than zero when $f'(\mathbf{p}(\bar{t})) > -0.5$ for the Laplace CDF, and is greater than zero when $f'(\mathbf{p}(\bar{t})) < -0.5$. As for the Logistic CDF, the derivative of $w(\bar{t})$ is less than zero when $f'(\mathbf{p}(\bar{t})) > -1$. Therefore $w(\bar{t})$ is biased with various angles between the normal vector and the view direction. The bias tends to be larger when the ray gets closer to the tangential direction.

Different with VolSDF [32], NeuS [24] models the accumulated transmittance $T(t)$ as the Logistic CDF of SDF. The density and weight function are defined as

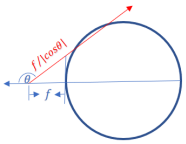
$$\begin{aligned}\sigma(\mathbf{p}(t)) &= \max\left(\frac{-f'(\mathbf{p}(t))\Psi_\beta'(f(\mathbf{p}(t)))}{\Psi_\beta(f(\mathbf{p}(t)))}, 0\right) \\ w(\mathbf{p}(t)) &= \max\left(-f'(\mathbf{p}(t))\Psi_\beta'\left(f(\mathbf{p}(t))\right), 0\right)\end{aligned} \tag{7}$$

The derivative of $w(\bar{t})$ respect to $\bar{t}$ at any intersection point from outside to inside is

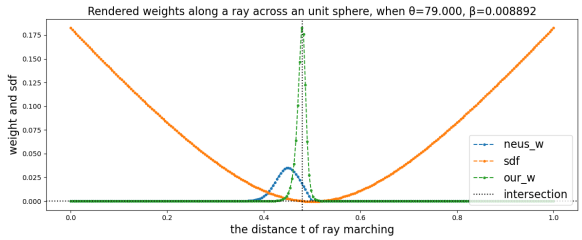
$$\frac{\mathrm{d}w}{\mathrm{d}t}(\bar{t}) = -\frac{1}{4\beta}f''(\mathbf{p}(\bar{t})). \tag{8}$$

It is only equal to zero when $f''(\mathbf{p}(\bar{t})) = 0$, which is a first-order approximation of local plane. If the local surface is convex, $f''(\mathbf{p}(\bar{t})) > 0$. On the contrary, $f''(\mathbf{p}(\bar{t})) < 0$ for the locally concave surface. It means that $w(\bar{t})$ is biased when the local surface is not a plane. Even if it can promise that $w(\bar{t})$ always reaches local maximum for a local plane, $w(\bar{t})$ is reduced much by the cosine value when the viewing direction gets closer to the tangential direction. Thus the rendered depth is biased obviously.

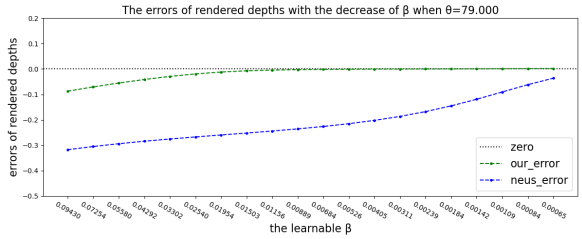
So it is necessary to transform SDF to density with an unbiased function. Following the above analysis, we scale $f(\mathbf{p}(t))$ with the inverse of the absolute value of $f'(\mathbf{p}(t))$,



(a) A ray across the sphere with a large angle.

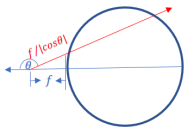


(b) Weights rendered by NeuS and Our model with a specific β.

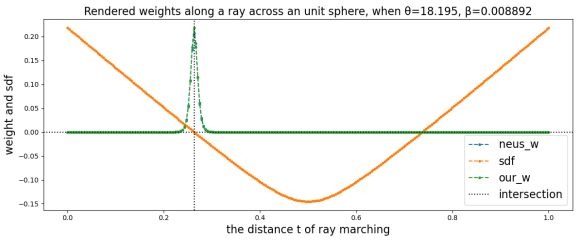


(c) Biases of $\hat{t}$ rendered by NeuS and Our model, with the descent of β.

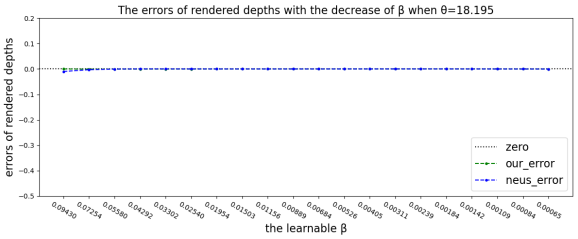
Figure 3. Simulation of SDF-based rendering when a ray goes across the sphere with a large angle.



(a) A ray across the sphere with a small angle.



(b) Weights rendered by NeuS and Our model with a specific β.



(c) Biases of $\hat{t}$ rendered by NeuS and Our model, with the descent of β.

Figure 4. Simulation of SDF-based rendering when a ray goes across the sphere with a small angle.

then take the scaled SDF into a Cumulative Distribution Function. It is formulated as

$$\sigma(\mathbf{p}(t)) = \alpha \Psi_\beta \left(\frac{-f\left(\mathbf{p}(t)\right)}{\left| f'\left(\mathbf{p}(t)\right) \right|} \right) \tag{9}$$

where $\alpha = \frac{1}{\beta}$ for the Logistic CDF, or $\alpha = \frac{2}{\beta}$ for the Laplace CDF. Here we focus on the Logistic CDF, and the result of the Laplace CDF is similar. The derivative of $w(\bar{t})$ respect to $\bar{t}$ at any intersection point is

$$\begin{aligned} \frac{\mathrm{d}w}{\mathrm{d}t}(\bar{t}) &= T(\bar{t}) \left(\alpha \Psi'_\beta(0) \left(\frac{-f(\mathbf{p}(\bar{t}))}{|f'(\mathbf{p}(\bar{t}))|} \right)' - \alpha^2 \Psi_\beta(0)^2 \right) \\ &= \frac{1}{4\beta^2} T(\bar{t}) \left(-\frac{f'(\mathbf{p}(\bar{t}))}{|f'(\mathbf{p}(\bar{t}))|} - 1 \right) \\ &= \begin{cases} 0, & if\ f'\left(\mathbf{p}(\bar{t})\right) < 0 \\ -\frac{1}{2\beta^2} T(\bar{t}), & if\ f'\left(\mathbf{p}(\bar{t})\right) > 0 \end{cases} \end{aligned} \tag{10}$$

It means that the derivative of $w(\bar{t})$ is equal to zero for a intersection point from outside to inside, and less than zero for a intersection point from inside to outside. Thus $w(\bar{t})$ always reaches local extremum at any intersection point from outside to inside.

We also validate the bias of rendered weight and depth of different methods for multiple planes and an unit sphere. As shown in Fig. 1 and Fig. 3 respectively, we sample a ray passing though multiple planes or an unit sphere at a direction which is close to tangent line, and compute $w(t)$ and $\hat{t}$ by different SDF-based rendering models. Then the gaps between rendered depth $\hat{t}$ and real depth t^* are measured with the decrease of β. As for ideal multiple planes in Fig. 1b, both of $w(t)$ rendered by our model and NeuS [24] reach maximum at t^*, but our $w(t^*)$ is much greater than $w((t^*)$ of NeuS [24]. The gaps between rendered depth $\hat{t}$ and real depth t^* are shown in Fig. 1c. It is seen that rendered depth $\hat{t}$ by NeuS [24] is inaccurate along a view direction which is close to tangent line. Compared with NeuS [24], the error of our $\hat{t}$ is very small.

Fig. 3 presents the cases of unit sphere for a large angle. As shown in Fig. 3b, $w(t)$ rendered by NeuS [24] is biased obviously, and the maximum is small too. Moreover, Fig. 3c indicates that its rendered depth $\hat{t}$ drifts even if β approaches zero, which is consistent with the above analysis for a locally convex surface. On the contrary, our $w(t)$ still reaches maximum at t^*, and the value is much greater. The gaps of our $\hat{t}$ between real depth t^* tends to be zero when β decreases.

The cases for small angles with respect to multiple planes and a unit sphere are shown in Fig. 2 and Fig. 4 respectively. It is seen that the weights rendered by our method and NeuS [24] are very close, and both reach maximum at t^*. The errors of all the $\hat{t}$ decrease quickly to zero with the descent of β.

3.3. Supervision with zero-level set

It is noted that the bias of rendered depth is still obvious when β is relatively large, and the rendered weight at the first intersection point is much less than 1 for a large β. The bias can not be erased at early stage if the model is trained only by color loss. Supervising the neural implicit surface directly with available geometry priors can reduce the bias effectively. There exist many excellent multi-view stereo works focused on depth prediction, such as MVSNet [29] and CasMVSNet [6]. Although these methods suffer from holes due to texture-less regions and are hard to obtain complete surfaces, they can provide cheap and reliable geometry priors for learning a neural implicit surface.

Since the rendered depth is biased at early training stage, it is not appropriate to supervise rendered depth with real depth directly. Alternatively, we map the predicted depths to dense point clouds at each view, then encourage SDF values of these points to be 0. In other words, the zero-level set of the neural implicit surface is supervised by dense point clouds at each view. The zero-level set loss of SDF network at each view is

$$\mathcal{L}_{SDF} = \frac{1}{N}\sum_{i=1}^{N} prob(\mathbf{p}_i)|f\left(\mathbf{p}_i\right) - 0|, \tag{11}$$

where $0 \leq prob(\mathbf{p}_i) \leq 1$ denotes the probability predicted by a certain MVS network, and it is set to 0 if the depth is invalid.

3.4. Optimization details

Loss function. Similar to previous works, we adopt L1 loss to minimize the error between rendered colors and ground truth colors:

$$\mathcal{L}_{rgb} = \frac{1}{N}\sum_{i}^{N} |\hat{C}_i - C_i|. \tag{12}$$

Here $\hat{C}_i$ is the rendered color along a certain ray, and C_i denotes the ground truth color.

Eikonal loss on the sampled points is also added to regularize the SDF field:

$$\mathcal{L}_{eik} = \frac{1}{MN}\sum_{i,j}^{MN} \|\nabla f(\mathbf{p}_{i,j}) - 1\|^2. \tag{13}$$

Finally the zero-level set loss of dense point clouds is added, and the total loss is defined as

$$\mathcal{L} = \mathcal{L}_{rgb} + \lambda\mathcal{L}_{eik} + \gamma\mathcal{L}_{SDF}. \tag{14}$$

We use $\lambda = 0.1$ and $\gamma = 1.0$ for all experiments.

Architecture. Our network is built based on NeuS [24]. The SDF network is encoded by 8-layers MLP with 256 hidden units. The view-dependent color network is encoded by 4-layers MLP with 256 hidden units. Since we focus on reconstruction without masks, NeRF++ [35] is also used to model the background. The frequencies of positional encoding for 3D position and viewing direction are 6 and 4 respectively.

Annealling Sampling. Valid point clouds are obtained by removing outliers of dense point clouds from CasMVSNet [6] with photometric filtering and geometric consistency filtering. We also introduce an annealling sampling strategy to balance the weights of pixels with valid point clouds and the weights of pixels without point clouds. At the beginning of the training process, we only sample pixels with the valid point clouds per view. The ratio of the valid pixels per batch is reduced with the growth of training steps, until it reaches the proportion of all valid pixels with points in an image. This simple strategy encourages our network to learn more from geometry priors at early stage, and pay more attention to other pixels without geometry priors at later stage. At the same time, We sample 3D points long a ray with the hierarchical sampling strategy used by NeuS [24].

4. Experiments

In this section, we provide quantitative and qualitative comparisons with state-of-the-art neural implicit surface approaches on the public object-centered datasets. Since we focus on reconstruction without masks, the approaches with mask supervision are not presented. Then we conduct an ablation study to evaluate the impact of different parts of our technical contributions.

4.1. Datasets

The DTU dataset [7] is widely used for evaluation of object-centered 3D reconstruction. There are 49 or 64 posed images with the resolution of 1600 × 1200 and scanned dense point clouds in each scene. Challenging cases are included in different scenes, such as specular reflection, texture-less regions and thin structures. Same as NeuS [24] and VolSDF [32], we use 15 scenes selected by IDR [33] to compare our method with others. Following the DTU evaluation pipeline, the reconstruction quality is measured with the chamfer distance, which is the average of accuracy and completeness.

4.2. Baselines

We mainly compare with state-of-the-art neural implicit surfaces methods without mask supervision: NeuS [24], VolSDF [32], NeuralWarp [2]. The results of MVSDF [34] are also presented, because it also uses a pre-trained depth estimation network. We also show the results of COLMAP [20] without mask supervision provided by NeuS [24].

ScanID	24	37	40	55	63	65	69	83	97	105	106	110	114	118	122	Mean
COLMAP	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
MVSDF	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	0.89
VolSdf	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	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
NeuS-12	0.93	1.07	0.81	0.38	1.02	0.60	0.58	1.43	1.15	0.78	0.57	1.16	0.35	0.45	0.46	0.78
NeuralWarp	**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
Our-50	0.56	0.92	0.39	0.39	0.85	0.58	**0.51**	1.20	**0.90**	0.78	**0.42**	0.84	**0.32**	0.43	0.44	0.64
Our	**0.49**	**0.71**	**0.37**	**0.36**	0.80	**0.56**	0.52	**1.17**	0.97	**0.66**	0.48	**0.73**	**0.32**	**0.42**	**0.42**	**0.60**

Table 1. Quantitative evaluation of chamfer distances on DTU. Most results come from original papers, except NeuS-12 which is cleaned by masks with a dilation of 12 pixels. The best results are indicated in **Bold**, while the second best results are underlined.

ScanID	24	37	40	55	63	65	69	83	97	105	106	110	114	118	122	Mean
VolSDF	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	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
Our	**28.95**	**27.43**	**28.30**	**28.90**	**33.12**	**34.09**	**31.09**	34.23	**29.92**	**33.29**	**35.25**	**32.93**	**31.75**	**37.20**	**37.21**	**32.24**

Table 2. Comparisons of PSNR for 2D view synthesis on DTU. Our method produces better PSNR than others.

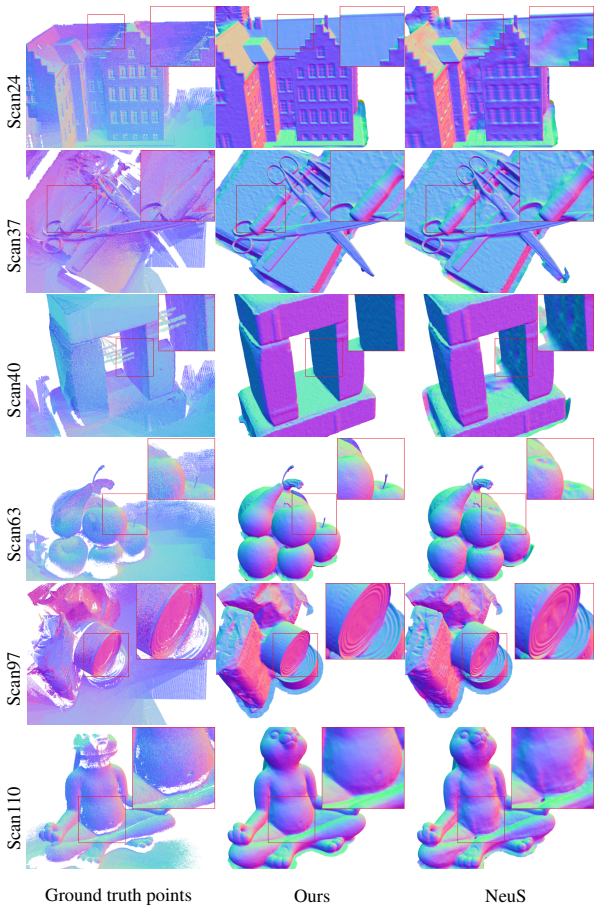


Figure 5. Rendered normal maps on DTU. The first column shows ground truth point clouds from DTU. The second and third ones show our method and NeuS, respectively

4.3. Implementation details

The region of interest is normalized as a unit sphere, centered at the object. 512 rays are sampled per batch with the hierarchical sampling strategy in NeuS [24]. Geometric initialization [1] and weight normalization are also used. As for the geometry priors, we choose CasMVSNet [6] as the pretrained MVS network, which can provide reliable depth maps and point clouds per view. In order to avoid training and testing on the same dataset, we select the checkpoint pretrained on BlendedMVS [31][1]. The depth prediction is fast and only takes some minutes for a single scene. The training of our SDF and color networks take around 9 hours on a single NVIDIA A30 GPU for 300k iterations per scene. After network training, we run Marching Cubes algorithm [11] to extract each mesh from the zero-level set of trained SDF network. Similar to previous work [2, 24, 32], the output meshes are cleaned with the dilated visibility masks.

4.4. Quantitative Comparisons

As shown in Table 1, we compare our method with several neural implicit surfaces methods on DTU. Most results of compared methods come from the original papers. Our method outperforms NeuS [24] and VolSDF [32] markedly. It also outperforms NeuralWarp [2], which uses an extra warping-based loss after training VolSDF [32] for 50k iterations with batches of 1024 pixels. Similar to our work, MVSDF [34] also uses a supervised depth estimation network to guide the SDF network, but it adopts the differentiable surface rendering provided by IDR [33] rather than SDF-based volume rendering. It should be noted that a custom filtering is used in MVSDF [34] while visual hull is used to clean meshes in others, so its results are not directly

[1] https://github.com/kwea123/CasMVSNet_pl

comparable. Moreover, we find that the radiuses for mask dilatation are different between NeuS [24], VolSDF [32] and NeuralWarp [2]: 50 pixels for NeuS [24] and VolSDF [32], 12 pixels for NeuralWarp [2]. For fare comparison, we also measure the meshes of our method with a dilation of 50 pixels, and the ones of NeuS [24] with a dilation of 12 pixels, denoted as Ours-50, NeuS-12 respectively. The dilation of 12 pixels is used in our final results. Our method still achieves better results with a dilation of 50 pixels. The evaluation of 2D view synthesis by PSNR on DTU is reported in Table 2. It is seen that our method outperforms NeuS [24] and VolSDF [32]. It is able to capture more texture details compared with the existing SDF-based volume rendering methods.

4.5. Qualitative Comparisons

We visually compare our method with NeuS [24] in Fig. 5 and Fig.6. Rendered normal maps are presented in Fig. 5. Columns represent the ground truth points, meshes of NeuS and Ours, respectively. Rows represent different scenes. As shown in Fig. 5, wrong concave surfaces occur in the results of NeuS, such as the rooftop of building in Scan24, the thin structure in Scan37, the inner wall in Scan40, the top area of the front apple in Scan63, the bottom and edge of the upper can in Scan97, and the abdominal region in Scan110. The stem of the apple on the right is lost. Compared with NeuS, the results of our full model is much better, and there are more high frequency details in our meshes. The rendered images are shown in Fig.6. Benefiting from our unbiased rendering model and geometry priors, the rendering quality is also improved. More texture details are captured.

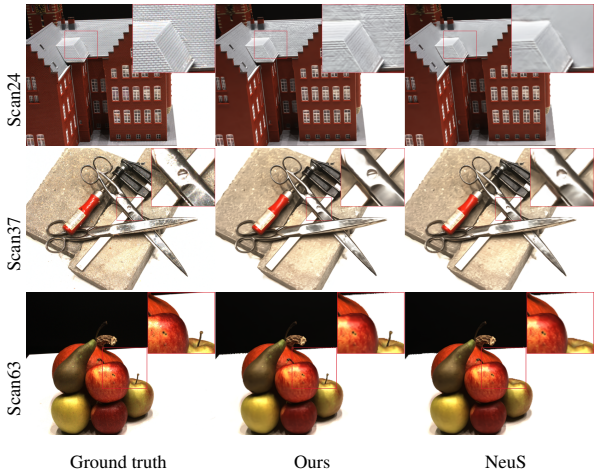


Figure 6. Rendered 2D images on DTU. The first column shows ground truth 2D views from DTU. The second and third columns show our method and NeuS.

4.6. Ablation study

To evaluate different parts of our technical contributions, an ablation study is conducted on the DTU dataset. We take NeuS [24] as our baseline, and test the effect of unbiased rendering and geometry priors independently. As shown in the row "w/o sdf" of Table 3, our method outperforms NeuS [24] markedly even if the SDF loss is not used. It also gets comparable performance against NeuralWarp [2]. As shown in the row "w/o unbaised", the supervision of geometry priors is powerful, and it reduces the error of reconstruction remarkably. Finally we combine all of them to achieve the best performance, as shown in the last row.

Method	Chamfer distance
NeuS	0.84
NeuS-12	0.78
Our(w/o sdf)	0.71
Our(w/o unbaised)	0.65
Full model	0.60

Table 3. Ablation study on DTU. "w/o sdf" denotes our unbiased rendering without geometry priors. "w/o unbaised" denotes NeuS with the same geometry priors.

5. Conclusions

In this paper, we analyze the bias of existing SDF-based volume rendering strategies, and provide an additional condition for the unbiased SDF-based volume rendering: The rendered depth should be equal to the distance between the first intersection point and the camera center along the ray. In order to reduce the bias, we introduce a novel transformation from the SDF field to the density field. We scale the SDF field with the cosine of the angle between the viewing direction and the surface normal vector, then the scaled SDF field is combined with a certain CDF to model the density field. Validations on toy data indicate that the bias of rendered depth is reduced. Moreover, we find that the bias can not be removed fully for a large deviation of SDF at early training stage. Thus we supervise the SDF with cloud points obtained from a pre-trained MVS network. Experiments on DTU benchmark show our model outperforms the recent neural implicit surface methods.

Limitations. There are still several limitations in our method. First, it is time-consuming to reconstruct a certain object with high resolution. A promising solution is to speed up the convergence with recent strategies proposed by Instant-NGP [15] and Plenoxels [3]. Second, it is interesting to extend our model to capture 3D dynamic scenes. Third, it is worthy of studying to train general models across various scenes. Finally, it is challenging to recover the correct geometry for complex objects with obvious specular or translucent materials in the wild.

References

[1] 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. 7

[2] 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. 2, 3, 6, 7, 8

[3] Sara Fridovich-Keil, Alex Yu, Matthew Tancik, Qinhong Chen, Benjamin Recht, and Angjoo Kanazawa. Plenoxels: Radiance fields without neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5501–5510, 2022. 2, 8

[4] Qiancheng Fu, Qingshan Xu, Yew-Soon Ong, and Wenbing Tao. Geo-neus: geometry-consistent neural implicit surfaces learning for multi-view reconstruction. *arXiv preprint arXiv:2205.15848*, 2022. 3

[5] Silvano Galliani, Katrin Lasinger, and Konrad Schindler. Massively parallel multiview stereopsis by surface normal diffusion. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 873–881, 2015. 1, 2

[6] Xiaodong Gu, Zhiwen Fan, Siyu Zhu, Zuozhuo Dai, Feitong Tan, and Ping Tan. Cascade cost volume for high-resolution multi-view stereo and stereo matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2495–2504, 2020. 2, 6, 7

[7] Rasmus Jensen, Anders Dahl, George Vogiatzis, Engin Tola, and Henrik Aanæs. Large scale multi-view stereopsis evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 406–413, 2014. 2, 6

[8] Michael Kazhdan and Hugues Hoppe. Screened poisson surface reconstruction. *ACM Transactions on Graphics (ToG)*, 32(3):1–13, 2013. 1, 2

[9] 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. 2

[10] Hai Li, Xingrui Yang, Hongjia Zhai, Yuqian Liu, Hujun Bao, and Guofeng Zhang. Vox-surf: Voxel-based implicit surface representation. *IEEE Transactions on Visualization and Computer Graphics*, 2022. 3

[11] 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. 7

[12] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4460–4470, 2019. 2

[13] 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 *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I*, pages 405–421, 2020. 1, 2

[14] Pierre Moulon, Pascal Monasse, Romuald Perrot, and Renaud Marlet. Openmvg: Open multiple view geometry. In *International Workshop on Reproducible Research in Pattern Recognition*, pages 60–74. Springer, 2016. 1, 2

[15] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *arXiv preprint arXiv:2201.05989*, 2022. 2, 8

[16] 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. 1, 2

[17] 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. 1, 2

[18] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 165–174, 2019. 1, 2

[19] 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. 2

[20] Johannes L Schönberger, Enliang Zheng, Jan-Michael Frahm, and Marc Pollefeys. Pixelwise view selection for unstructured multi-view stereo. In *European conference on computer vision*, pages 501–518. Springer, 2016. 1, 2, 6

[21] Jiaming Sun, Xi Chen, Qianqian Wang, Zhengqi Li, Hadar Averbuch-Elor, Xiaowei Zhou, and Noah Snavely. Neural 3d reconstruction in the wild. In *ACM SIGGRAPH 2022 Conference Proceedings*, pages 1–9, 2022. 2

[22] Delio Vicini, Sébastien Speierer, and Wenzel Jakob. Differentiable signed distance function rendering. *ACM Transactions on Graphics (TOG)*, 41(4):1–18, 2022. 2

[23] Fangjinhua Wang, Silvano Galliani, Christoph Vogel, Pablo Speciale, and Marc Pollefeys. Patchmatchnet: Learned multi-view patchmatch stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14194–14203, 2021. 2

[24] 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. *Advances in Neural Information Processing Systems*, 34:27171–27183, 2021. 1, 2, 3, 4, 5, 6, 7, 8

[25] Yiqun Wang, Ivan Skorokhodov, and Peter Wonka. Improved surface reconstruction using high-frequency details. *arXiv preprint arXiv:2206.07850*, 2022. 3

[26] Tong Wu, Jiaqi Wang, Xingang Pan, Xudong Xu, Christian Theobalt, Ziwei Liu, and Dahua Lin. Voxurf: Voxel-based efficient and accurate neural surface reconstruction. *arXiv preprint arXiv:2208.12697*, 2022. 3

[27] Qingshan Xu and Wenbing Tao. Multi-scale geometric consistency guided multi-view stereo. In *Proceedings of*

the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 5483–5492, 2019. 1, 2

[28] Jiayu Yang, Jose M Alvarez, and Miaomiao Liu. Non-parametric depth distribution modelling based depth inference for multi-view stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8626–8634, 2022. 2

[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. 2, 6

[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. 2

[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. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1790–1799, 2020. 2, 7

[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. 1, 2, 3, 4, 6, 7, 8

[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*, 33:2492–2502, 2020. 1, 2, 6, 7

[34] Jingyang Zhang, Yao Yao, and Long Quan. Learning signed distance field for multi-view surface reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6525–6534, 2021. 2, 6, 7

[35] Kai Zhang, Gernot Riegler, Noah Snavely, and Vladlen Koltun. Nerf++: Analyzing and improving neural radiance fields. *arXiv preprint arXiv:2010.07492*, 2020. 1, 2, 6

[36] Enliang Zheng, Enrique Dunn, Vladimir Jojic, and Jan-Michael Frahm. Patchmatch based joint view selection and depthmap estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1510–1517, 2014. 1, 2