NeAT: Learning Neural Implicit Surfaces with Arbitrary Topologies from Multi-view Images

Xiaoxu Meng Weikai Chen Bo Yang
Digital Content Technology Center, Tencent Games
{xiaoxumeng,weikaichen,brandonyang}@global.tencent.com

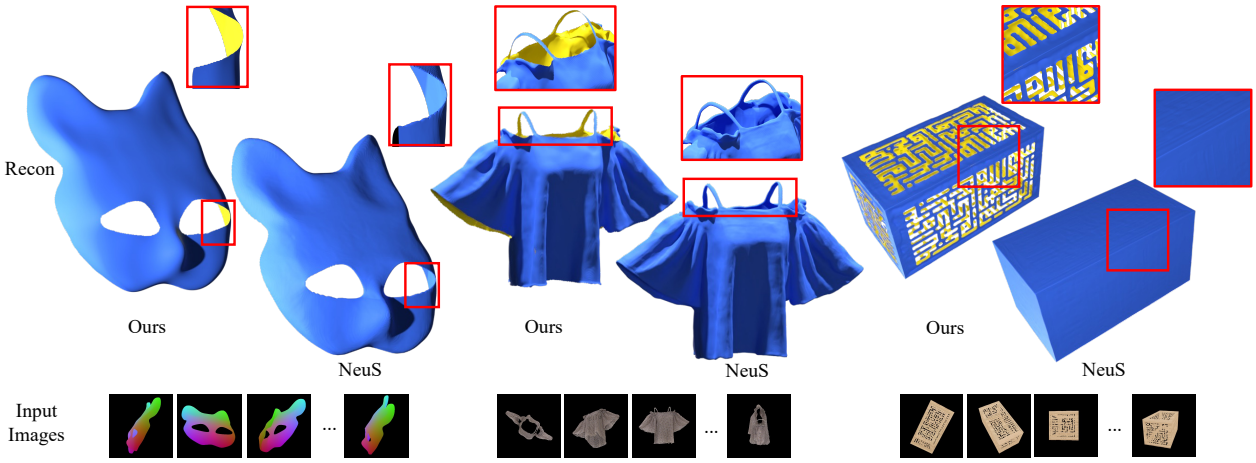

Figure 1. We show three groups of surface reconstruction from multi-view images. The front and back faces are rendered in blue and yellow respectively. Our method (left) is able to reconstruct high-fidelity and intricate surfaces of arbitrary topologies, including those non-watertight structures, e.g. the thin single-layer shoulder strap of the top (middle). In comparison, the state-of-the-art NeuS [48] method (right) can only generate watertight surfaces, resulting in thick, double-layer geometries.

Abstract

Recent progress in neural implicit functions has set new state-of-the-art in reconstructing high-fidelity 3D shapes from a collection of images. However, these approaches are limited to closed surfaces as they require the surface to be represented by a signed distance field. In this paper, we propose NeAT, a new neural rendering framework that can learn implicit surfaces with arbitrary topologies from multi-view images. In particular, NeAT represents the 3D surface as a level set of a signed distance function (SDF) with a validity branch for estimating the surface existence probability at the query positions. We also develop a novel neural volume rendering method, which uses SDF and validity to calculate the volume opacity and avoids rendering points with low validity. NeAT supports easy field-to-mesh conversion using the classic Marching Cubes algorithm. Extensive experiments on DTU [20], MGN [4], and Deep Fashion 3D [19] datasets indicate that our approach is able to faithfully reconstruct both watertight and non-watertight surfaces. In particular, NeAT significantly outperforms the state-of-the-art methods in the task of open surface reconstruction both quantitatively and qualitatively.

1. Introduction

3D reconstruction from multi-view images is a fundamental problem in computer vision and computer graphics. Recent advances in neural implicit functions [26, 36, 48, 55] have brought impressive progress in achieving high-fidelity reconstruction of complex geometry even with sparse views. They use differentiable rendering to render the inferred implicit surface into images which are compared with the input images for network supervision. This provides a promising alternative of learning 3D shapes directly from 2D images without 3D data. However, existing neural rendering methods represent surfaces as signed distance function (SDF) [27, 55] or occupancy field [36, 38], limiting their output to *closed* surfaces. This leads to a barrier in reconstructing a large variety of real-world objects with open boundaries, such as 3D garments, walls of a scanned 3D scene, *etc*. The recently proposed NDF [10], 3PSDF [6]

and GIFS [56] introduce new implicit representations supporting 3D geometry with arbitrary topologies, including both closed and open surfaces. However, none of these representations are compatible with existing neural rendering frameworks. Leveraging neural implicit rendering to reconstruct *non-watertight* shapes, i.e., shapes with *open* surfaces, from multi-view images remains a virgin land.

We fill this gap by presenting *NeAT*, a *Ne*ural rendering framework that reconstructs surfaces with *A*rbitrary *T*opologies using multi-view supervision. Unlike previous neural rendering frameworks only using color and SDF predictions, we propose a validity branch to estimate the surface existence probability at the query positions, thus avoiding rendering 3D points with low validity as shown in Figure 2. In contrast to 3PSDF [6] and GIFS [56], our validity estimation is a differentiable process. It is compatible with the volume rendering framework while maintaining its flexibility in representing arbitrary 3D topologies. To correctly render both closed and open surfaces, we introduce a sign adjustment scheme to render both sides of surfaces, while maintaining unbiased weights and occlusion-aware properties as previous volume renderers. In addition, to reconstruct intricate geometry, a specially tailored regularization mechanism is proposed to promote the formation of open surfaces. By minimizing the difference between the rendered and the ground-truth pixels, we can faithfully reconstruct both the validity and SDF field from the input images. At reconstruction time, the predicted validity value along with the SDF value can be readily converted to 3D mesh with the classic field-to-mesh conversion techniques, e.g., the Marching Cubes Algorithm [30].

We evaluate NeAT in the task of multi-view reconstruction on a large variety of challenging shapes, including both closed and open surfaces. NeAT can consistently outperform the current state-of-the-art methods both qualitatively and quantitatively. We also show that NeAT can provide efficient supervision for learning complex shape priors that can be used for reconstructing non-watertight surface only from a single image. Our contributions can be summarized as:

- A neat neural rendering scheme of implicit surface, coded *NeAT*, that introduces a novel validity branch, and, *for the first time*, can faithfully reconstruct surfaces with arbitrary topologies from multi-view images.
- A specially tailored learning paradigm for NeAT with effective regularization for open surfaces.
- NeAT sets the new state-of-the-art on multi-view reconstruction on open surfaces across a wide range of benchmarks.

2. Related Work

3D Geometric Representation A 3D surface can be represented explicitly with voxels [5, 11, 23, 32, 44], point clouds [1, 12, 24, 31, 53], and meshes [16, 47, 50], or can be represented implicitly with neural implicit functions, which have gained popularity for their continuity and the arbitrary-resolution property. Watertight surfaces could be represented by occupancy functions [9, 33, 41], signed distance functions [34, 39, 52], or other signed implicits [2, 13]. These approaches are limited to closed surfaces as they require the space to be represented as "inside" and "outside." To lift the limitation, unsigned distance function (UDF) [10, 45, 46] is proposed to represent a much broader class of shapes containing open surfaces. However, the signless property of UDF prevents itself from applying the classic field-to-mesh conversion techniques [30]. Instead, these UDF approaches support exporting open surfaces by applying the Ball-Pivoting algorithm [3], meshUDF [17], and Neural Dual Contouring [8], which are prone to disconnected surface patches with inconsistent normals and coherence artifacts. GIFS [56] represents non-watertight shapes by encoding whether two points are separated by any surface instead of dividing a 3D space into predefined inside/outside regions. Three-pole signed distance function (3PSDF) [6] introduces the *null* sign in addition to the conventional *in* and *out* labels. The *null* sign stops the formation of closed isosurfaces, thus enabling the representation of both watertight- and open-surfaces. However, 3PSDF [6] and GIFS [56] model the reconstruction of open surfaces as a classification problem, thus preventing these implicit representations from being differentiable. As a result, these methods do not support differentiable downstream tasks like differentiable rendering. Inspired by the *null* sign of 3PSDF, we propose to represent an open surface as a signed distance function with a validity branch to estimate the surface existence probability at the query positions, which bypasses its limitation of non-differentiability while keeping its capability of modeling arbitrary shapes.

Implicit Surfaces Reconstruction from Multi-view Images It is well-known that a 3D database is more challenging to acquire than a 2D database. As such, learning shapes from 2D supervision is important and necessary. Multiple differentiable rendering (DR) techniques have been proposed to circumvent the difficulty of explicit correspondence matching in 3D reconstruction. Two popular types of DR are differentiable rasterization [7, 14, 22, 25] and differentiable ray casting. A popular branch of differentiable ray casting is surface rendering [21, 26, 27, 37, 55], which assumes that the ray's color only relies on the color of the intersection point. Surface rendering methods represent the geometry as an implicit function and learn the surface representation from 2D images via differentiable rendering techniques. NeRF [35] and follow-up volume rendering methods assume that the ray's color relies on all the sampled points along the ray. UNISURF [38] improves the reconstruction quality by shrinking the sample region of volume rendering during optimization. NeuS [48], VolSDF [54] and HF-NeuS [49]

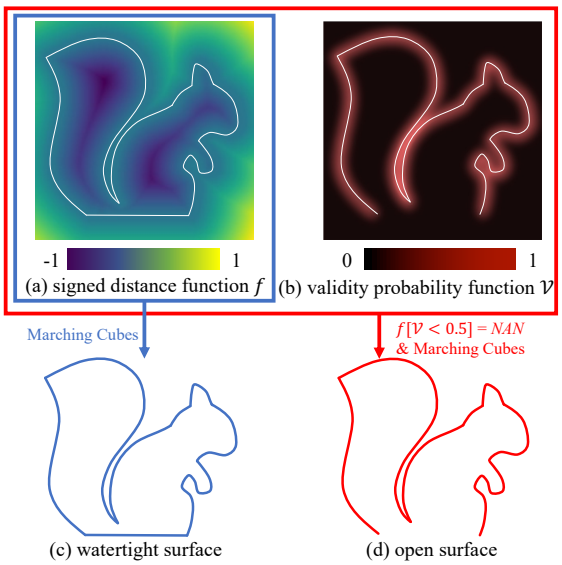

Figure 2. (a) is the signed distance function (SDF); (b) is the validity probability function $\mathcal{V}$; (c) is the watertight surface extracted from (a) SDF; (d) is the open surface extracted from (a) SDF and (b) validity probability. In our mesh extraction process, we set the SDF of the 3D query points with low validity (here $\mathcal{V} < 0.5$) to *NAN* and extract the open surface with the Marching Cubes algorithm.

develop volume density functions for watertight surfaces applied to SDFs, which combine the advantages of surface rendering-based and volume rendering-based methods. However, as the above methods all rely on the representation of the signed distance function or occupancy field, they can only reconstruct watertight shapes. NeuralUDF [29] and NeUDF [28], two concurrent works, propose to represent arbitrary surfaces as a UDF and develop unbiased density functions that correlate the property of UDF with the volume rendering scheme. However, converting UDF to a mesh typically suffer from artifacts, inconsistent normals, and large computational costs. Compared with the UDF-based approaches, our method represents the scene by a signed distance function with a validity branch, and thus is compatible with easy field-to-mesh conversion methods, such as the classic Marching Cubes algorithm, ensuring high-fidelity and normal-consistent meshing results from the implicit field.

3. Volumetric Rendering with NeAT

Our representation is able to reconstruct 3D surfaces with arbitrary topologies without 3D ground-truth data for training. As shown in Figure 2, by taking the SDF (Figure 2 (a)) and validity probability (Figure 2 (b)) into consideration together, we acquire additional information that the bottom line in Figure 2 (a) is invalid. We discard parts of the reconstructed surface according to the validity score and extract an open surface as shown in Figure 2 (d) with the Marching Cubes algorithm [30].

3.1. Formulation

Given N images $I_{gt}(k)_{k=1}^{N}$ with a resolution of $(W,\ H)$ together with corresponding camera intrinsics, extrinsics, and object masks $M_{gt}(k)_{k=1}^{N}$, our goal is to reconstruct the surface of the object. The framework of our method is shown in Figure 3. Given a sampled pixel $\mathbf{o}$ on an input image, we project it to the 3D space and get the sampled 3D points on the ray emitting from the pixel as $\{\mathbf{p}\,(t) = \mathbf{o} + t\mathbf{v} \mid t \geq 0\}$, where $\mathbf{o}$ is the center of the camera and $\mathbf{v}$ is the unit direction vector of the ray. Then, we predict the signed distance value $f(\mathbf{p}(t))$, validity probability $\mathcal{V}(\mathbf{p}(t))$, and the RGB value $c(\mathbf{p}(t))$ of the points by our fully connected neural networks called NeAT-Net. Specifically, NeAT-Net includes:

- SDF-Net: a mapping function $f(\cdot) : \mathbf{R}^3 \rightarrow \mathbf{R}$ to represent the signed distance field.
- Validity-Net: a mapping function $\mathcal{V}(\cdot) : \mathbf{R}^3 \rightarrow \mathbf{R}$ to represent the validity probability;
- Color-Net: a mapping function $c(\cdot) : \mathbf{R}^3 \rightarrow \mathbf{R}^3$ to predict the per-point color of the 3D space.

The outputs of the three networks are delivered to our novel NeAT renderer to render images and masks from the implicit representations. Our renderer supports both open and closed surfaces, and therefore it provides the capability of reconstructing arbitrary shapes.

The predicted mask M_{pred} could be inferred from the rendering weights $w(\mathbf{p}(t))$ for each sampling point, and the predicted image I_{pred} could be calculated from the RGB $c(\mathbf{p}(t))$ and the rendering weights $w(\mathbf{p}(t))$:

$$\begin{aligned} M_{pred}(\mathbf{o}, \mathbf{v}) &= \int_0^{+\infty} w(\mathbf{p}(t))dt, \\ I_{pred}(\mathbf{o}, \mathbf{v}) &= \int_0^{+\infty} w(\mathbf{p}(t))c(\mathbf{p}(t)))dt. \end{aligned} \tag{1}$$

The predicted masks and images are used for loss calculation during training, which will be illustrated in Section 3.3. After training is completed, we go through the testing module as shown in Figure 3; we set the SDFs to NAN for 3D points with $\mathcal{V}(\mathbf{p})$ less than 0.5, and feed them to Marching Cubes algorithm to produce the final mesh.

3.2. Construction of NeAT Renderer

According to Equation 1, one key issue in the rendering process is to find an appropriate weight function $w(\mathbf{p}(t))$. We split this task into two steps: 1) building a probability density function to estimate volume density from SDF; 2) estimating the weight function $w(\mathbf{p}(t))$ from the volume density and the validity probability.

Construction of Probability Density Function. Due to aiming at building arbitrary surfaces, we first introduce the difference between rendering watertight and open surfaces.

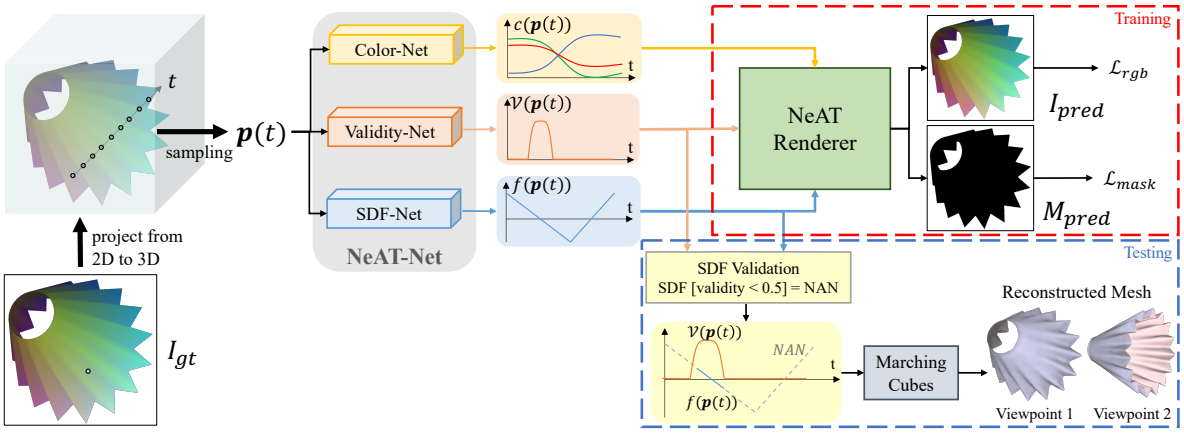

Figure 3. The framework of our approach. We project a sampled pixel on the input image I_{gt} to 3D to get the sampled 3D points $\mathbf{p}(t)$ on a ray. Next, the SDF-Net, Validity-Net, and Color-Net take $\mathbf{p}(t)$ as the input to predict the signed distance $f(\mathbf{p}(t))$, validity probability $\mathcal{V}(\mathbf{p}(t))$, and the RGB $c(\mathbf{p}(t))$, respectively. Then our NeAT renderer generates the color I_{pred} and the mask M_{pred} for NeAT-Net optimization. In the mesh exportation (testing) stage, we update the SDF by assigning the low-validity points with a nan value, thus preventing the decision boundary from forming at those regions. Finally, we export arbitrary surfaces from the updated SDF with the Marching Cubes Algorithm.

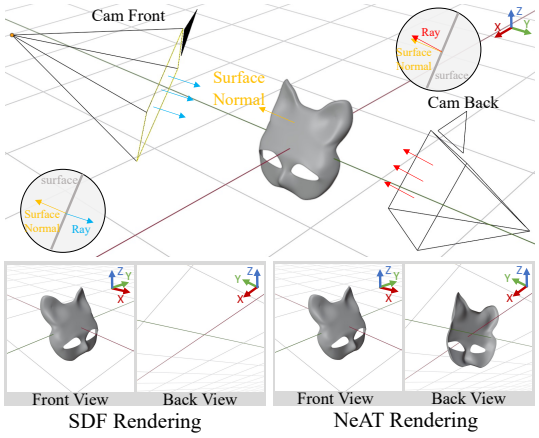

Figure 4. The SDF rendering scheme only renders the surfaces when the ray enters the surface from outside to inside. For an open surface whose surface normal aligns with the back camera's viewing direction, the back camera receives an empty rendering. Our NeAT rendering scheme renders both sides of the surfaces.

The first difference is the rendering of back-faces. The state-of-the-art watertight surface reconstruction approaches [37, 48, 55] only render the front faces of the surface and ignores the back faces. Such a scheme would fail for open surfaces: as shown in Figure 4 (L), the back camera receives an empty rendering of the open surface. While, we render each surface point with ray intersections, as shown in Figure 4 (R).

The second difference is the definition of "inside" and "outside", which do not exist for non-watertight surfaces. Therefore, we leverage the local surface normal to determine the sign of the distance as in 3PSDF [6]. For a local region around a surface, we use positive normal direction as pseudo "outside" with positive-signed distance, and vice versa.

We expect that the rendering behaves the same when the ray crosses a surface from either direction. The state-of-the-art volume rendering work, NeuS [48], uses logistic density distribution $\phi_s(f(\mathbf{p}))$, also known as the derivative of the sigmoid mapping function $\Phi_\mathbf{s}(f(\mathbf{p}))$, as the probability density function. However, it is not applicable in our scenario – for surfaces with opposite normal directions, $\Phi_s(f(\mathbf{p}))$ will lead to different density values as $\Phi_s(f(\mathbf{p})) \neq \Phi_s(-f(\mathbf{p}))$.

We therefore modify the SDF value by flipping its sign in the regions where the SDF value increases along the camera ray. The probability density function is defined as

$$\sigma(\mathbf{p}) = \phi_s(-Sign(\mathbf{v}\cdot\mathbf{n})f(\mathbf{p})), \tag{2}$$

where $\mathbf{v}$ is the unit direction vector of the ray and $\mathbf{n}$ is the gradient of the signed distance function. Such definition assures the same rendering behaviors when ray enters the surface from either direction.

Construction of Opaque Density Function. According to NeuS [48], the weight function $w(\mathbf{p}(t))$ should have two properties: unbiased and occlusion-aware. Similarly, we define unbiased rendering weight $w(\mathbf{p}(t))$ with Equation 3 and define an occlusion-aware weight function based on the opaque density $\rho(t)$ with Equation 4.

$$\begin{cases} w(\mathbf{p}(t)) = \dfrac{\phi_s(-Sign(\mathbf{v}\cdot\mathbf{n})f(\mathbf{p}(t)))}{\int_{-\infty}^{+\infty}\phi_s(-Sign(\mathbf{v}\cdot\mathbf{n})f(\mathbf{p}(t)))} & (3) \\ w(\mathbf{p}(t)) = \exp(-\int_0^t \rho(u)du)\rho(t) & (4) \end{cases}$$

Solving Equation 3 and Equation 4, we get

$$\rho(t) = \frac{-\frac{d\Phi_s}{dt}(-Sign(\mathbf{v}\cdot\mathbf{n})f(\mathbf{p}(t)))}{\Phi_s(-Sign(\mathbf{v}\cdot\mathbf{n})f(\mathbf{p}(t)))} \tag{5}$$

Please checkout the supplemental for the derivation.

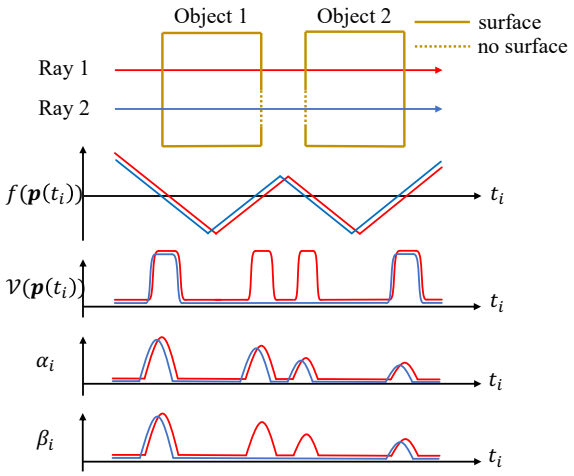

Figure 5. Illustration of the rendering with validity probability.

Discretization. We adopt the classic discretization scheme in differentiable volumetric rendering [35,48] for the opacity and weight function. For a set of sampled points along the ray $\{p_i = \mathbf{o}+t_i\mathbf{v}|i = 1, ..., n, t_i < t_{i+1}\}$, the rendered pixel color is

$$I_{pred}(\mathbf{o}, \mathbf{v}) = \sum_{i=1}^{n} \Pi_{j=1}^{i-1}(1 - \alpha_j)\alpha_i c_i \tag{6}$$

where c_i is the estimated color for the i-th sampling point; α_i is the discrete opacity value in SDF rendering

$$\alpha_i = \frac{\Phi_s(-Sign(\mathbf{v}\cdot\mathbf{n})f(\mathbf{p}(t_i))) - \Phi_s(-Sign(\mathbf{v}\cdot\mathbf{n})f(\mathbf{p}(t_{i+1})))}{\Phi_s(-Sign(\mathbf{v}\cdot\mathbf{n})f(\mathbf{p}(t_i)))} \tag{7}$$

Now we have built an unbiased and occlusion-aware volume weight function that supports rendering the front and back faces with the SDF representation.

Rendering with Validity Probability. To render both closed and open surfaces, we multiply the validity probability of the 3D query points to their opacity value in the rendering process. The discrete opacity value β_i of the i-th sampled point is

$$\beta_i = \alpha_i \cdot \mathcal{V}(\mathbf{p}(t_i)) \tag{8}$$

We show a 2D illustration of rendering two objects with open boundaries in Figure 5. Ray 2 only has two intersections with the objects due to the existences of open gaps (marked as dotted lines). Ray 1 and Ray 2 share the same SDF $f(\mathbf{p}(t_i))$ and discrete opacity value α_i. However, according to the validity branch $\mathcal{V}(\mathbf{p}(t_i))$, Ray 1 has four valid regions while Ray 2 only has two. By considering the validity probabilities, the discrete opacity value β_i of the gaps in Ray 2 are set to zero, avoiding generating false surfaces in reconstruction.

Therefore, the final rendered pixel color of a surface is

$$I_{pred}(\mathbf{o}, \mathbf{v}) = \sum_{i=1}^{n} \Pi_{j=1}^{i-1}(1 - \beta_j)\beta_i c_i \tag{9}$$

3.3. Training

We supervise the training of NeAT-Net with five losses. The first three are **RGB Loss**, **Mask Loss**, and **Eikonal Loss**, the same as used in previous neural rendering works [48, 55]. They are defined as

$$\mathcal{L}_{rgb} = \sum_{i,j} ||I_{pred}(i,j) - I_{gt}(i,j)|| \cdot M_{gt}(i,j) \tag{10}$$

$$\mathcal{L}_{mask} = \sum_{i,j} BCE(M_{pred}(i,j), M_{gt}(i,j)) \tag{11}$$

$$\mathcal{L}_{eikonal} = \frac{1}{N}\sum_{\mathbf{p}}(|\frac{\partial f(\mathbf{p})}{\partial \mathbf{p}}| - 1)^2 \tag{12}$$

where BCE is the binary cross entropy.

Rendering Probability Loss In the physical world, the existence of surfaces is binary (exist/not exist). As a result, the validity probability of a 3D sampling point is either 0 (with no surface) or 1 (with surface). We therefore add the binary cross entropy of $\mathcal{V}(\mathbf{p})$ as an extra regularization:

$$\mathcal{L}_{bce} = \frac{1}{N}\sum_{\mathbf{p}} BCE(\mathcal{V}(\mathbf{p}), \mathcal{V}(\mathbf{p})). \tag{13}$$

Rendering Probability Regularization For real-world objects with open structures, the surfaces are sparsely distributed in the 3D space. To prevent NeAT-Net from predicting redundant surfaces, we introduce a sparsity loss to promote the formation of open surfaces:

$$\mathcal{L}_{sparse} = \frac{1}{N}\sum_{\mathbf{p}} \mathcal{V}(\mathbf{p}). \tag{14}$$

We optimize the following loss function

$$\begin{aligned}\mathcal{L} = \mathcal{L}_{rgb} + \lambda_{mask}\cdot\mathcal{L}_{mask} + \lambda_{eikonal}\cdot\mathcal{L}_{eikonal} \\ + \lambda_{bce}\cdot\mathcal{L}_{bce} + \lambda_{sparse}\cdot\mathcal{L}_{sparse}.\end{aligned} \tag{15}$$

4. Experiments

4.1. Experiment Setup

Tasks and Datasets. We validate NeAT using three types of experiments. We first conduct multi-view reconstruction for real-world watertight objects to ensure that NeAT achieves comparable reconstruction quality on watertight surfaces. We conduct this experiment on 10 scenes from the *DTU Dataset* [20]. Each scene contains 49 or 64 RGB images and masks with a resolution of 1600×1200. Second, we reconstruct open surfaces from multi-view images. We run this experiment on eight categories from the Deep Fashion 3D Dataset [19] and five categories from the Multi-Garment Net Dataset [4], which contain clothes with a wide variety of materials, appearance, and geometry, including challenging cases for reconstruction algorithms, such as camisoles. Finally, we construct an autoencoder, which takes a single image as the input and provides validation on the challenging task of single-view reconstruction on open

(a) Ref Images (b) Ours (c) NeuS (d) IDR (e) HFS (f) COLMAP

Figure 6. Comparisons on watertight surface reconstruction.

surfaces. We conduct this experiment on the *dress* category from the Deep Fashion 3D Dataset [19]. We randomly select 116 objects as the training set and 25 objects as the test set. All experiments are compared with the SOTA methods for better verification. To avoid thin closed reconstructions during the training process, we employ a smaller learning rate for the SDF-Net and a larger learning rate for the Validity-Net. Please refer to the implementation of NeAT-Net in the supplementary.

Implementation details. For the reconstruction experiments on open surfaces, we render the ground truth point clouds from Deep Fashion 3D Dataset [19] with Pytorch3D [40] at a resolution of 256^2. To get diverse supervision data, we uniformly sample 648 and 64 viewpoints on the unit sphere for Deep Fashion 3D Dataset and Multi-Garment Net Dataset (MGN), respectively. For the single view reconstruction experiment, we uniformly sample 64 viewpoints on the unit sphere as the camera positions. We use an ResNet-18 [18] encoder to predict a latent code $\mathbf{z}$ describing the surface's geometry and color. We then use the concatenation of $\{\mathbf{z}, \mathbf{p}\}$ as the input to NeAT-Net (decoder) to evaluate the SDF, validity, and color at the query positions. We optimize the autoencoder by comparing the 2D rendering and the ground truth image. In the evaluation stage, we accept a single image as the input and directly export the evaluated SDF and validity as 3D mesh.

Evaluations. For multiview reconstruction on watertight surfaces, we measure the Chamfer Distance (CD) with *DTU MVS 2014 evaluation toolkit* [20]. For the reconstruction experiments on open surfaces, we measure the CD with the *PCU* Library [51]. For all the experiments, we evaluate the result meshes at resolution 512^3.

4.2. Multiview Reconstruction on Closed Surfaces

We compare our approach with the state-of-the-art volume and surface rendering based methods - HFS [49], NeuS [48] and IDR [55], and a classic mesh reconstruction and novel view synthesis method – NeRF [35]. We report the quantitative results in Table 1.

We also show visual comparison with a widely-used MVS method: COLMAP [42, 43]. We show qualitative results in Fig. 6. The results reconstructed with the proposed method show comparable quality compared with the state-of-the-art.

CD↓	Ours	NeuS	IDR	NeRF	HFS
scan 55	0.47	0.38	0.48	0.66	**0.37**
scan 69	0.84	**0.60**	0.77	1.50	0.66
scan 83	1.28	1.43	1.33	**1.20**	1.27
scan 97	1.09	**0.96**	1.16	1.96	1.00
scan 105	**0.75**	0.78	0.76	1.27	0.86
scan 106	0.76	**0.52**	0.67	0.66	0.57
scan 110	**0.80**	1.44	0.90	2.61	1.24
scan 114	0.38	**0.36**	0.42	1.04	0.41
scan 118	0.56	**0.46**	0.51	1.13	0.52
scan 122	0.55	**0.49**	0.53	0.99	**0.49**
average	0.749	0.742	0.753	1.302	**0.741**

Table 1. Quantitative evals on real-world object reconstruction.

(a) GT (b) Ours (c) NeuS (d) IDR (e) HFS

Figure 7. Comparisons on open surface reconstruction. Row 1 – 4 are evaluated on Deep Fashion 3D Dataset [19] and Row 5 is evaluated on Multi-Garment Net Dataset [4]. NeAT is able to reconstruct high-fidelity open surfaces while NeuS [48], IDR [55] and HFS [49] fail to recover the correct topologies.

	CD ($\times 10^{-3}$) $\downarrow$	Ours	NeuS	IDR	HFS
D3D	long slv upper	**4.483**	6.864	11.494	9.695
	short slv upper	**4.517**	6.048	9.043	7.800
	no slv upper	**3.418**	4.856	17.710	8.576
	long slv dress	**4.843**	6.135	9.203	8.235
	short slv dress	**4.276**	7.951	8.506	7.705
	no slv dress	**3.706**	5.406	6.785	7.565
	pants	**5.391**	11.847	10.880	16.205
	dress	**3.889**	5.673	6.983	11.644
	average	**4.315**	6.847	10.075	9.678
MGN	LongCoat	**7.601**	8.038	12.058	10.398
	TShirtNoCoat	**8.481**	9.910	15.709	13.128
	ShirtNoCoat	**5.281**	8.084	9.509	11.299
	ShortPants	**15.324**	15.480	16.329	18.332
	Pants	**9.191**	12.188	19.931	19.414
	average	**9.176**	10.740	14.707	14.514

Table 2. Quantitative evaluation on *Deep Fashion 3D Dataset* (D3D) [19] with chamfer distance averaged over five examples per category, and Multi-Garment Net Dataset (MGN) [4] with chamfer distance averaged on two examples per category.

4.3. Multiview Reconstruction on Open Surfaces

We conduct this experiment on eight categories from Deep Fashion 3D [19] and five categories from the MGN dataset [4]. We compare our approach with two state-of-the-art volume rendering based methods – NeuS [48] and HFS [49], and a surface rendering based method – IDR [55].

We report the Chamfer Distance averaged on five examples for each category from Deep Fashion 3D Dataset [19] and report the Chamfer Distance averaged on two examples for each category from Multi-Garment Net Dataset in Table 2. NeAT generally provides lower numerical errors compared with the state-of-the-arts. We show qualitative results in Fig. 7. NeAT also provides lower numerical errors in F-score. Please refer to the supplemental for the comparisons.

In most cases, NeuS [48] and IDR [55] are able to reconstruct the geometry with thick, watertight surfaces. While, for the pants in Figure 7, NeuS fails to recover the shape of the waist. NeAT is able to reconstruct high-fidelity open surfaces with consistent normals, including the thin straps of the camisoles and dresses.

4.4. Single View Reconstruction on Open Surfaces

We construct an autoencoder, which accepts a single image as the input, and exports the 3D mesh as the output. For this experiment, we compare our approach against the state-of-the-art single-view reconstruction method: DVR [37] and the volume rendering based method: NeuS [48].

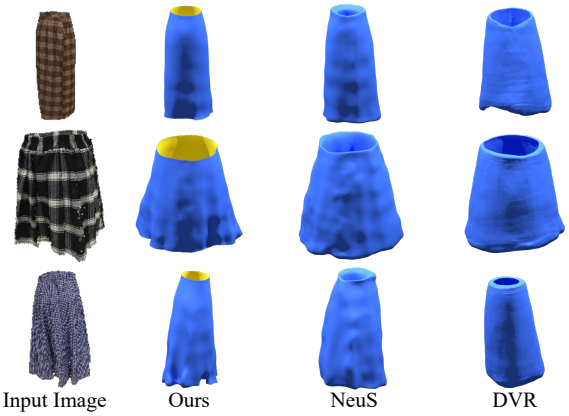

Figure 8. With given single-view images, ours predicts accurate 3D geometry of arbitrary shapes with the autoencoder. NeAT achieves CD = 0.0771 averaged on the 25 objects from the test set, which outperforms NeuS [48] (CD = 0.0778) and DVR [37] (CD = 0.0789).

The qualitative results is shown in Fig 8. Our method is able to infer accurate 3D shape representations from single-view images when only using 2D multi-view images and object masks as supervision. Qualitatively, in contrast to the DVR [37] and NeuS [48] autoencoder, our method is able to reconstruct open surfaces. Quantitatively, our method achieves CD = 0.0771, which outperforms NeuS (CD = 0.0778) and DVR (CD = 0.0789) averaged on all the 25 objects from the test set.

4.5. Ablation Studies

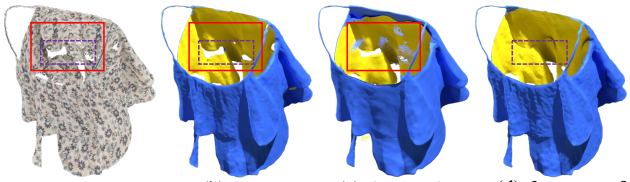

Figure 9. Ablation study on the regularizations about validity.

Regularizations on validity. We conduct an ablation study on the regularizations about validity, i.e. $\mathcal{L}_{bce}$ and $\mathcal{L}_{sparse}$. As shown in Figure 9 (c), by setting $\mathcal{L}_{bce} = 0$, the renderer tends to generate rendering probability between 0 and 1, thus resulting in noisy faces in the output mesh; as shown in Figure 9 (d), by setting $\mathcal{L}_{sparse} = 0$, the renderer will keep the redundant surfaces, instead of learning a validity space as sparse as possible.

Reconstruct with different number of views. We additionally show results on reconstruction with different number of views. As shown in Figure 10, our method is able to reconstruct open surfaces even with sparse viewpoints. The reconstruction quality improves with the increase of views, quantitively and qualitatively.

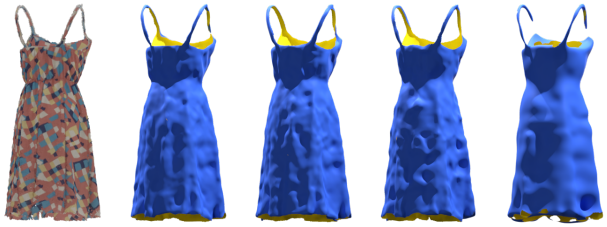

Figure 10. Ablation study on multi-view reconstruction with different number of views.

5. Discussions and Conclusions

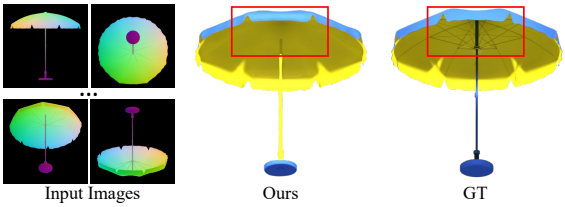

Figure 11. A failure case: our method fails to reconstruct the thin stretchers of the umbrella.

Limitations and Future works. Figure 7 Row 1 illustrates that the void space around the collar is obscured by limited input views and occlusions, causing it to be mistakenly connected with the jacket's main body by all reconstruction methods. In addition, the output normal orientation of NeAT is influenced by the initialization of SDF-Net. However, by using geometric initialization [15], we can easily set the surface to have an initial outward normal distribution, allowing us to obtain out-facing 3D reconstructions. Finally, our method has difficulty in reconstructing very thin closed surfaces, such as the umbrella stretchers in Figure 11.

A future avenue would be introducing more advanced adaptive sampling and weighting mechanisms to reconstruct highly intricate structures. Another direction for future work is extending NeAT to handle in-the-wild images without camera parameters, which can enable our method to leverage more image sources for 3D unsupervised learning.

Conclusions. We have proposed NeAT, a novel approach to reconstruct high-fidelity arbitrary surfaces with consistent normals from multi-view images. By representing the surface as a combination of the SDF and the validity probability, we develop a new volume rendering method for learning the implicit representation. Our method outperforms the state-of-the-art neural surface reconstruction methods on reconstructing open surfaces and achieves comparative results on reconstructing watertight surfaces.

References

[1] Panos Achlioptas, Olga Diamanti, Ioannis Mitliagkas, and Leonidas Guibas. Learning representations and generative models for 3d point clouds, 2018. 2

[2] Matan Atzmon and Yaron Lipman. Sal: Sign agnostic learning of shapes from raw data. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2

[3] 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

[4] Bharat Lal Bhatnagar, Garvita Tiwari, Christian Theobalt, and Gerard Pons-Moll. Multi-garment net: Learning to dress 3d people from images. In *IEEE International Conference on Computer Vision (ICCV)*. IEEE, oct 2019. 1, 5, 7

[5] Angel X. Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. ShapeNet: An Information-Rich 3D Model Repository. Technical Report arXiv:1512.03012 [cs.GR], Stanford University — Princeton University — Toyota Technological Institute at Chicago, 2015. 2

[6] Weikai Chen, Cheng Lin, Weiyang Li, and Bo 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. 1, 2, 4

[7] Wenzheng Chen, Huan Ling, Jun Gao, Edward Smith, Jaakko Lehtinen, Alec Jacobson, and Sanja Fidler. Learning to predict 3d objects with an interpolation-based differentiable renderer. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. 2

[8] Zhiqin Chen, Andrea Tagliasacchi, Thomas Funkhouser, and Hao Zhang. Neural dual contouring. *ACM Transactions on Graphics (Special Issue of SIGGRAPH)*, 41(4), 2022. 2

[9] Zhiqin Chen and Hao Zhang. Learning implicit fields for generative shape modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5939–5948, 2019. 2

[10] Julian Chibane, Aymen Mir, and Gerard Pons-Moll. Neural unsigned distance fields for implicit function learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, December 2020. 1, 2

[11] Christopher B. Choy, Danfei Xu, JunYoung Gwak, Kevin Chen, and Silvio Savarese. 3d-r2n2: A unified approach for single and multi-view 3d object reconstruction. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *Computer Vision – ECCV 2016*, pages 628–644, Cham, 2016. Springer International Publishing. 2

[12] Haoqiang Fan, Hao Su, and Leonidas Guibas. A point set generation network for 3d object reconstruction from a single image. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2463–2471, 2017. 2

[13] Kyle Genova, Forrester Cole, Aaron Maschinot, Aaron Sarna, Daniel Vlasic, and William T Freeman. Unsupervised training for 3d morphable model regression. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8377–8386, 2018. 2

[14] Kyle Genova, Forrester Cole, Aaron Maschinot, Aaron Sarna, Daniel Vlasic, and William T. Freeman. Unsupervised training for 3d morphable model regression. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8377–8386, 2018. 2

[15] Amos Gropp, Lior Yariv, Niv Haim, Matan Atzmon, and Yaron Lipman. Implicit geometric regularization for learning shapes. In *Proceedings of Machine Learning and Systems 2020*, pages 3569–3579. 2020. 8

[16] Thibault Groueix, Matthew Fisher, Vladimir G. Kim, Bryan Russell, and Mathieu Aubry. AtlasNet: A Papier-Mâché Approach to Learning 3D Surface Generation. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2018. 2

[17] Benoit Guillard, Federico Stella, and Pascal Fua. Meshudf: Fast and differentiable meshing of unsigned distance field networks. In *European Conference on Computer Vision*, 2022. 2

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 6

[19] Zhu Heming, Cao Yu, Jin Hang, Chen Weikai, Du Dong, Wang Zhangye, Cui Shuguang, and Han Xiaoguang. Deep fashion3d: A dataset and benchmark for 3d garment reconstruction from single images. In *Computer Vision – ECCV 2020*, pages 512–530. Springer International Publishing, 2020. 1, 5, 6, 7

[20] 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, 2014. 1, 5, 6

[21] Yue Jiang, Dantong Ji, Zhizhong Han, and Matthias Zwicker. Sdfdiff: Differentiable rendering of signed distance fields for 3d shape optimization. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2

[22] Hiroharu Kato, Yoshitaka Ushiku, and Tatsuya Harada. Neural 3d mesh renderer. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3907–3916, 2018. 2

[23] 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*, pages 1–12, 2022. 2

[24] Chen-Hsuan Lin, Chen Kong, and Simon Lucey. Learning efficient point cloud generation for dense 3d object reconstruction. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2018. 2

[25] Shichen Liu, Weikai Chen, Tianye Li, and Hao Li. Soft rasterizer: A differentiable renderer for image-based 3d reasoning. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 7707–7716, 2019. 2

[26] Shichen Liu, Shunsuke Saito, Weikai Chen, and Hao Li. Learning to infer implicit surfaces without 3d supervision. *Advances in Neural Information Processing Systems*, 32, 2019. 1, 2

[27] Shaohui Liu, Yinda Zhang, Songyou Peng, Boxin Shi, Marc Pollefeys, and Zhaopeng Cui. Dist: Rendering deep implicit signed distance function with differentiable sphere tracing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 1, 2

[28] Yu-Tao Liu, Li Wang, Jie Yang, Weikai Chen, Xiaoxu Meng, Bo Yang, and Lin Gao. Neudf: Leaning neural unsigned distance fields with volume rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 3

[29] Xiaoxiao Long, Cheng Lin, Lingjie Liu, Yuan Liu, Peng Wang, Christian Theobalt, Taku Komura, and Wenping 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 (CVPR)*, 2023. 3

[30] William E. Lorensen and Harvey 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. 2, 3

[31] Priyanka Mandikal, K L Navaneet, Mayank Agarwal, and R Venkatesh Babu. 3D-LMNet: Latent embedding matching for accurate and diverse 3d point cloud reconstruction from a single image. In *Proceedings of the British Machine Vision Conference (BMVC)*, 2018. 2

[32] Daniel Maturana and Sebastian Scherer. Voxnet: A 3d convolutional neural network for real-time object recognition. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 922–928, 2015. 2

[33] 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. 2

[34] Mateusz Michalkiewicz, Jhony Kaesemodel Pontes, Dominic Jack, Mahsa Baktashmotlagh, and Anders Eriksson. Implicit surface representations as layers in neural networks. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4742–4751, 2019. 2

[35] 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. 2, 5, 6

[36] 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

[37] Michael Niemeyer, Lars Mescheder, Michael Oechsle, and Andreas Geiger. Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2, 4, 7, 8

[38] 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

[39] 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. 2

[40] Nikhila Ravi, Jeremy Reizenstein, David Novotny, Taylor Gordon, Wan-Yen Lo, Justin Johnson, and Georgia Gkioxari. Accelerating 3d deep learning with pytorch3d. *arXiv:2007.08501*, 2020. 6

[41] Shunsuke Saito, , Zeng Huang, Ryota Natsume, Shigeo Morishima, Angjoo Kanazawa, and Hao Li. Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization. *arXiv preprint arXiv:1905.05172*, 2019. 2

[42] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 6

[43] 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. 6

[44] Hang Su, Subhransu Maji, Evangelos Kalogerakis, and Erik Learned-Miller. Multi-view convolutional neural networks for 3d shape recognition. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 945–953, 2015. 2

[45] Rahul Venkatesh, Tejan Karmali, Sarthak Sharma, Aurobrata Ghosh, R. Venkatesh Babu, László A. Jeni, and Maneesh Singh. Deep implicit surface point prediction networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 12653–12662, October 2021. 2

[46] Rahul Venkatesh, Sarthak Sharma, Aurobrata Ghosh, Laszlo Jeni, and Maneesh Singh. Dude: Deep unsigned distance embeddings for hi-fidelity representation of complex 3d surfaces. *arXiv preprint arXiv:2011.02570*, 2020. 2

[47] Nanyang Wang, Yinda Zhang, Zhuwen Li, Yanwei Fu, Hang Yu, Wei Liu, Xiangyang Xue, and Yu-Gang Jiang. Pixel2mesh: 3d mesh model generation via image guided deformation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(10):3600–3613, 2021. 2

[48] 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. *NeurIPS*, 2021. 1, 2, 4, 5, 6, 7, 8

[49] Yiqun Wang, Ivan Skorokhodov, and Peter Wonka. Hf-neus: Improved surface reconstruction using high-frequency details. *arXiv preprint arXiv:2206.07850*, 2022. 2, 6, 7

[50] Chao Wen, Yinda Zhang, Zhuwen Li, and Yanwei Fu. Pixel2mesh++: Multi-view 3d mesh generation via deformation. In *ICCV*, 2019. 2

[51] Francis Williams. Point cloud utils, 2022. https://www.github.com/fwilliams/point-cloud-utils. 6

[52] Qiangeng Xu, Weiyue Wang, Duygu Ceylan, Radomir Mech, and Ulrich Neumann. Disn: Deep implicit surface network for high-quality single-view 3d reconstruction. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. 2

[53] Yaoqing Yang, Chen Feng, Yiru Shen, and Dong Tian. Foldingnet: Point cloud auto-encoder via deep grid deformation. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 206–215, 2018. 2

[54] Lior Yariv, Jiatao Gu, Yoni Kasten, and Yaron Lipman. Volume rendering of neural implicit surfaces. In *Thirty-Fifth Conference on Neural Information Processing Systems*, 2021. 2

[55] 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, 4, 5, 6, 7

[56] Jianglong Ye, Yuntao Chen, Naiyan Wang, and Xiaolong Wang. Gifs: Neural implicit function for general shape representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12829–12839, June 2022. 2