Learning Neural Implicit Surfaces with Object-Aware Radiance Fields

Yiheng Zhang, Zhaofan Qiu, Yingwei Pan, Ting Yao, and Tao Mei
University of Science and Technology of China HiDream.ai Inc.
{yihengzhang.chn, zhaofanqiu, panyw.ustc, tingyao.ustc}@gmail.com, tmei@hidream.ai

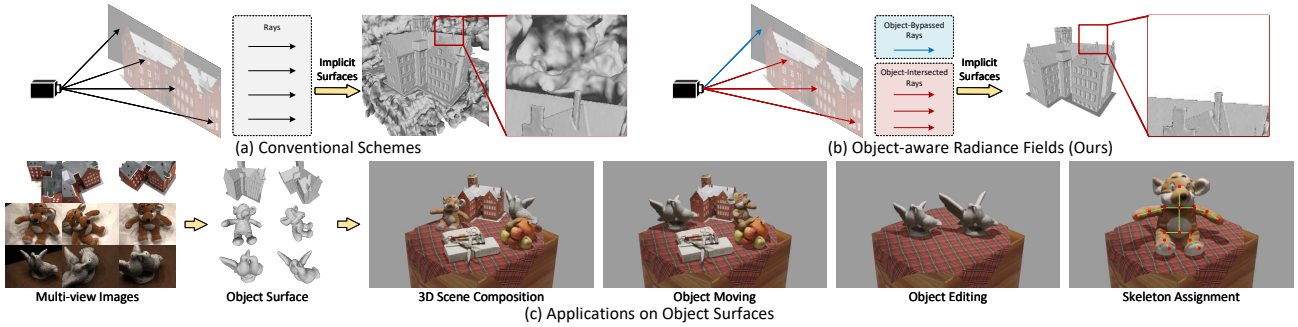

Figure 1: The implicit surfaces learnt via (a) conventional scheme which suffers from noisy background geometry representations. In this work, we introduce (b) an object-aware volumetric scene representation by automatically recognizing rays as object-intersected or object-bypassed, aiming to highlight the key object surface in the foreground. (c) Such object surface benefits a series of applications, e.g., 3D scene composition, object moving/editing, and even skeleton assignment.

Abstract

Recent progress on multi-view 3D object reconstruction has featured neural implicit surfaces via learning high-fidelity radiance fields. However, most approaches hinge on the visual hull derived from cost-expensive silhouette masks to obtain object surfaces. In this paper, we propose a novel Object-aware Radiance Fields (ORF) to automatically learn an object-aware geometry reconstruction. The geometric correspondences between multi-view 2D object regions and 3D implicit/explicit object surfaces are additionally exploited to boost the learning of object surfaces. Technically, a critical transparency discriminator is designed to distinguish the object-intersected and object-bypassed rays based on the estimated 2D object regions, leading to 3D implicit object surfaces. Such implicit surfaces can be directly converted into explicit object surfaces (e.g., meshes) via marching cubes. Then, we build the geometric correspondence between 2D planes and 3D meshes by rasterization, and project the estimated object regions into 3D explicit object surfaces by aggregating the object information across multiple views. The aggregated object information in 3D explicit object surfaces is further reprojected back to 2D planes, aiming to update 2D object regions and enforce them to be multi-view consistent. Extensive experiments on DTU and BlendedMVS verify the capability of ORF to produce comparable surfaces against the state-of-the-art models that demand silhouette masks.

1. Introduction

Multi-view 3D reconstruction, i.e., the task of reconstructing 3D geometry/surface of objects from multi-view 2D images, has played a fundamental challenge to computer vision and computer graphics communities for decades. In the early stage, the mainstream solution is the classical Multi-View Stereo (MVS) [3, 5, 12, 20, 34, 35] that exploits photometric consistency across different camera views to learn explicit geometry representations (e.g., meshes or voxel grids). The ultimate reconstruction relies heavily on the quality of cross-view matching. In practice, such matching often fails to associate objects with sparse textures, resulting in severe artifact or missing parts on surfaces. To address this issue, recent studies turn their focus on investigating how to represent 3D surfaces as implicit geometry representations. Many consider learning a continuous implicit function that formulates neural implicit surfaces in occupancy field [25, 31] or signed distance field [29]. In between, surface rendering techniques [27, 47, 49] are designed to optimize these fields, leading to impressive reconstruction quality via differentiable rendering from images. Nevertheless, learning such implicit geometry representations requires additional object masks of scenes, since the color of each ray is assumed to only correspond to a single point where a surface intersects with this ray. Moreover, the gradient of differentiable rendering is only backpropagated to the local surface near intersection, resulting in a sub-optimal solution for neural implicit surfaces.

To mitigate these limitations of local gradient propagation and the demand for the input object masks during fields optimization, a series of volume rendering based neural radiance fields techniques [9, 28, 41, 46] start to emerge. Recently, the seminal work of NeRF [26] builds up the foundation of volumetric scene in neural radiance fields for view synthesis, and performs classical volume rendering via alpha-compositing colors of the sampled points along rays. The subsequent methods further remould classical volume rendering by imposing implicit surface representations, e.g., occupancy network [28] or signed distance function [9, 41, 46]. These approaches produce more accurate surfaces by reconstructing the holistic scenery. Nevertheless, the learnt implicit surfaces are inevitably composed of rich geometry representations in both foreground and background (Figure 1 (a)), which are not distinguished during radiance fields optimization. Accordingly, existing techniques utilize additional cost-expensive silhouette masks to trim the learnt 3D meshes derived from implicit surfaces. The removal of vertices and/or surfaces outside the visual hull directly highlights the geometry of the key objects in the foreground, yielding object surfaces.

In this work, we devise a new Object-aware Radiance Field (ORF), which goes one step further to eliminate the need of the visual hull derived from human-annotated object masks in scenes for learning object surfaces. Our launching point is to introduce an object-aware volumetric scene representation by inferring the foreground/object and background radiance fields on-the-fly. Both the geometric correspondences between multi-view 2D object regions and 3D implicit or explicit object surfaces are exploited to boost the learning of object-aware volumetric scene representation. ORF is henceforth able to encourage the reconstruction of foreground/object geometry without additional 3D supervision of visual hull. Specifically, we first estimate 2D object regions directly based on the multi-view images. Next, on the basis of the estimated 2D object regions, ORF capitalizes on a transparency discriminator to automatically recognize the transparency of each ray in the radiance field. A low transparency indicates that the ray intersects with the object, and rays with high transparency are considered to be object-bypassed. Such predicted transparency of rays is further regarded as prior information to regularize the radiance field in an object-aware manner. Especially, as shown in Figure 1 (b), all the sampled rays are divided into object-intersected rays and object-bypassed rays according to predictions given by the transparency discriminator, leading to 3D implicit object surfaces. After that, we leverage marching cubes to directly convert the 3D implicit object surfaces into 3D explicit object surfaces. The inherent geometric correspondence between 2D planes and 3D explicit object surfaces are thus constructed via rasterization. This geometric correspondence enables the projection from the estimated 2D object regions into 3D explicit object surfaces, and meanwhile the object information across multiple views are aggregated. Furthermore, ORF projects the aggregated object information in 3D explicit object surfaces back to 2D planes, thereby updating the 2D object regions and enforcing them multi-view consistent. The whole process refines the estimated 2D object regions and 3D implicit/explicit object surfaces, pursuing an object reconstruction without trivial background surfaces.

In sum, we have made the following contributions: (**I**) ORF designs a transparency discriminator to automatically capture useful object-aware inductive bias, which further supervises radiance fields to learn object-aware geometry reconstruction. (**II**) ORF additionally mines the inherent geometric correspondence between multi-view 2D object regions and 3D object surfaces to refine them along with volume rendering. (**III**) We evaluate ORF on two widely-used benchmarks (DTU and BlendedMVS), demonstrating the effectiveness of our proposal.

2. Related Work

Multi-view 3D Reconstruction is one of the fundamental tasks in 3D vision, which attempts to reconstruct the 3D geometry from images captured from multiple views. The early works on this task has proceeded along two different dimensions: matching features across views [4, 34] and representing shapes with a voxel grid [1, 5, 10, 20, 30, 36, 38, 39]. The first dimension reconstructs the 3D scenes by matching the pixels with similar appearance, and usually requires the complex designs for fusing depth information [8, 24] or meshing [17, 18]. The second dimension represents the 3D geometry as a volume grid and is limited by the cubic increase of memory requirements. After that, inspired by the recent advance of deep learning techniques, deep neural networks are exploited as an alternative of hand-designed traditional components in 3D reconstruction. For instance, the visual feature for matching [14, 21, 23, 40, 48], depth fusion [11, 33] and depth map prediction [15, 43, 44] can be learnt by neural networks, leading to better performances thanks to the high learning capacity of deep models.

Instead of explicitly reconstructing the 3D geometry, the **neural implicit surfaces** are introduced to encode the characteristics of 3D scenes by neural networks. Specifically, the surfaces are represented by a neural network which outputs either an occupancy field [25, 31] or a Signed Distance Function (SDF) [29]. In this scheme, the implicit representations of surfaces are learnt via surface rendering [27, 47] which determines the radiance directly on the surface of an object and provides a differentiable rendering formulation using implicit gradients. Recently, Neural Radiance Fields (NeRF) [26] is proposed to implicitly represent scenes by volume rendering which learns alpha-compositing of the radiance field along camera rays, which benefits 3D con-

tent creation tasks [51, 6, 7, 42]. The main focus of NeRF is the quality of novel view synthesis, but the geometry is not guaranteed. To promote the quality of learnt surfaces, Unisurf [28] unifies radiance fields and occupancy-based implicit surfaces, enabling plausible reconstruction. VolSDF [46] and NeuS [41] exploit SDF as implicit surface representations and obtain better results. NeuralWarp [9] further boosts up the reconstruction by integrating photo consistency between multi-view images.

Our work also falls into the category of learning neural implicit surfaces with volume rendering. Despite the progress in improving rendering quality or surface extraction, the way to reconstruct object surfaces without requiring the visual hull derived from object masks in scenes has not been fully explored. The related work of DFFs [19] distills off-the-shelf 2D vision encoders into a 3D feature field to enable the decomposition of scenes conditioning on user-specified queries. Different from [19], our ORF exploits the geometric correspondence between multi-view 2D object regions and 3D object surfaces to automatically separate objects and background, and enhance the quality of the learnt object surfaces.

3. Object-aware Radiance Fields

This section first reviews the standard scheme to represent scenes via radiance fields without the consideration of object region separation (Section 3.1). Next, we describe how the designed transparency discriminator in our ORF distinguishes the object-intersected rays and object-bypassed rays in radiance fields based on the estimated object regions. And then we introduce how to learn an object-aware radiance field by optimizing rays separately with the help of the transparency discriminator (Section 3.2). Furthermore, we design a novel rasterization-based aggregation mechanism to additionally construct the geometric correspondence between 2D object regions and 3D object surfaces via rasterization. Such geometric correspondence enables the aggregation of object information from 2D planes to 3D explicit object surfaces. The aggregated object information in turn updates the 2D object regions in a multi-view consistent manner, pursuing a more high-fidelity object reconstructions (Section 3.3). Figure 2 depicts an overview of our ORF architecture.

3.1. Representing Scene via Radiance Fields

Neural radiance field and its recent variants aim to represent 3D scenes by two implicit functions, i.e., geometry function and radiance function, which are approximated via two individual neural networks. The geometry network models the geometric structure of 3D scenes, and the radiance network encodes the color emitted by any region in space from all directions. The two networks represent the characteristics in the 3D scene and codetermine the rendered novel views. When rendering the target view $\mathbf{R}$ taken from the direction $\mathbf{d}$, the rendered color $\mathbf{R}(\mathbf{p})$ of each pixel $\mathbf{p}$ is computed by both geometry network $\mathcal{G}$ and radiance network $\mathcal{C}$ in a differentiable way. Specifically, we first sample N_p points $\mathbf{x}_i,\ i = 1, ..., N_p$ along the corresponding ray $\mathfrak{R}(\mathbf{p})$ going through the camera center and the target pixel $\mathbf{p}$. Note that here we omit the notations of target pixel for simplicity. The surface normal $\mathbf{n}_i$, the occupancy value α_i and the radiance value $\mathbf{c}_i$ are measured by the geometry network $\mathcal{G}$ and the radiance network $\mathcal{C}$ as

$$\begin{aligned} &\mathbf{n}_i,\ \ \alpha_i = \mathcal{G}(\mathbf{x}_i), \\ &\mathbf{c}_i = \mathcal{C}(\mathbf{x}_i, \mathbf{n}_i, \mathbf{d}). \end{aligned} \tag{1}$$

The rendering of the scene at pixel $\mathbf{p}$ is approximated as a weighted summation of radiance values $\mathbf{c}_i$ at each sampled point by using the occupancy values as the weights:

$$\mathbf{R}(\mathbf{p}) = \sum_{i=1}^{N_p} \alpha_i \prod_{j<i} (1 - \alpha_j)\mathbf{c}_i, \tag{2}$$

where $\prod_{j<i}(1 - \alpha_j)$ is to simulate the occlusion caused by the opaque pixels in front of the target pixel. As a practical choice, we utilize transformed Signed Distance Field (SDF) encoding [41, 46] as the geometry function. Eq. (2) is actually the discrete approximation of an integral along the camera ray by sampling a limited number of points. The sampling strategies have been discussed in [26, 28, 41, 46] to improve the geometry estimation.

At training stage, the two neural networks are jointly optimized by encouraging the reconstruction of the given views of the 3D scene. Given one reference view $\mathbf{I}^r$, the reconstruction loss for $\mathbf{p} \in \mathbf{I}^r$ is calculated by the L1 distance between the reference view $\mathbf{I}^r$ and the neural volume rendering results $\mathbf{R}$ as

$$\mathrm{L}_{rec} = |\mathbf{I}^r(\mathbf{p}) - \mathbf{R}(\mathbf{p})|, \tag{3}$$

where $|\cdot|$ is the L1 loss. This loss function simply treats each camera ray corresponding to $\mathbf{p}$ equally and constrains the difference between the input pixel and the rendered result, while ignoring the distinction of rays that either intersect or bypass the objects. As a result, the implicit 3D model often contains some trivial surfaces of the background regions (Figure 1), and thus requires the object masks derived visual hull to execute an additional trimming step at inference.

3.2. Learning Object-aware Radiance Fields

To alleviate these issues, we propose to automatically distinguish the object-intersected and object-bypassed rays, making the learnt radiance fields object-aware without manually annotating the visual hull for each scene. To materialize this idea, here we first elaborate how to achieve the high-quality ray/transparency prediction via a devised transparency discriminator (Figure 2 (a)), followed by the learning scheme with different objectives (Figure 2 (b)).

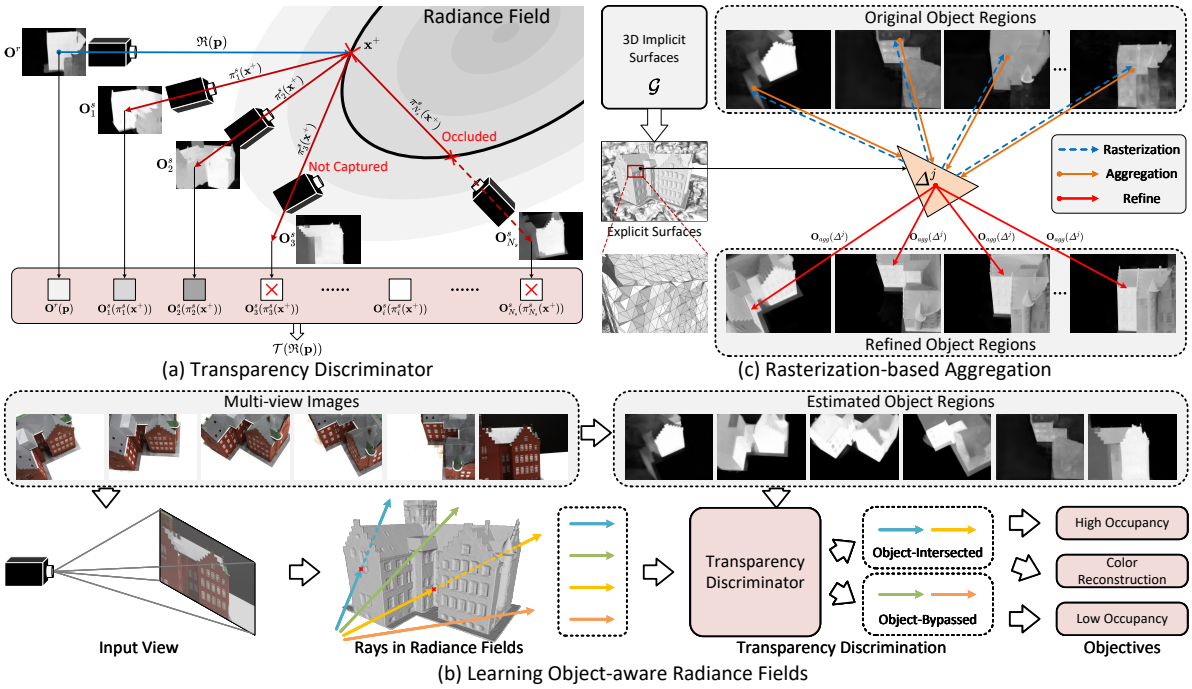

Figure 2: An overview of the proposed Object-aware Radiance Fields (ORF) framework for learning neural implicit surfaces.

Transparency Discriminator. Formally, we consider the ray $\mathfrak{R}(\mathbf{p})$ derived from pixel $\mathbf{p}$ of the reference view $\mathbf{I}^r$. One intuitive way to inspect whether the $\mathfrak{R}(\mathbf{p})$ intersects with the object is to predict the probability that the pixel $\mathbf{p}$ belongs to the estimated object regions $\mathbf{O}^r$ from the perspective of $\mathbf{I}^r$. Note that here we employ an off-the-shelf salient object segmentation model [32] to learn the prior knowledge of generic salient objects. Replacing the saliency model with other segmentation models is flexible to generalize to other reconstruction scenarios. Such saliency map generated from $\mathbf{I}^r$ provides coarse object information that indicates the 2D object regions $\mathbf{O}^r$. However, the complex object appearance and challenging variation of views probably affect the quality of object information. As such, we further design a transparency discriminator that jointly exploits the 2D object regions estimated from different views for ray discrimination in radiance field.

In particular, given a reference image $\mathbf{I}^r$ and multiple source images $(\mathbf{I}_i^s, i = 1, \ldots, N_s)$ captured from different viewpoints, we can roughly estimate the object regions in each image individually via salient object segmentation. Let $\mathbf{O}^r$ and $\mathbf{O}_i^s$ $(i = 1, \ldots, N_s)$ denote the corresponding 2D object regions. We start from the ray $\mathfrak{R}(\mathbf{p})$ corresponding to pixel $\mathbf{p} \in \mathbf{I}^r$ that intersects with the implicit surface for the first time at point $\mathbf{x}^+$ in the radiance field. To locate the point $\mathbf{x}^+$ in the world coordinates, we sample points along $\mathfrak{R}(\mathbf{p})$ from its corresponding camera center and feed the sampled points into the geometry network $\mathcal{G}$ in order. Then, following [46], the signed distance between each sampled point and its nearest surface is derived from the output of $\mathcal{G}$. The first point we sampled that has a positive signed distance is regarded as $\mathbf{x}^+$.

After that, we mine the geometric correspondence between 3D implicit surfaces and 2D planes by projecting the point $\mathbf{x}^+$ into the 2D plane of each source image. By denoting the projection as $\pi_i^s(\mathbf{x}^+)$ $(i = 1, \ldots, N_s)$ in source image coordinates, we formulate $\pi_i^s(\mathbf{x}^+)$ as:

$$\pi_i^s(\mathbf{x}^+) = K_i^s(R_i^s\mathbf{x}^+ + \mathbf{t}_i^s), \tag{4}$$

where K_i^s is the internal calibration matrix of the i-th source camera, and $(R_i^s, \mathbf{t}_i^s)$ are the extrinsic parameters of the camera (R_i^s: 3×3 rotation matrix, $\mathbf{t}_i^s$: 3-dim translation vector). In this way, each projection associates the point $\mathbf{x}^+$ in world coordinates to pixels in source images.

Based on the estimated 2D object regions from multiple views $\mathbf{O}^r$ and $\mathbf{O}_i^s$ $(i = 1, \ldots, N_s)$, a transparency discriminator is devised to predict the transparency $\mathcal{T}(\mathfrak{R}(\mathbf{p}))$. Here the transparency can be interpreted as the probability that the ray $\mathfrak{R}(\mathbf{p})$ bypasses the target object. Specifically, in an effort to take all 2D object regions into account, we measure the transparency for $\mathfrak{R}(\mathbf{p})$ by averaging the probabilities of pixels belonging to objects in multiple views with regard to the same point $\mathbf{x}^+$:

$$\mathcal{T}(\mathfrak{R}(\mathbf{p})) = 1 - \frac{\mathbf{O}^r(\mathbf{p}) + \sum_{i=1}^{N_s} \mathbf{O}_i^s(\pi_i^s(\mathbf{x}^+))}{1 + N_s}. \tag{5}$$

Such formulation assumes that the intersection point $\mathbf{x}^+$ are observed by each source camera without any occlusion. But in practice, the intersection points may be projected outside of the source images or occluded by other surfaces which

are closer to the source cameras. To tackle this issue, we extend the formulation in Eq. (5) with a binary indicator $V(\cdot)$:

$$\mathcal{T}(\mathfrak{R}(\mathbf{p})) = 1 - \frac{\mathbf{O}^r(\mathbf{p}) + \sum_{i=1}^{N_s} V(\pi_i^s(\mathbf{x}^+))\mathbf{O}_i^s(\pi_i^s(\mathbf{x}^+))}{1 + \sum_{i=1}^{N_s} V(\pi_i^s(\mathbf{x}^+))}. \tag{6}$$

We set $V(\pi_i^s(\mathbf{x}^+))$ as 1 when $\mathbf{x}^+$ is within the maximum area that can be captured by the camera of $\mathbf{I}_i^s$ and meanwhile it is also the first intersection between ray $\mathfrak{R}(\pi_i^s(\mathbf{x}^+))$ and implicit surfaces. In the implementation, we first check if the $\pi_i^s(\mathbf{x}^+)$ locates inside the source image $\mathbf{I}_i^s$. Then we estimate the first intersection point of ray $\mathfrak{R}(\pi_i^s(\mathbf{x}^+))$ and check whether it is nearby $\mathbf{x}^+$. Accordingly, our transparency discriminator is able to identify an object-intersected ray with low transparency, and a ray with high transparency is considered to bypass the object.

Objective. After classifying each ray via the transparency discriminator, we assign different objectives to different kinds of rays, pursuing an object-aware radiance field. Figure 2 (b) shows an overview of the learning scheme for our ORF. In general, given a group of multi-view images of the target object, camera rays associated with pixels in images are utilized for volume rendering to optimize the radiance field, as described in Section 3.1. In the radiance field, a ray may terminate at the surface of objects (blue and golden), the ground plane (orange), or the point at infinity (green). To differentiate these kinds of rays, we feed them into a transparency discriminator and classify them as object-intersected or object-bypassed according to object regions estimated from the multi-view images.

Specifically, for the object-intersected ray whose predicted transparency is less than 0.5, we learn in advance that the ray will go through the target object. Hence, we additionally constrain the ray to have high occupancy as a supplement to the reconstruction loss in Eq. (3):

$$\mathrm{L}_{occ} = -log(\sum_{i=1}^{N_p} \alpha_i \prod_{j<i}(1-\alpha_j)), \mathcal{T}(\mathfrak{R}(\mathbf{p})) \leq 0.5. \tag{7}$$

In contrast, for the object-bypassed ray with $\mathcal{T}(\mathfrak{R}(\mathbf{p})) > 0.5$, the occupancy is enforced to be small to ensure that no trivial surface appears in the background regions:

$$\mathrm{L}_{occ} = -log(1 - \sum_{i=1}^{N_p} \alpha_i \prod_{j<i}(1-\alpha_j)), \mathcal{T}(\mathfrak{R}(\mathbf{p})) > 0.5. \tag{8}$$

In addition, we do not require the rendering of object-bypassed ray to reconstruct the input color, which means the reconstruction loss $|\mathbf{I}^r(\mathbf{p}) - \mathbf{R}(\mathbf{p})|$ is omitted when $\mathfrak{R}(\mathbf{p})$ is object-bypassed.

Hence, the overall objective function integrates the reconstruction loss between the reference view and the rendered results in Eq. (3), and the occupancy regularization loss in Eq. (7) and Eq. (8). To further enhance the quality of the learnt surfaces, we also consider two extra losses following the recent works. The first one is the eikonal loss L_{eik} [13] which encourages the geometry network to output a function similar to a signed distance field. The second one is the patch warping loss L_{warp} [9] to improve the reconstruction ability by warping the existing patches from the reference views. We measure the overall objective as:

$$\mathrm{L} = \lambda_1 \mathrm{L}_{rec} + \lambda_2 \mathrm{L}_{occ} + \lambda_3 \mathrm{L}_{eik} + \lambda_4 \mathrm{L}_{warp}, \tag{9}$$

where $\lambda_1 = 1$, $\lambda_2 = 0.1$, $\lambda_3 = 0.1$, and $\lambda_4 = 1$ are the trade-off parameters. Here we set λ_3 and λ_4 following [9], and λ_2 is determined through experimental study.

3.3. Rasterization-based Aggregation

The new rasterization-based aggregation mechanism additionally exploits the geometric correspondence between 3D explicit surfaces and 2D planes. This design aims to further refine the quality of the estimated 2D object regions and make them multi-view consistent.

Technically, as shown in Figure 2 (c), the mechanism first converts the 3D implicit surfaces (i.e., geometry network $\mathcal{G}$) into explicit surfaces (a explicit polygonal 3D representation like mesh) via marching cubes [22]. After that, we compute the projection from the 3D explicit surfaces (i.e., triangles) to the pixels of 2D images $\mathcal{I}$ by rasterization. Here the rasterization refers to the typical process of taking a triangle and figuring out which pixels it covers [37]. Formally, let Δ^j denote the j-th triangle in the mesh, which corresponds to a set of pixels $\mathcal{P}_i^j \subseteq \mathbf{O}_i$ in the i-th view. Note that the pixel set $\mathcal{P}_i^j$ is set as empty when Δ^j is invisible to the camera due to invalid projections (e.g., the triangle is projected outside the image) or occlusion. Such correspondence between triangles and pixels enables the aggregation of the object information of 2D object regions into 3D explicit surface Δ^j. In particular, we aggregate the object probabilities of all the pixels belonging to the triangle Δ^j from multiple views, yielding the triangle-level object probability of $\mathbf{O}_{agg}(\Delta^j)$:

$$\mathbf{O}_{agg}(\Delta^j) = \frac{\sum_{i=1}^{|\mathcal{I}|} \sum_{\mathbf{p}\in\mathcal{P}_i^j} \mathbf{O}_i(\mathbf{p})}{\epsilon + \sum_{i=1}^{|\mathcal{I}|} \sum_{\mathbf{p}\in\mathcal{P}_i^j} \mathbf{1}}, \tag{10}$$

where ϵ is a constant term added to the denominator for numerical stability. After traversing all the 3D surfaces with the triangle-level aggregation (Eq. (10)), we re-project the triangle-level object probabilities of 3D surfaces back to 2D images. More specifically, for one pixel $\mathbf{p} \in \mathcal{P}_i^j \subseteq \mathbf{O}_i$ in each viewpoint, the object probability of $\mathbf{O}_i(\mathbf{p})$ is updated as the same triangle-level object probability $\mathbf{O}_{agg}(\Delta^j)$. By doing so, the updated 2D multi-view object regions are enforced to be consistent across different views.

Table 1: Comparisons of our ORF with other volume rendering techniques on DTU dataset for learning neural implicit surfaces. Here all the involved techniques do not use object masks as additional supervision for training. The top part reports the reconstruction performances when using object masks to trim the estimated meshes, and the bottom part directly compares the reconstruction performances of the estimated meshes without using object masks. The best performances are marked in bold and the second-best results are underlined.

ScanID	24	37	40	55	63	65	69	83	97	105	106	110	114	118	122	Mean
with object masks: using the visual hull derived from human-annotated object masks for mesh trimming																
NeRF [26]	1.90	1.60	1.85	0.58	2.28	1.27	1.47	1.67	2.05	1.07	0.88	2.53	1.06	1.15	0.96	1.49
Unisurf [28]	1.32	1.36	1.72	0.44	1.35	0.79	0.80	1.49	1.37	0.89	0.59	1.47	0.46	0.59	0.62	1.02
VolSDF [46]	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 [41]	1.00	1.37	0.93	0.43	1.10	**0.65**	**0.57**	1.48	1.09	0.83	**0.52**	1.20	**0.35**	**0.49**	0.54	0.84
NeuralWarp [9]	**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
without object masks: without mask-dependent trimming at inference																
VolSDF [46]	1.25	1.70	1.31	0.91	2.90	1.08	0.90	1.62	1.24	1.09	0.70	1.39	0.59	0.71	0.87	1.22
NeuS [41]	1.59	1.98	1.44	0.95	1.82	0.74	0.64	1.63	1.30	1.41	0.59	1.33	0.44	0.51	0.54	1.13
NeuralWarp [9]	0.97	2.54	1.52	0.41	2.54	0.74	0.79	1.06	1.53	1.40	0.75	0.72	0.39	0.57	0.62	1.10
ORF	0.56	**0.69**	0.43	0.45	**0.74**	0.85	0.75	**0.76**	**0.86**	0.71	0.61	**0.69**	0.38	0.81	0.52	**0.65**

4. Experiments

We evaluate the effectiveness of ORF on DTU [16] and BlendedMVS [45] for multi-view 3D reconstruction. We first show both qualitative and quantitative results of ORF in comparison to existing techniques on DTU. Next, we conduct experimental analysis to validate the designs in ORF. Finally, we perform the qualitative comparison over samples derived from BlendedMVS for evaluation.

4.1. Datasets and Experimental Settings

DTU is a popular multi-view 3D reconstruction benchmark, which consists of 80 scenes with large variability in materials, appearance and geometry of objects. For each scene, the dataset provides 49 or 64 images (resolution: 1,600×1,200) captured from multiple camera views. The ground-truth point cloud of each scene is acquired with laser sensor. In our experiments, we follow [9, 28, 41, 46, 47] to use the selected 15 scenes in DTU for comparison. We strictly follow [9, 41] and use the official evaluation code of DTU [16] to compute the quantitative metrics. Specifically, the point clouds of objects (w/o background) in DTU are regarded as ground truth (GT). For a reconstructed mesh, we sample points on triangle surfaces in an evenly spaced manner (radius=0.2 as in [16]). Then, we report the final reconstruction score by averaging the accuracy and completeness of the sampled points (SP) (i.e., the chamfer distances of SP→GT and GT→SP). It is also worthy to note that existing volume rendering techniques commonly use the additional object regions manually annotated by [27, 47] to trim the predicted mesh for only evaluating the reconstruction inside the visual hull. In contrast, our ORF learns an object-aware volumetric scene representation and directly constructs the 3D geometry of objects in the scene without the use of the visual hull derived from annotated object regions of scenes.

BlendedMVS is a large-scale benchmark for multi-view 3D reconstruction, including 113 scenes with multi-view images. Compared to DTU, the backgrounds in BlendedMVS are more complex. We select samples from the lowres set of BlendedMVS, and each scene is equipped with 24 to 64 images (resolution: 768×576). Note that the ground truth of object regions for each image is not available.

Implementation Details. The whole network of ORF is built over VolSDF [46], and we adopt the same architecture as in existing volume rendering methods [28, 46, 47] for fair comparison. The whole network uses sphere initialization [2]. During training, we train our ORF with a two-stage strategy. We first optimize the network with the reconstruction loss and eikonal loss under the same setting in VolSDF (100k iterations with learning rate exponentially decayed from 5e-4 to 5e-5, batch size: 1,024 rays). Next, the network is fine-tuned with the overall objective in Eq. (9) (100k iterations with learning rate of 5e-4, batch size: 512 rays). The object probability is calculated by the transparency discriminator in each forward pass using the latest radiance field. The mesh extraction for rasterization-based aggregation is only conducted once after the first training stage of ORF.

4.2. Comparison on DTU Dataset

Quantitative Results. We compare with several neural surface reconstruction techniques under two different evaluation settings for fair comparison: 1) baselines using the visual hull derived from human-annotated object masks for mesh trimming, 2) all re-implemented SDF-based baselines and our ORF without mask-dependent trimming at inference. Such two settings are denoted as *with/without object masks*, respectively. All the mentioned baselines are grouped into three directions: volume rendering with classical radiance fields [26], occupancy network [28], and SDF-based (Signed Distance Function) fields [46, 41, 9]. Table 1 summarizes the performance comparisons on each evaluation scenario. Note that all baselines and our ORF

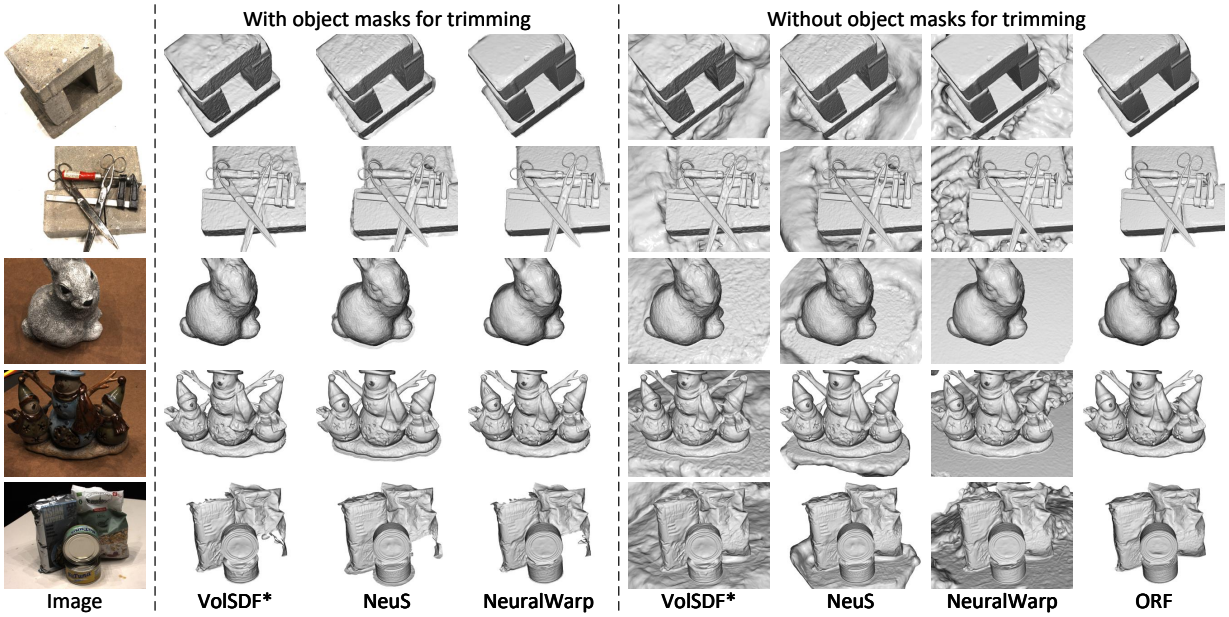

Figure 3: Comparisons on qualitative results of our ORF with other neural radiance fields techniques on DTU dataset for 3D geometry reconstruction with or without object masks at inference. * denotes our implementation of VolSDF.

do not use human-annotated object masks of scenes to optimize geometry/radiance nets during training. At inference, these baselines commonly require masks to remove the background of the learnt mesh. Instead, ORF can directly produce the object mesh without masks.

In general, our ORF exhibits better reconstruction quality against all these baselines. Remarkably, ORF achieves 0.65 in the metric of Chamfer distance without the use of the visual hull derived from object masks in scenes, which is better than the reconstruction accuracy (0.68) of NeuralWarp when exploiting demands object masks at inference. The results generally highlight the key advantage of learning object-aware geometry reconstruction for neural implicit surfaces. Specifically, under the evaluation scenario with object masks, NeRF produces unsatisfactory reconstruction results since the lack of 3D geometry constraints. Unisurf unifies volume/surface rendering and enables better results. VolSDF, NeuS, and NeuralWarp exploit SDF and obtain promising reconstructions. Nevertheless, a performance drop is observed for each SDF-based method when object masks are unavailable at inference, since the primary estimated mesh contains more trivial geometry of the background. In contrast, our ORF achieves competitive reconstruction accuracy through learning an object-aware radiance field without the use of object masks derived visual hull. This confirms the effectiveness of exploring object awareness via transparency discriminator and rasterization-based aggregation.

Qualitative Results. We then visually examine the reconstruction quality of our proposal by comparing ORF with three SDF-based approaches (VolSDF, NeuS, NeuralWarp) on five selected scenes from DTU dataset. Figure 3 shows the qualitative results of the reconstructed meshes. Note that here we show the reconstruction results of the two different evaluation settings (with or without object masks at inference for trimming estimated meshes). In general, all the three approaches reconstruct high-fidelity object surfaces after mask-dependent mesh trimming. In between, by constraining volume rendering with photo-consistency objective across multiple views, NeuralWarp produces more accurate surfaces with high-fidelity details (e.g., the bricks with sparse textures in the first two scenes) against VolSDF and NeuS. Furthermore, when the object masks are unavailable, the output meshes of each baselines become noisy with more trivial geometry and even some artifact of the background. In contrast, under this challenging setting, our ORF performs visually on par with the NeuralWarp, while requiring no object mask dependent mesh trimming at inference. The results again validate the merit of our ORF.

4.3. Experimental Analysis

Ablation Study. Here we investigate how each design in our ORF influences the overall reconstruction performance. Table 2 and Figure 4 details the performances and the corresponding qualitative results across different ablated runs of ORF, respectively. We start from a basic SDF-based radiance field without ray discrimination (i.e., NeuralWarp), which achieves 1.10 of Chamfer distance. The estimated mesh of this basic model inevitably contains trivial background geometry. Next, by solely using the outputs of salient object segmentation (**Saliency**) to distinguish object-intersected and object-bypassed rays, we observe a clear performance boost and the background is effectively screened. However, the independently estimated object probability of each 2D object region may be inaccurate and inconsistent across different viewpoints, resulting in error

Table 2: Ablation study on each design in ORF on DTU dataset.

Transparency Discriminator			Rasterization-based	Chamfer
Saliency	+Multiview	+Verifier	Aggregation	Distance
				1.10
✓				0.90
✓	✓			0.87
✓	✓	✓		0.76
✓	✓	✓	✓	0.65

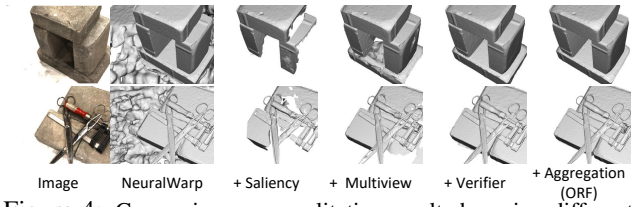

Figure 4: Comparisons on qualitative results by using different ablated runs of our ORF on DTU dataset for 3D geometry reconstruction object masks dependent mesh trimming at inference.

accumulation and holes in the reconstructions. To address this issue, we enhance the strategy of transparency discrimination by jointly exploiting the 2D object regions estimated from multiple views (**Multiview**), leading to slightly better reconstruction. We then leverage the projection verification strategy (**Verifier**) to screen out invalid projections (e.g., the points are projected outside of the source images or occluded by other surfaces), which further boosts up the reconstruction. Finally, by capitalizing on **Rasterization-based Aggregation** to refine the estimated 2D object regions, the mIoU of object regions is improved from 94% to 97% and ORF achieves the highest reconstruction quality.

Effect of the Trade-off Parameter λ_2. To clarify the effect of the trade-off parameter λ_2 in Eq. (9), we detail the reconstruction performances (*i.e.*, Chamfer distances) with different trade-off parameters in Table 3. In the extreme case of $\lambda_2 = 0$, no occupancy regularization objective for foreground and background rays is utilized and the whole model degenerates to the conventional SDF-based approach. When enlarging λ_2 as 0.05, a better reconstruction accuracy is attained, which basically demonstrates that the foreground-aware inductive bias benefits volume rendering. The performance change is relatively smooth when λ_2 varies in the range from 0.05 to 0.20, and the best reconstruction accuracy is attained when $\lambda_2 = 0.10$. Furthermore, the reconstruction accuracy slightly decreases when $\lambda_2 \geq 0.30$. We speculate that this may be the result of amplifying the noise of ray discrimination if we set a higher weight to occupancy regularization objective.

Effect of the Saliency Model's Performance. To verify how the quality of saliency influences the reconstruction, we perform salient object segmentation using another two models and obtain additional two versions of saliency maps. We evaluate the three different versions of saliency for both reconstruction and segmentation tasks on DTU (reconstruction: 0.65, 0.67, 0.67; segmentation: 94%, 93%, 91% mean IoU). The results show that the saliency segmentation performance on DTU has almost saturated, and the reconstruction performances of ORF only fluctuate within the range of 0.02, which eases the difficulty on choosing saliency segmentation model in practice.

Table 3: Effect of the trade-off parameter λ_2 in ORF on DTU.

λ_2	0.00	0.05	0.10	0.20	0.30	0.40	0.50
Chamfer Dist.	1.10	0.66	0.65	0.67	0.69	0.70	0.71

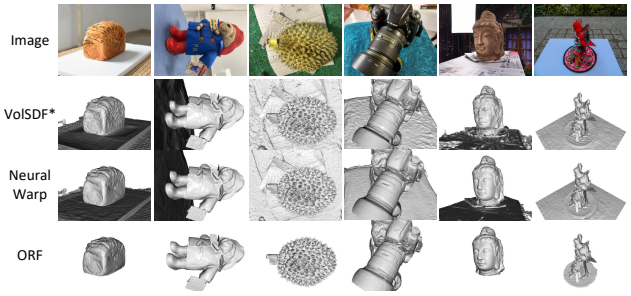

Figure 5: Comparisons on qualitative results of our ORF with other techniques on BlendedMVS for 3D geometry reconstruction without object masks at inference. * denotes our re-implemented VolSDF without the hand-crafted parametrization in NeRF++.

4.4. Comparison on BlendedMVS Dataset

Figure 5 further showcases the geometry reconstruction results with different techniques for six scenes in BlendedMVS. It is worthy to note that VolSDF originally utilizes an additional hand-crafted parametrization as in NeRF++ [50] to tackle the complex backgrounds by modeling the volume outside a radius 3 sphere with another NeRF network. For fair comparison, here we re-implement VolSDF by removing this parametrization. Similar to the observations on DTU under the setting without object masks, both VolSDF and NeuralWarp produce high-fidelity object surfaces, while more trivial background geometry is inevitably included. Instead, our ORF nicely removes the trivial geometry and performs nearly on par with these approaches.

5. Conclusions

In this work, we circumvent the use of object masks derived visual hull, and shape a new paradigm of learning object-aware geometry reconstruction for neural implicit surfaces. To verify our claim, we novelly remould the classical neural radiance fields by involving a new transparency discriminator to distinguish object-intersected and object-bypassed rays on-the-fly. Such ray discrimination serves as object-aware inductive bias to enable a reconstruction of 3D geometry of the key object in the scene. Moreover, we exploit the geometric correspondence between multi-view 2D object regions and 3D implicit/explicit surfaces to facilitate the learning of object surfaces. Extensive experiments conducted on DTU and BlendedMVS validate our proposal.

References

[1] Motilal Agrawal and Larry S Davis. A probabilistic framework for surface reconstruction from multiple images. In *CVPR*, 2001.

[2] Matan Atzmon and Yaron Lipman. Sal: Sign agnostic learning of shapes from raw data. In *CVPR*, 2020.

[3] Connelly Barnes, Eli Shechtman, Adam Finkelstein, and Dan B Goldman. Patchmatch: A randomized correspondence algorithm for structural image editing. *ACM Trans. on Graphics*, 2009.

[4] Michael Bleyer, Christoph Rhemann, and Carsten Rother. Patchmatch stereo-stereo matching with slanted support windows. In *BMVC*, 2011.

[5] Adrian Broadhurst, Tom W Drummond, and Roberto Cipolla. A probabilistic framework for space carving. In *ICCV*, 2001.

[6] Jingwen Chen, Yiheng Zhang, Zhongwei Zhang, Yingwei Pan, and Ting Yao. 3d-producer: A hybrid and user-friendly 3d reconstruction system. In *CAAI International Conference on Artificial Intelligence*, 2022.

[7] Yang Chen, Yingwei Pan, Yehao Li, Ting Yao, and Tao Mei. Control3d: Towards controllable text-to-3d generation. In *ACM MM*, 2023.

[8] Brian Curless and Marc Levoy. A volumetric method for building complex models from range images. In *SIGGRAPH*, 1996.

[9] François Darmon, Bénédicte Bascle, Jean-Clément Devaux, Pascal Monasse, and Mathieu Aubry. Improving neural implicit surfaces geometry with patch warping. In *CVPR*, 2022.

[10] Jeremy S De Bonet and Paul Viola. Poxels: Probabilistic voxelized volume reconstruction. In *ICCV*, 1999.

[11] Simon Donne and Andreas Geiger. Learning non-volumetric depth fusion using successive reprojections. In *CVPR*, 2019.

[12] Yasutaka Furukawa and Jean Ponce. Accurate, dense, and robust multiview stereopsis. *IEEE Trans. on PAMI*, 2009.

[13] Amos Gropp, Lior Yariv, Niv Haim, Matan Atzmon, and Yaron Lipman. Implicit geometric regularization for learning shapes. In *ICML*, 2020.

[14] Wilfried Hartmann, Silvano Galliani, Michal Havlena, Luc Van Gool, and Konrad Schindler. Learned multi-patch similarity. In *ICCV*, 2017.

[15] Po-Han Huang, Kevin Matzen, Johannes Kopf, Narendra Ahuja, and Jia-Bin Huang. Deepmvs: Learning multi-view stereopsis. In *CVPR*, 2018.

[16] Rasmus Jensen, Anders Dahl, George Vogiatzis, Engin Tola, and Henrik Aanæs. Large scale multi-view stereopsis evaluation. In *CVPR*, 2014.

[17] Michael Kazhdan, Matthew Bolitho, and Hugues Hoppe. Poisson surface reconstruction. In *Proceedings of the fourth Eurographics symposium on Geometry processing*, 2006.

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

[19] Sosuke Kobayashi, Eiichi Matsumoto, and Vincent Sitzmann. Decomposing nerf for editing via feature field distillation. In *NeurIPS*, 2022.

[20] Kiriakos N Kutulakos and Steven M Seitz. A theory of shape by space carving. *IJCV*, 2000.

[21] Vincent Leroy, Jean-Sébastien Franco, and Edmond Boyer. Shape reconstruction using volume sweeping and learned photoconsistency. In *ECCV*, 2018.

[22] William E. Lorensen and Harvey E. Cline. Marching cubes: A high resolution 3d surface construction algorithm. In *SIGGRAPH*, 1987.

[23] Wenjie Luo, Alexander G Schwing, and Raquel Urtasun. Efficient deep learning for stereo matching. In *CVPR*, 2016.

[24] Paul Merrell, Amir Akbarzadeh, Liang Wang, Philippos Mordohai, Jan-Michael Frahm, Ruigang Yang, David Nistér, and Marc Pollefeys. Real-time visibility-based fusion of depth maps. In *ICCV*, 2007.

[25] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *CVPR*, 2019.

[26] 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 *ECCV*, 2020.

[27] Michael Niemeyer, Lars Mescheder, Michael Oechsle, and Andreas Geiger. Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision. In *CVPR*, 2020.

[28] Michael Oechsle, Songyou Peng, and Andreas Geiger. Unisurf: Unifying neural implicit surfaces and radiance fields for multi-view reconstruction. In *ICCV*, 2021.

[29] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *CVPR*, 2019.

[30] Despoina Paschalidou, Osman Ulusoy, Carolin Schmitt, Luc Van Gool, and Andreas Geiger. Raynet: Learning volumetric 3d reconstruction with ray potentials. In *CVPR*, 2018.

[31] Songyou Peng, Michael Niemeyer, Lars Mescheder, Marc Pollefeys, and Andreas Geiger. Convolutional occupancy networks. In *ECCV*, 2020.

[32] Xuebin Qin, Zichen Zhang, Chenyang Huang, Masood Dehghan, Osmar R Zaiane, and Martin Jagersand. U2-net: Going deeper with nested u-structure for salient object detection. *Pattern Recognition*, 106:107404, 2020.

[33] Gernot Riegler, Ali O. Ulusoy, Horst Bischof, and Andreas Geiger. Octnetfusion: Learning depth fusion from data. In *3DV*, 2017.

[34] Johannes L Schönberger, Enliang Zheng, Jan-Michael Frahm, and Marc Pollefeys. Pixelwise view selection for unstructured multi-view stereo. In *ECCV*, 2016.

[35] Steven M Seitz, Brian Curless, James Diebel, Daniel Scharstein, and Richard Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *CVPR*, 2006.

[36] Steven M Seitz and Charles R Dyer. Photorealistic scene reconstruction by voxel coloring. *IJCV*, 1999.

[37] P. Shirley, M. Ashikhmin, and S. Marschner. *Fundamentals of Computer Graphics*. Ak Peters Series. Taylor & Francis, 2005.

[38] Shubham Tulsiani, Tinghui Zhou, Alexei A Efros, and Jitendra Malik. Multi-view supervision for single-view reconstruction via differentiable ray consistency. In *CVPR*, 2017.

[39] Ali O. Ulusoy, Andreas Geiger, and Michael J. Black. Towards probabilistic volumetric reconstruction using ray potentials. In *3DV*, 2015.

[40] Benjamin Ummenhofer, Huizhong Zhou, Jonas Uhrig, Nikolaus Mayer, Eddy Ilg, Alexey Dosovitskiy, and Thomas Brox. Demon: Depth and motion network for learning monocular stereo. In *CVPR*, 2017.

[41] 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. In *NeurIPS*, 2021.

[42] Haibo Yang, Yang Chen, Yingwei Pan, Ting Yao, Zhineng Chen, and Tao Mei. 3dstyle-diffusion: Pursuing fine-grained text-driven 3d stylization with 2d diffusion models. In *ACM MM*, 2023.

[43] Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. Mvsnet: Depth inference for unstructured multi-view stereo. In *ECCV*, 2018.

[44] Yao Yao, Zixin Luo, Shiwei Li, Tianwei Shen, Tian Fang, and Long Quan. Recurrent mvsnet for high-resolution multi-view stereo depth inference. In *CVPR*, 2019.

[45] 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 *CVPR*, 2020.

[46] Lior Yariv, Jiatao Gu, Yoni Kasten, and Yaron Lipman. Volume rendering of neural implicit surfaces. In *NeurIPS*, 2021.

[47] Lior Yariv, Yoni Kasten, Dror Moran, Meirav Galun, Matan Atzmon, Basri Ronen, and Yaron Lipman. Multiview neural surface reconstruction by disentangling geometry and appearance. In *NeurIPS*, 2020.

[48] Sergey Zagoruyko and Nikos Komodakis. Learning to compare image patches via convolutional neural networks. In *CVPR*, 2015.

[49] Jingyang Zhang, Yao Yao, and Long Quan. Learning signed distance field for multi-view surface reconstruction. In *ICCV*, 2021.

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

[51] Zicheng Zhang, Yinglu Liu, Congying Han, Yingwei Pan, Tiande Guo, and Ting Yao. Transforming radiance field with lipschitz network for photorealistic 3d scene stylization. In *CVPR*, 2023.