JCAD: 103604

ARTICLE IN PRESS

Computer-Aided Design xxx (xxxx) xxx

Contents lists available at ScienceDirect

Computer-Aided Design

journal homepage: www.elsevier.com/locate/cad

DiffSVR: Differentiable Neural Implicit Surface Rendering for Single-View Reconstruction with Highly Sparse Depth Prior

Artem Komarichev, Jing Hua, Zichun Zhong *

Department of Computer Science, Wayne State University, Detroit, 48202, MI, USA

ARTICLE INFO

Article history:
Received 26 May 2023
Received in revised form 20 July 2023
Accepted 22 July 2023

Keywords:
Depth-aware occupancy function
Differentiable surface rendering
Single-view reconstruction

ABSTRACT

It is well-known that 3D shape and texture reconstruction from a single-view image is a very challenging and an ill-posed problem, especially without 3D supervision/ground truth. Existing neural implicit surface reconstruction approaches do easily get trapped in the local minima and cannot produce high-fidelity geometry and high-quality textures (and rendered images) under single-view setting, even with provided highly sparse depth prior. In this paper, we propose a new self-supervised learning method *DiffSVR* that represents a complicated surface as a new *depth-aware occupancy function* (DOF) and utilizes an end-to-end differentiable surface rendering paradigm to optimize the neural DOF field relying only on single-view image with highly sparse depth information. The developed *surface-aware sampling*, *occupancy self-labeling*, and *differentiable surface rendering with inverse computation* techniques can enhance both the neural implicit surface reconstruction and the neural renderer. The extensive experiments and comparisons on two real-world benchmark datasets (e.g., DTU and KITTI) demonstrate that our approach not only numerically outperforms the current state-of-the-art methods by a large margin, but also produces surface mesh model with qualitatively better geometric details and more accurate textures, as well as exhibits good performance on generalizability and flexibility. The code and data are available at https://github.com/akomarichev/DiffSVR.

1. Introduction

Reconstructing and learning 3D shapes from 2D images is a fundamental problem in computer vision for decades [1–9]. In the recent years, there are many learning-based single-view shape reconstruction methods to address this challenging and ill-pose problem, such as 3D-R2N2 [10], PSGN [11], Pixel2Mesh [12], Image2Mesh [13], 3D FFD [14], 3D Face [4], 3D Hand [15], DeepOrganNet [16], 3DSVPC [17], etc. However, all of them either need 3D supervision or can only work with simple 3D geometry shapes. With the recent advances in neural implicit representations with differentiable and continuous formulation, such as signed distance functions (SDF) [18], occupancy [19], and their variants, the research in this direction has become very promising. They are flexible to represent 3D surfaces with complex geometry and arbitrary topologies, without pre-defining a volumetric or a mesh template.

Recently, leveraging differentiable neural rendering with implicit representation, DVR [20] tries to push the limit of single-view 3D reconstruction problem to the case without 3D supervision. It can deal with single-view reconstruction along with dense depth map, and optimize the implicit geometry and the texture by minimizing the difference of rendered views and ground truth views. However, DVR utilizes depths only in their loss function and lacks an analytic implicit model for effectively sampling the occupancy field to represent a complicated surface with occlusions, so that it will easily get trapped in the local minima under the limited-view setting. Besides that, there are some multi-view neural surface reconstruction methods [21–24]. These methods either heavily depend on a large number of input views (dense views), which are not able to adapt to single-view input, or are based on volume rendering, heavily depending on dense sampling points along the rays, which are computationally expensive. Some examples are shown in Fig. 1.

In reality, it is often difficult or costly to obtain the dense depth map or dense multi-view images. Therefore, high-fidelity reconstruction with sparse depth map and single- / limited-view RGB image constitutes practical solution for real-world applications. For instance, it is more practical to develop a novel RGB-D camera with a high-resolution RGB image along with a low-resolution/sparse depth map for a superbly fast and lightweight 3D reconstruction in robotics or autonomous driving systems. Similarly, a LiDAR-based depth map is always very sparse and the problem of effectively fusing sparse depth map with the dense RGB image in the 3D reconstruction system is promising and useful.

* Corresponding author.
E-mail address: zichunzhong@wayne.edu (Z. Zhong).

https://doi.org/10.1016/j.cad.2023.103604
0010-4485/

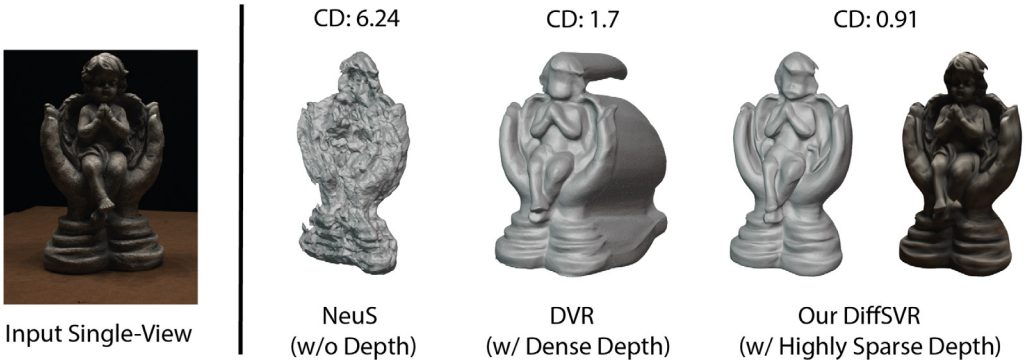

Fig. 1. The illustration of our DiffSVR and other state-of-the-art methods with single-view input. Our approach can reconstruct better quality shape and texture.

In this work, we propose *DiffSVR*, a novel single-view surface reconstruction method with two significant advantages: (1) it needs only a single-view image with highly sparse depth information for successful 3D reconstruction; (2) it significantly improves the capability of the differentiable surface rendering-based reconstruction to next level allowing for handling of complex structures and textures as shown in Fig. 1. The key *contributions* of our work are:

- We propose a new self-supervised learning method that represents the complicated surface as a novel analytic implicit model *depth-aware occupancy function* (DOF) and adopts an *end-to-end differentiable surface rendering paradigm* to train the neural DOF representation only relying on single-view image with highly sparse depth;
- The proposed DOF includes *surface-aware sampling* and *occupancy self-labeling* components, which lead to better capturing and representing shape geometric and topological details at the implicit representation level. The developed differentiable surface rendering with inverse computation can further enhance the optimization of both the neural implicit surface and texture;
- The extensive experiments and comparisons demonstrate that our approach not only numerically achieves new state-of-the-art performance, but also produces surface mesh with qualitatively better geometric details and more accurate textures, as well as exhibits good performance on generalizability and flexibility on two large-scale real-world benchmark datasets (e.g., DTU and KITTI).

2. Related work

In this section, we provide the overview of the two main directions in 3D surface reconstruction, i.e., classical 3D surface reconstruction methods utilizing multi-view images and recent 3D surface reconstruction methods through learning neural implicit representations.

Classical Multi-View Surface Reconstruction. 3D surface reconstruction from a given set of multi-view images is a classical problem in computer vision and computer graphics fields [25]. Traditional multi-view stereo (MVS) methods focus on estimating the depth of each pixel by matching features points across different views [26–29]. This category of methods requires additional post-processing step of depth map fusion into a dense point cloud [27,30–32] and followed by Poisson surface reconstruction meshing [33,34]. The quality of the reconstructed surface heavily relies on the quality of the features matching among corresponding views and suffers from artifacts and missing parts in the reconstructed surface. Another category of traditional MVS methods relies on volumetric reconstruction [5,35–41] by estimating occupancies of the voxel grid from multi-view images and suffers from low-quality reconstruction with high computational cost.

Neural Implicit Surface Reconstruction. Recently, the neural implicit approaches [19,20,22,42–49] have emerged to overcome problems of the traditional MVS methods and learn better quality 3D geometry and appearance with higher efficiency. These methods allow to learn implicit surface through optimizing network parameters according to the implicit formulations and do not suffer from discretization artifacts or low quality in the reconstructed surface due to its continuous nature. We divide these methods into two main categories: implicit surface rendering-based reconstruction and implicit volume rendering-based reconstruction.

Recent neural implicit surface rendering-based methods [20,21] propose a differentiable surface rendering formulation with implicit gradient calculation to optimize neural implicit surface from multi-view images. DVR [20] introduces a differentiable rendering method for implicit surface and appearance representations by calculating analytically depth gradients. DVR only uses depth in their loss function and lacks an analytic implicit model for effectively sampling the occupancy field to represent a complicated surface. IDR [21] introduces the implicit neural network to learn 3D geometry and neural renderer to approximate texture with appropriate surface lighting conditioned on the viewing direction. However, both IDR and DVR fail to correctly reconstruct accurate 3D surface and render high-quality images under limited-view setting (only consider a single surface point for each ray), not to mention the very challenging single-view case.

Volume rendering-based reconstruction approaches, such as NeRF [50] or follow-up works [51–61], propose the differentiable volume rendering techniques to optimize neural α-compositing radiance fields along the ray as well as some variants for dynamic scene rendering, fast inference, sparse-view input, etc. These methods show impressive performance for synthesizing high-quality novel images, but produce unsatisfactory low-quality 3D geometry.

Alternatively, there are some recent methods that try to combine advantages of implicit surface-based and volumetric approaches together in one model. Oechsle et al. [22] proposed an approach to unify both surface rendering and volume rendering with the sampling strategy. Wang et al. [23] proposed a volume rendering scheme to learn signed distance function (SDF) that represents a scene space as a density-based function. One of the most recent works proposed by Long et al. [24] presents a neural volume rendering based method for the task of surface reconstruction from sparse-view images. However, these methods, similar to DVR and IDR approaches, have low-quality reconstruction results on the single-view image.

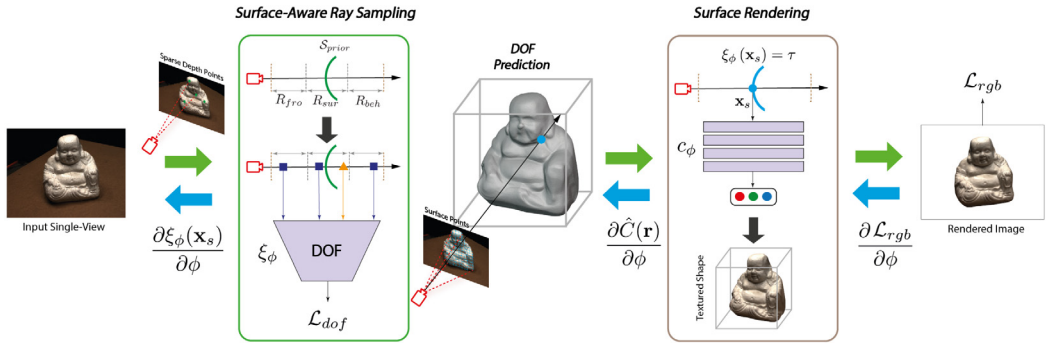

Fig. 2. The architecture of *DiffSVR* on a single-view image. Firstly, we extract $\mathcal{S}_{prior}$ from sparse/dense depth points that guide our *surface-aware ray sampling* and *occupancy self-labeling* to construct more effective DOF. Secondly, for each point of the predicted object surface from DOF we compute our *differentiable surface-rendering* to obtain appearance. Finally, our proposed differentiable backpropogation can enhance both the neural implicit surface reconstruction and the neural renderer. Evaluation of $\mathcal{L}_{rgb}$ is performed within predicted object surface mask during DOF prediction step.

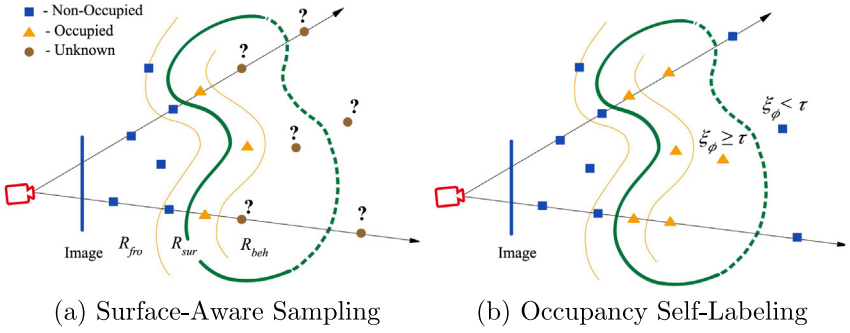

Fig. 3. The illustration of the proposed *surface-aware sampling* and *occupancy self-labeling* in our DOF.

3. Method

The motivation of this work is that given a single-view image (with highly sparse depth information) of an object, we reconstruct the 3D surface with its appearance. Due to the limited information, it is very critical to design an effective neural surface representation and renderer scheme to facilitate the task. Volume rendering-based surface reconstruction method requires very dense points along the ray. Moreover, it always requires extra effort to represent and extract the implicit surface from the 3D volume space. In this DiffSVR work, we investigate into the new neural geometry-aware surface rendering techniques to enhance the implicit models for the proposed neural surface and texture representations built from a single-view input only. The main technical components of our DiffSVR are (1) an analytic implicit model *depth-aware occupancy function (DOF)* with *surface-aware ray sampling* and *occupancy self-labeling*, and (2) *differentiable neural surface rendering for self-supervised learning*. In our proposed model, an object to be reconstructed is represented by implicit functions w.r.t. geometry and texture. The overview of our proposed method is shown in Fig. 2.

3.1. Depth-Aware Occupancy Function (DOF)

Occupancy Networks [19,20] define *occupancy function* f_ϕ that for every possible point $\mathbf{x}_v \in \mathbb{R}^3$ in a 3D space assigns binary occupancy probability between 0 and 1:

$$f_\phi(\mathbf{x}_v) : \mathbb{R}^3 \rightarrow [0, 1], \quad (1)$$

where f_ϕ represents a neural network with parameters ϕ. The implicit surface $\mathcal{S}$ of the 3D object is defined as the set of points (i.e., the decision boundary of the binary occupancy classifier), where occupancy probabilities equal to a given threshold: $\mathcal{S} = \left\{\mathbf{x}_s \in \mathbb{R}^3 \mid f_\phi(\mathbf{x}_s) = \tau\right\}$, where a threshold parameter $\tau \in [0, 1]$ defines 3D surface of an object implicitly.

As we know, depth information is very useful in 3D shape geometry reconstruction, especially in our challenging single-view task. Inspired by this, we introduce *depth-aware occupancy function (DOF)* ξ_ϕ by taking advantage of depth prior information, which is not considered in the original occupancy function definition. Given sparse/dense depth prior **D** from the single-view, obtained by a depth camera or estimated by MVS algorithms [27], we can represent *surface prior* $\mathcal{S}_{prior}$ as a set of surface points $\mathbf{x}_s = \mathbf{c}_o + d_s\mathbf{r}_v$, where $\mathbf{c}_o$ is the origin of the camera, d_s is the provided depth, and $\mathbf{r}_v$ is the ray view direction for a given pixel. Once we define the surface prior, we can sample points and assign labels along the ray view direction based on the occupancy function. In the following, we introduce two main components of the proposed DOF: *surface-aware ray sampling* and *occupancy self-labeling*.

Surface-Aware Ray Sampling. The essential effectiveness and efficiency of the 3D reconstruction depend on how to sample the points on the ray. Given a ray, compared with the existing state-of-the-art methods, e.g, DVR [20], IDR [21], UNISURF [22], NeuS [23], SparseNeuS [24] with equally-spaced or empirically-hierarchical sampling points on the ray, our method can adaptively and efficiently sample limited number of points along the ray with the surface-aware mechanism to cover the corresponding labeling values in the occupancy field. Moreover, none of the above surface reconstruction methods have taken use of the depth prior information in their methodology. We will demonstrate DOF's effectiveness in the experiment and ablation study.

In DOF, based on the above surface prior $\mathcal{S}_{prior}$, we divide the unit cube with the object inside into three main regions R_{fro}, R_{sur}, and R_{beh} as shown in Fig. 3(a). R_{fro} represents the region where all

possible points lie in front of the object's surface. R_{sur} represents the region where all possible points locate very close to the object's surface on both sides. R_{beh} represents the region where all possible points locate behind the object's surface, which is used for occlusion inference. More details are given in the following subsection. Additionally, we define d_{min} and d_{max} as the unit box boundaries. Based on this surface-awareness strategy, we only need very few number of sampling points in each region to define an effective occupancy field. Our experiments will justify this nice property and advantages. On the contrary, the previous methods always need very dense sampling points on each ray in order to preserve the accuracy of the surface reconstruction.

More specifically, let us consider the random ray that penetrates unit cube where the object is located. Our goal is to sample points for each ray in aforementioned three regions respectively. In order to realize this, we first sample depths within those regions and then transform depths into 3D space. For R_{fro} we sample N_{fro} depth values $\mathcal{D}_{fro}$ from the interval $[d_{min}, d_s)$ as follows:

$$\mathcal{D}_{fro} \sim \left[d_s - (d_s - d_{min})\mathcal{U}\left(N_{fro}\right)\right], \tag{2}$$

where $\mathcal{U}(\cdot)$ is sampled from a uniform distribution on the interval $[0, 1)$, and d_s is a given or estimated surface depth value. Then, for R_{sur} we sample N_{sur} depth values $\mathcal{D}_{sur}$ from the interval $(d_s - \sigma, d_s + \sigma)$ centered around d_s:

$$\mathcal{D}_{sur} \sim \left[d_s - \sigma\mathcal{U}\left(\frac{N_{sur}}{2}\right), d_s + \sigma\mathcal{U}\left(\frac{N_{sur}}{2}\right)\right], \tag{3}$$

where σ represents the small value for sampled depths lying close to the given or estimated depth d_s. Finally, we sample N_{beh} depths $\mathcal{D}_{beh}$ within the interval $(d_s, d_{max}]$:

$$\mathcal{D}_{beh} \sim \left[d_s + (d_{max} - d_s)\mathcal{U}\left(N_{beh}\right)\right]. \tag{4}$$

After that we transform the sampled depth values into 3D points for three regions as follows:

$$\begin{aligned} \mathbf{X}_{fro} &= \mathbf{c}_o + \mathcal{D}_{fro}\mathbf{r}_v, \\ \mathbf{X}_{sur} &= \mathbf{c}_o + \mathcal{D}_{sur}\mathbf{r}_v, \\ \mathbf{X}_{beh} &= \mathbf{c}_o + \mathcal{D}_{beh}\mathbf{r}_v. \end{aligned} \tag{5}$$

Based on Eq. (5), we can effectively and efficiently sample points for occupancy function to better capture the occupancy probabilities of the 3D object within the unit cube.

Occupancy Self-Labeling. As one can notice that it is straightforward to assign occupancy label to sampled points in R_{fro}, R_{sur} regions based on the single-view depth information. Any sampled points in front of the surface of the object have non-occupancy labels with zero values, and are defined as $\mathcal{O}_{fro}$. The points sampled around the surface are assigned occupancy labels $\mathcal{O}_{sur}$ depending on which side of the surface the sampled points fall in. However, to label points sampled behind the surface is not straightforward, because with the provided single-view image(s), we cannot see through the object. In this case, we propose an *occupancy self-labeling* method as shown in Fig. 3(b), where we assign the occupancy label on-the-fly to any point in R_{beh} region by evaluating occupancy values by the current optimized DOF ξ_ϕ as follows, so as to better predict the occupancy labels for the occluded parts:

$$\begin{aligned} \mathcal{O}_{beh} \in \text{occupied} &= \left\{\mathbf{X}_{beh} \in \mathbb{R}^3 \mid \xi_\phi\left(\mathbf{X}_{beh}\right) \geq \tau\right\}, \\ \mathcal{O}_{beh} \in \text{non-occupied} &= \left\{\mathbf{X}_{beh} \in \mathbb{R}^3 \mid \xi_\phi\left(\mathbf{X}_{beh}\right) < \tau\right\}. \end{aligned} \tag{6}$$

After that, we can optimize parameters ϕ of our network ξ_ϕ according to the sampled points and their corresponding occupancy values.

DOF Definition. Our idea of the constrained surface-aware ray sampling alongside with the occupancy self-labeling improves the implicit representation of the object by wisely sampling points along the rays and assigning occupancy labels to them on-the-fly. Therefore, our complete DOF is defined as the following:

$$\begin{aligned} &\xi_\phi(\mathbf{x}_v) : \mathbb{R}^3 \rightarrow [0, 1] : \\ &\mathbf{x}_v \in \left\{\mathbf{X}_{fro}, \mathbf{X}_{sur}, \mathbf{X}_{beh}\right\} \rightarrow o_{\mathbf{x}_v} \in \left\{\mathbf{O}_{fro}, \mathbf{O}_{sur}, \mathbf{O}_{beh}\right\}. \end{aligned} \tag{7}$$

3.2. End-to-end differentiable surface rendering

In order to realize self-supervised learning from 2D single-view image, we design the inverse computation by backpropagating the differentiation/gradients from the color loss between the rendered images and the input images to the differentiation/gradients of the implicit DOF, so as to synergically optimize and reconstruct 3D object geometry and texture in an end-to-end way. To learn the color of every point $\mathbf{x}_s$ of the predicted surface $\mathcal{S}$ in a 3D space, we propose a *color function*: $c_\phi(\mathbf{x}_s) : \mathbb{R}^3 \rightarrow \mathbb{R}^3$. We optimize the color function c_ϕ (i.e., represented as a neural network with parameters ϕ) during training together with the DOF ξ_ϕ as a simple regression task. However, as reported by Oechsle et al. [44], learning color generation in this way remains an ill-posed problem and needs additional constraints. Thus, in order to provide more surface information at the given surface point $\mathbf{x}_s \in \mathbb{R}^3$ we estimate normal $\mathbf{n}_\phi(\mathbf{x}_s) \in \mathcal{N}$ and extract DOF surface embeddings $\xi_\phi(\mathbf{x}_s) \in \mathcal{F}$ from the last fully-connected layer of the DOF network, and then the *surface-aware color function* becomes:

$$c_\phi(\mathbf{x}_s, \xi_\phi(\mathbf{x}_s), \mathbf{n}_\phi(\mathbf{x}_s)) : \mathbb{R}^3 \times \mathcal{F} \times \mathcal{N} \rightarrow \mathbb{R}^3. \tag{8}$$

The normal vector $\mathbf{n}_\phi(\mathbf{x}_s)$ is a normalized gradient of the DOF neural network ξ_ϕ as $\mathbf{n}_\phi(\mathbf{x}_s) = \frac{\nabla_{\mathbf{x}_s}\xi_\phi(\mathbf{x}_s)}{\|\nabla_{\mathbf{x}_s}\xi_\phi(\mathbf{x}_s)\|_2}$.

In order to find a surface point at the given ray $\mathbf{r}$, we evaluate DOF network ξ_ϕ on n equally-sampled points along that ray to detect the very first change from non-occupied to occupied value. Once we find such interval, we run secant method [20] along the ray to extract surface point $\mathbf{x}_s$.

Our main goal in this work is to learn the implicit surface and texture field from single-view 2D images. Based on the DOF representation in the previous section, we introduce the single-view RGB reconstruction loss $\mathcal{L}_{rgb}$:

$$\mathcal{L}_{rgb} = \sum_{\mathbf{r}} \left\| \hat{C}(\mathbf{r}) - C(\mathbf{r}) \right\|_1, \tag{9}$$

where $\| \cdot \|_1$ is ℓ_1-loss, $\hat{C}(\mathbf{r})$ is a rendered image for pixel/ray $\mathbf{r}$ from implicit DOF learned by our model i.e., $c_\phi(\mathbf{x}_s, \xi_\phi(\mathbf{x}_s), \mathbf{n}_\phi(\mathbf{x}_s))$, and $C(\mathbf{r})$ represents ground truth image (input image).

In order to effectively minimize RGB loss and learn parameters of our networks, we need to compute gradients explicitly. To obtain gradients of $\mathcal{L}_{rgb}$ (i.e. $\frac{\partial \mathcal{L}_{rgb}}{\partial \phi}$) w.r.t. the network parameters ϕ, i.e., differentiating $c_\phi(\mathbf{x}_s)$, we inspire the gradient computation idea from [20]:

$$\frac{\partial \hat{C}(\mathbf{r})}{\partial \phi} = \frac{\partial c_\phi(\mathbf{x}_s)}{\partial \phi} + \frac{\partial c_\phi(\mathbf{x}_s)}{\partial \mathbf{x}_s} \cdot \frac{\partial \mathbf{x}_s}{\partial \phi}. \tag{10}$$

To calculate $\frac{\partial \mathbf{x}_s}{\partial \phi}$ related to implicit DOF field is not straightforward. Instead, we can calculate gradient of the surface depth w.r.t. the ϕ parameters using implicit differentiation. We notice that $\frac{\partial \mathbf{x}_s}{\partial \phi} = \mathbf{r}_v \frac{\partial d_s}{\partial \phi}$, where $\mathbf{r}_v$ is a ray direction. Then, differentiating the DOF $\xi_\phi(\mathbf{x}_s)$, we have: $\frac{\partial d_s}{\partial \phi} = -\left(\frac{\partial \xi_\phi(\mathbf{x}_s)}{\partial \mathbf{x}_s} \cdot \mathbf{r}_v\right)^{-1} \frac{\partial \xi_\phi(\mathbf{x}_s)}{\partial \phi}$.

Finally, Eq. (10) can be rewritten as: $\frac{\partial \hat{\mathbf{C}}(\mathbf{r})}{\partial \phi}$

$$= \frac{\partial c_\phi(\mathbf{x}_s)}{\partial \phi} - \frac{\partial c_\phi(\mathbf{x}_s)}{\partial \mathbf{x}_s} \cdot \mathbf{r}_v \left(\frac{\partial \xi_\phi(\mathbf{x}_s)}{\partial \mathbf{x}_s} \cdot \mathbf{r}_v\right)^{-1} \frac{\partial \xi_\phi(\mathbf{x}_s)}{\partial \phi}. \tag{11}$$

We would like to emphasize that we learn and optimize DiffSVR network parameters ϕ of DOF and color functions together. Our neural renderer with its end-to-end differentiable gradient computation, as shown in Eq. (11) (differentiations of color function $\partial \mathbf{c}_\phi$ and DOF $\partial \xi_\phi$), is of great importance on single-view surface and appearance reconstruction which will be demonstrated in the ablation study. Our differentiable idea has been derived from [20], but the main difference is that we develop a differentiable neural renderer originally defined on a general occupancy field to our proposed neural DOF as shown in Eq. (11).

3.3. Loss function

To optimize the parameters ϕ of color network c_ϕ and DOF network ξ_ϕ, we propose a regularized loss function:

$$\mathcal{L} = \mathcal{L}_{rgb} + \lambda \mathcal{L}_{dof}, \tag{12}$$

where λ is a coefficient, $\mathcal{L}_{rgb}$ is a RGB color loss based on the rendered single-view image, and $\mathcal{L}_{dof}$ is a binary classification loss, which is an implicit shape geometric regularization based on the DOF. We set $\lambda = 10$ in our experiments since $\mathcal{L}_{rgb}$ loss is much denser (in sampling) than $\mathcal{L}_{dof}$ loss in the total loss function.

$$\begin{aligned} \mathcal{L}_{rgb} &= \sum_{\mathbf{x}_s \in \mathcal{S}} \| c_\phi(\mathbf{x}_s, \xi_\phi(\mathbf{x}_s), \mathbf{n}_\phi(\mathbf{x}_s)) - C(\mathbf{r}) \|_1, \\ \mathcal{L}_{dof} &= \sum_{\mathbf{x}_v \in \mathbb{R}^3} BCE(\xi_\phi(\mathbf{x}_v), o_{\mathbf{x}_v}), o_{\mathbf{x}_v} \in [0, 1], \end{aligned} \tag{13}$$

where BCE is a binary cross entropy, $o_{\mathbf{x}_v}$ is the occupancy labels $[0, 1]$ from DOF in Eq. (7). Our introduced DOF loss ($\mathcal{L}_{dof}$) well regularizes surface boundary of the object in 3D implicit geometry space, and color loss ($\mathcal{L}_{rgb}$) further optimizes the quality of the surface reconstruction as well as learns color/texture of the 3D object. It is noted that in our model, we do not need any explicit smoothing term, which makes our formulation more effective.

For incomplete/sparse depth prior, when a given ray lies inside the object and depth is not available, we ignore the DOF loss for that ray and apply only RGB loss if the network predicts surface point. The experiments show that our method is robust on the highly sparse depth case.

4. Experiments

4.1. Experimental settings

Datasets. We evaluate our DiffSVR framework on the DTU MVS dataset [62] on a single-view image setting. The real-world challenging DTU dataset contains different scans with 49 or 64 high-resolution images of 1200×1600 together with extrinsic and intrinsic camera parameters for separate views. We test our method on 15 scans, same as [20,21,23,24], with a wide variety of geometry and appearance. We evaluate our model and baselines under single-view setting with provided dense/sparse depth prior. Since DTU dataset does not provide the depth information, in this work we use IDR method [21] to obtain depth prior, and some other multi-view stereo algorithms can be used as well. It is noted that the estimated depth prior for DTU dataset in the experiments are noisy (no ground truth depths are used). In order to better show the practical usage and generalizability of our method, we additionally evaluate our DiffSVR framework on some scans from the large-scale outdoor KITTI dataset [63]. The resolution of the front camera images is cropped to a size of 1242×375 pixels and the resolution of the LiDAR sensor scans is 64-beam. We evaluate our method and DVR under the same single-view setting with the sparse captured depths from raw Velodyne points.

Baselines. We compare our method with the several state-of-the-art approaches under single-view setting, such as neural-based surface reconstruction methods, e.g., IDR [21], NeuS [23], and SparseNeuS [24]. Since DVR [20] uses depth prior in their method as well, we evaluate it on three scans with their provided depth information. In addition, we compare our approach with a classic MVS method COLMAP, where the mesh is reconstructed from generated point cloud with the followed screened Poisson surface reconstruction (sPSR) [34]. The best results in tables are shown in bold font.

Implementation Details. We perform all our experiments on a single NVIDIA GeForce RTX 3090 GPU with batch size of 1 with 2048 randomly selected pixels and train our models for 500 K iterations (about 12 hours in total). For surface extraction step, we start with an equally-spaced ray sampling of 16 sampled points per ray and iteratively increase it up to 128 by doubling each 50K, 150K, and 250K iterations. We set $\tau = 0.5$ in all our experiments. Our network architectures are inspired from DVR [20] and IDR [21]. More details of the network architectures and training details can be found in the Supplementary Material.

4.2. Comparisons

We perform qualitative and quantitative comparisons with recent state-of-the-art methods (without 3D supervision) on DTU dataset. We measure the reconstruction quality of the meshes with the Chamfer distances (CD) calculated as in [21,23]. Additionally, we also use mask IoU metric to measure the quality of the silhouettes of the reconstructed meshes from unseen views between the reconstructed mask and the ground truth one. To evaluate the quality of the image renderings, we report the standard PSNR and SSIM metrics [64]. We also include LPIPS [65], which more accurately reflects human perception. All image metrics have been evaluated inside predicted object mask where the background is masked out. For mask IoU and all image metrics, we perform evaluation only on unseen views. In our main paper, we include two main metrics: CD and PSNR. Additional evaluation metrics (i.e., mask IoU, SSIM, and LPIPS) are reported as well as more visualization results and image renderings are provided in Supplementary Material.

With Single-View Dense Depth Prior. We provide quantitative comparisons of our DiffSVR method (with single-view dense depth prior) with state-of-the-art methods on DTU dataset. We measure CD and PSNR metrics and report results in Table 1. The results show that our method outperforms all of these methods by a large margin on a single-view scenario. We also conduct qualitative comparisons with IDR, NeuS, and DVR. As shown in Fig. 4, our method reconstructs much better quality surface for front and occluded parts of the object than other methods. IDR and NeuS fail to reconstruct good quality surface, where details of the surface are noisy. The main reason is that these methods cannot infer the 3D or 2.5D depth information only from a single-view RGB image in a small dataset. This observation also justifies that it is very important to leverage the depth information or prior in the single-view reconstruction. With the help of the depth information, DVR method can reconstruct comparably good front surface with missing feature details, but fails to reconstruct occluded parts of the object, where the occluded parts have a long 'tail' due to the lack of appropriate modeling of the depth information. However, our method can predict a good quality for the occluded parts due to our analytic implicit model DOF and occupancy self-labeling technique as shown in Fig. 4. Also, mask IoU metric that is reported in Supplementary Material shows that occluded parts are well reconstructed comparing to other methods.

With Single-View Highly Sparse Depth Prior. We also provide quantitative (Table 2) and qualitative (Fig. 5) analysis of our

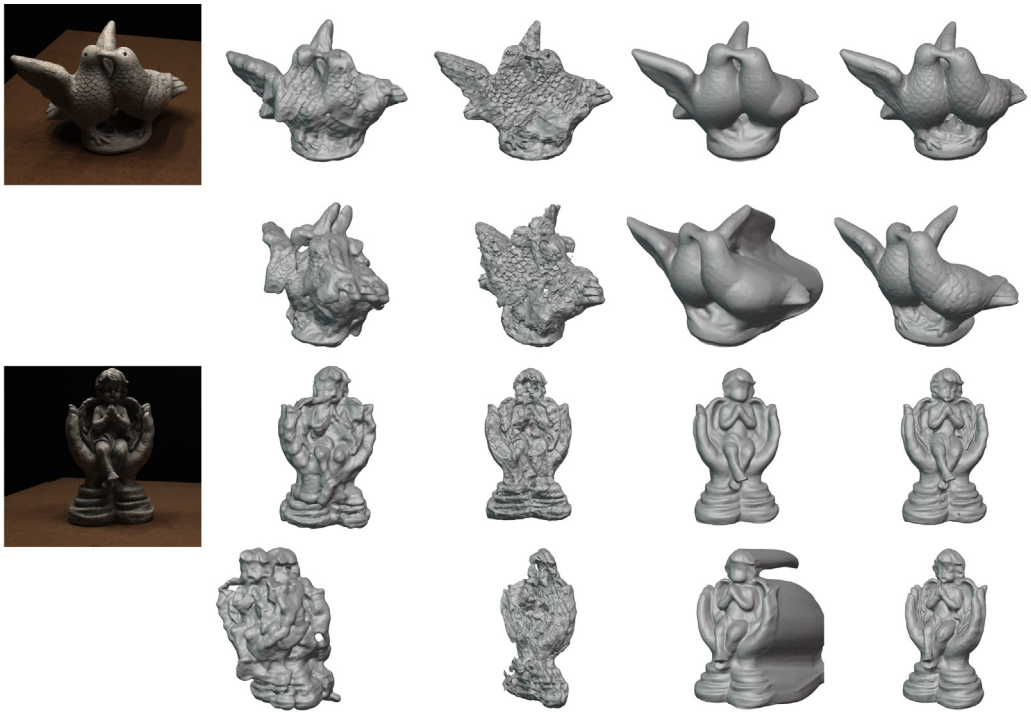

Fig. 4. Comparisons on surface reconstruction on DTU (dense case). Top row: input view, bottom row: novel view.

Fig. 5. Comparisons on surface reconstruction on DTU (sparse case). Top row: input view, bottom row: novel view. Left column: surface, right column: texture.

Table 1
Evaluation on DTU dataset with dense depth prior.

Scan	COLMAP[a]		DVR		IDR		NeuS		SparseNeuS[a]		Ours	
	CD	PSNR	CD	PSNR	CD	PSNR	CD	PSNR	CD	PSNR	CD	PSNR
24	**0.9**	–	–	–	8.86	7.21	7.63	12.02	1.29	–	1.86	**17.81**
37	2.89	–	–	–	8.63	6.33	8.11	7.97	2.27	–	**2.06**	**11.6**
40	1.63	–	–	–	7.14	5.69	7.98	8.81	1.57	–	**1.27**	**15.62**
55	1.08	–	–	–	9.33	9.5	6.99	10.36	**0.88**	–	0.91	**15.37**
63	2.18	–	–	–	8.26	7.86	6.15	11.8	1.61	–	**0.98**	**18.14**
65	1.94	–	2.51	11.25	7.21	9.19	5.42	8.89	1.86	–	**1.09**	**17.72**
69	1.61	–	–	–	6.8	12.36	6.39	14.1	1.06	–	**0.9**	**20.3**
83	1.30	–	–	–	6.43	13.5	8.92	13.82	1.27	–	**1.48**	**22.11**
97	2.34	–	–	–	7.44	8.63	7.25	12.48	1.42	–	**1.16**	**17.12**
105	1.28	–	–	–	7.79	8.18	7.96	12.35	1.07	–	**0.87**	**20.62**
106	1.10	–	1.67	19.3	6.54	13.33	6.12	14.02	0.99	–	**0.97**	**22.81**
110	1.42	–	–	–	7.74	14.95	5.91	13.4	**0.87**	–	1.1	**17.28**
114	0.76	–	–	–	7.12	11.32	5.54	13.6	**0.54**	–	0.6	**20.75**
118	1.17	–	1.70	20.93	7.78	12.84	6.24	16.18	1.15	–	**0.58**	**22.63**
122	**1.14**	–	–	–	7.41	11.96	6.43	14.18	1.18	–	1.28	**22.3**
Mean	1.52	–	1.96	17.16	7.63	10.19	6.87	12.27	1.27	–	**1.14**	**18.81**

Note.
[a]The results of COLMAP and SparseNeuS are provided from SparseNeuS [24] with three-view image setting.

Table 2
Evaluation of DVR and our method on DTU dataset with sparse depth prior (99% sparseness).

Scan	DVR (sparse)		Ours (sparse)	
	CD	PSNR	CD	PSNR
24	–	–	1.94	17.0
37	–	–	2.36	11.31
40	–	–	1.55	15.72
55	–	–	1.04	14.22
63	–	–	1.24	17.58
65	5.9	7.25	1.41	16.81
69	–	–	1.09	20.25
83	–	–	1.49	22.61
97	–	–	1.21	17.52
105	–	–	1.18	21.02
106	7.46	12.76	1.06	22.68
110	–	–	0.96	17.28
114	–	–	0.67	20.36
118	6.81	14.88	0.91	23.01
122	–	–	1.27	22.6
Mean	6.73	11.63	**1.29**	**18.66**

DiffSVR and DVR, which is one of the closest methods to ours (both of us use depths in the framework). In this subsection, we evaluate our method and DVR on a more challenging case, i.e., only with single-view sparse depth prior . Note that it is often difficult and costly to obtain the dense depth map; therefore, high-fidelity reconstruction with sparse depth map constitutes practical solution for real-world applications. Our method significantly outperforms DVR on the sparse case under the single-view setting. This experiment demonstrates that surface optimization fails in DVR with depth loss when only highly sparse depth provided (such as 99% sparseness). Our method can utilize highly sparse depth prior more effectively and is robust to extreme sparseness level. We can also provide better quality surface and appearance as shown in Fig. 5. Note that our method on sparse case also outperforms COLMAP, IDR, NeuS, and DVR and achieves comparable performance as SparseNeuS (with three views) from Table 1.

4.3. Robustness to sparse depth prior

The main goal of this experiment is to show the robustness of our method to the extremely high-level depth sparsity. Firstly, we evaluate different sparseness levels on scan 106 of DTU dataset on two main numerical metrics for geometry and appearance: CD and PSNR. As shown in Fig. 6, our method stays robust up to the depth sparseness of 99% (i.e., with only 1% of the original dense depth). Only beyond that threshold, the quality of the generated mesh starts degrading along with the quality of renderings. Secondly, we pick 99% sparseness and evaluate our method on all scans from DTU dataset with state-of-the-art DVR approach. Since DVR only provides three scans with depth prior, we evaluate it on such three scans with the same 99% sparseness level of ours. The quantitative results are provided in Table 2, which shows that our method outperforms DVR method by a large margin with highly sparse depth level. Additional evaluation on other metrics (e.g., mask IoU, SSIM, LPIPS) are provided in Supplementary Material.

Additionally, we conduct qualitative comparisons of our method trained with dense and sparse prior. As shown in Fig. 6, our method on the highly sparse depth preserves some levels of details and does not suffer from surface incompleteness. Also, it is noted that the generated surfaces with sparse depths have smooth surface without bumps or noises for the places where depth prior is not provided.

To sum it up, our extensive experiments show that the proposed DiffSVR is robust for utilizing highly sparse depth prior information. Moreover, we show the importance of using sparse depth prior for sampling points (much more effective), instead of using it as a loss (as a supervision) for optimizing implicit occupancy field, e.g., DVR [20], or only using RGB images, e.g., COLMAP [27], IDR [21], NeuS [23], SparseNeuS [24], which cannot deal well with single-view input to produce a valid surface reconstruction.

4.4. Ablation study

We provide ablation studies in Table 3 on sampling strategy for DOF and investigate the importance of the differentiable neural surface rendering with its backpropogation defined in Method section. The ablation study is performed on the scan 106 of DTU dataset with highly sparse depth prior.

Firstly, we evaluate the importance of our sampling strategy on different regions (R_{fro}, R_{sur}, and R_{beh}) in DOF by six different experiments as shown in Table 3. The first experiment represents the case when we only sample one single point per ray on the depth-based surface position. This experiment somehow mimics the DVR's idea and achieves comparably similar performance as DVR on sparse depth case (as shown in Table 2). The next experiment (i.e., '0-1-1-0') shows the case when we sample only two

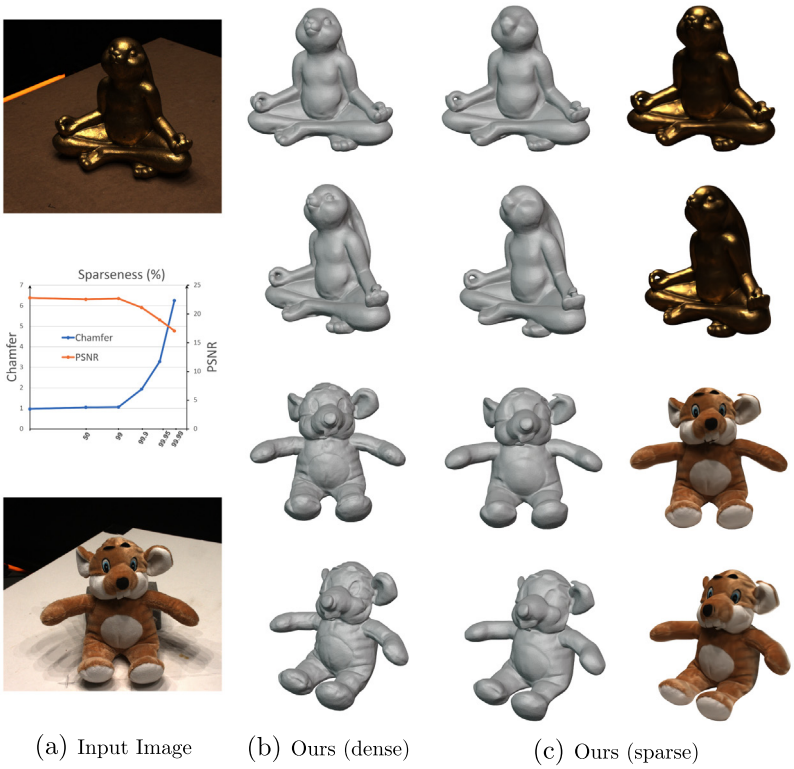

Fig. 6. The surface reconstruction results with sparse (99% sparseness) vs dense depth prior. Top row: input view, bottom row: novel view. The textured surface reconstruction with sparse depth prior is also provided.

Table 3
Ablation study on sampling in DOF and differentiable surface rendering.

	CD
Single surface point/ray	6.51
0-1-1-0	1.26
1-1-1-0	1.4
0-1-1-1	1.53
1-1-1-1	**1.22**
1-1-1-1 (uniform)	1.39
1-1-1-1 w/ NR w/o backprop	1.17
1-1-1-1 w/ NR w/ backprop (full model)	**1.06**

Table 4
Evaluation of DVR and our method on KITTI dataset.

Method	Scan#1				Scan#2			
	MAE	PSNR	SSIM	LPIPS	MAE	PSNR	SSIM	LPIPS
DVR (sparse)	1.483	8.35	0.632	0.382	0.981	8.85	0.658	0.379
Ours (sparse)	1.278	**25.53**	0.879	**0.191**	0.638	23.89	**0.832**	0.226
Ours (dense)	**0.953**	24.86	**0.869**	0.203	**0.427**	**24.08**	**0.832**	**0.224**

points in the region of R_{sur} around the provided sparse depth. The other two experiments represent the idea of sampling additional points either in front R_{fro} (i.e., '1-1-1-0') or behind R_{beh} the sparse depth with self-labeling (i.e., '0-1-1-1'). Finally, we run our model with full surface-aware sampling and self-labeling strategy. This setting outperforms all previous ones. Additionally, we perform one more experiment with uniform sampling around surface to show the importance of random sampling around the sparse-depth-aware surface in our sampling strategy. We have tested to sample more points in different regions, but the setting of '1-1-1-1' is sufficiently effective.

Secondly, we further investigate the importance of our proposed differentiable neural surface rendering to generate better surface geometry along with texture as shown in Table 2. We have done a couple of experiments with our neural renderer. In the first experiment, we add neural renderer (NR) to our best DOF model (i.e., '1-1-1-1'), but turn off our proposed backpropogation. In the second experiment, we turn on the proposed backpropogation, which is considered as our full model. The experiments in this paper are run under the last setting in Table 2 unless noted otherwise.

Additionally, we provide accompanying ablation study illustration in Fig. 7. Our full model (i.e., (h)) visually outperforms the case with single surface point sampled along the ray (i.e., (a)), as well as the case with uniform sampling (i.e., (f)). The rest of the cases has close front view performance because similarly we sample two points around the surface in the same way. From the backside view, our full model (i.e., (h)) generates more complete and reasonable results of the occluded part comparing to experiments in (b), (c), (d), (e), and (g).

4.5. Evaluation on KITTI

We perform qualitative and quantitative comparisons of our method with DVR (without 3D supervision) on KITTI dataset. We have randomly selected two frames from two different scans for our evaluation. In order to measure the reconstruction quality of the reconstructed surfaces we apply the mean square error (MAE) between the reconstructed mesh and the raw (dense) LiDAR points in meters. To evaluate the quality of the image renderings, we report the standard PSNR, SSIM metrics in [64], and LPIPS [65].

In reality, it is often difficult or costly to obtain the dense depth map or dense multi-view images, therefore, we evaluate our superbly fast and lightweight 3D reconstruction method from single-view RGB image and highly sparse depth with the practical applications in autonomous driving systems, robotics, etc. Worth mentioning, using sparse depth sensing significantly reduces the financial and computational costs, hence increasing the practical

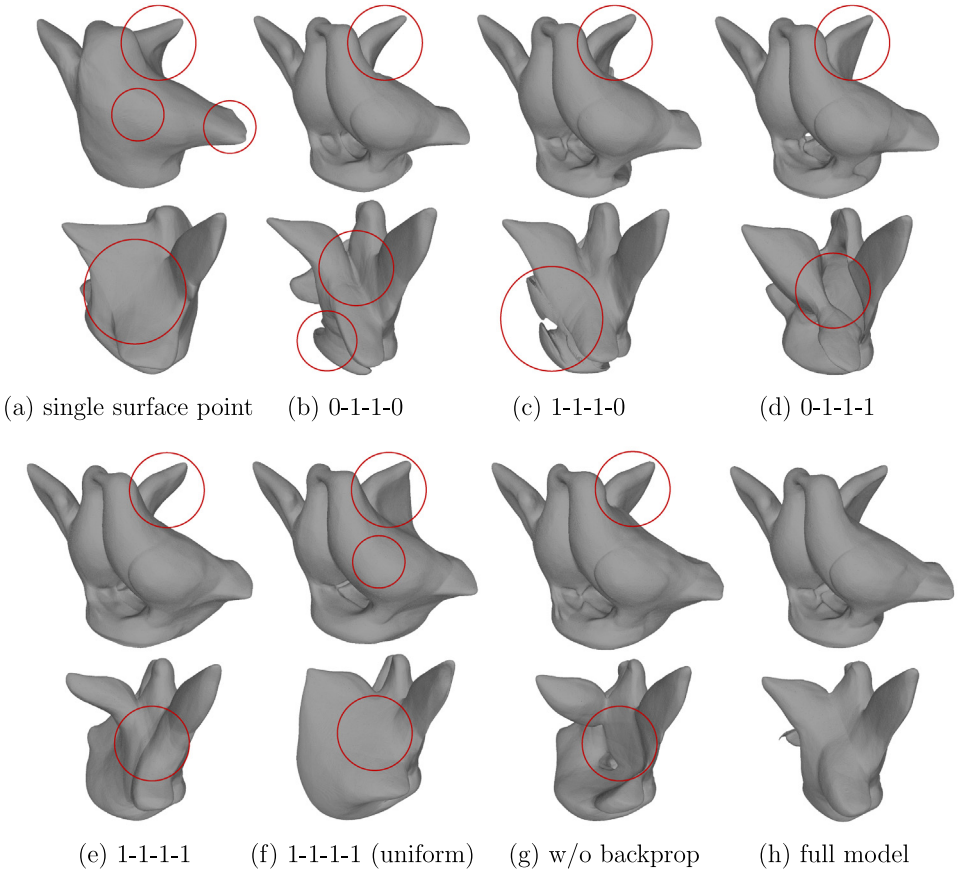

Fig. 7. Ablation study illustration. Top row: front view, bottom row: backside view. Red circles highlight some major differences between the full model results and other cases.

value/usage of our approach in such applications. Therefore, we provide quantitative (Table 4) and qualitative (Fig. 8) analysis of our DiffSVR and DVR, which is one of the closest methods to ours (both of us use depths in the framework). In this subsection, we evaluate our method and DVR on the same challenging case, i.e., only with single-view sparse depth prior. We reduce the full captured depths from raw LiDAR points up to 90%, where only 10% of depths are available for training in our experiments. This experiment demonstrates that both surface and texture optimization fails in DVR with depth loss when only highly sparse depth sensing provided (up to 90% sparseness). Our method significantly outperforms DVR under the single-view setting. It can utilize highly sparse depth prior more effectively and is robust to extreme sparseness level. Our method provides better quality surface and appearance as shown in Fig. 8. Additionally, we quantitatively evaluate our method on full captured depths (dense case), as shown in Table 4. The experiments show that increasing the depth resolution helps to improve the quality of the reconstructed surface geometry, as well as improve the texture quality from novel views as shown in figures in Supplementary Material.

Due to the large scale of outdoor scene with a limited input image resolution from the KITTI scans (which is even smaller than input image resolution from DTU scans, but the 3D scene complexity and size in KITTI are much higher than those of 3D objects in DTU), it is noted that the qualitative visualization results from KITTI scans are not as high-fidelity as the results from DTU scans. In the future, we will use more frames to realize the high-quality 3D reconstruction from KITTI scans.

5. Conclusion

In this work, we propose a novel DiffSVR framework with the new depth-aware occupancy function and differentiable surface rendering techniques that enhance the implicit geometry and texture representations for the 3D surface reconstruction from a single-view image. Our framework exhibits robustness to the high sparseness level of depth prior because of its generalizability and flexibility to available constraints. Through extensive experiments and comparisons, our method not only numerically achieves the new state-of-the-art performance, but also qualitatively produces surface with better geometric details and more accurate textures. Due to the fundamental challenge in the single-view reconstruction, the backside quality of ours does still need to be improved. In our future work, we will focus on dynamic object/scene reconstruction and enhancing the quality of the large-scale scene reconstruction with multi-view inputs, such as KITTI dataset.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Fig. 8. Comparisons on surface and texture reconstruction on KITTI (sparse case). Left column: scan#1, right column: scan#2. Top row shows RGB image with sparse depths as input. For each method first two rows: input view (surface + texture), second two rows: novel view (surface + texture).

Acknowledgments

We would like to thank the reviewers for their valuable comments. This work was partially supported by the National Science Foundation (NSF), USA under Grant Numbers OAC-1845962 and OAC-1910469.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.cad.2023.103604.

References

[1] Baumgart BG. Geometric modeling for computer vision. Stanford University; 1974.
[2] Loper MM, Black MJ. OpenDR: An approximate differentiable renderer. In: In proceedings of the European conference on computer vision. 2014, p. 154–69.
[3] Liu G, Ceylan D, Yumer E, Yang J, Lien J-M. Material editing using a physically based rendering network. In: In proceedings of the IEEE international conference on computer vision. 2017, p. 2261–9.
[4] Richardson E, Sela M, Or-El R, Kimmel R. Learning detailed face reconstruction from a single image. In: In proceedings of the IEEE conference on computer vision and pattern recognition. 2017, p. 1259–68.
[5] Paschalidou D, Ulusoy O, Schmitt C, Van Gool L, Geiger A. RayNet: Learning volumetric 3D reconstruction with ray potentials. In: In proceedings of the IEEE conference on computer vision and pattern recognition. 2018, p. 3897–906.
[6] Genova K, Cole F, Maschinot A, Sarna A, Vlasic D, Freeman WT. Unsupervised training for 3D morphable model regression. In: In proceedings of the IEEE conference on computer vision and pattern recognition. 2018, p. 8377–86.
[7] Kalogerakis E, Averkiou M, Maji S, Chaudhuri S. 3D shape segmentation with projective convolutional networks. In: In proceedings of the IEEE conference on computer vision and pattern recognition. 2017, p. 3779–88.

[8] Chen W, Ling H, Gao J, Smith E, Lehtinen J, Jacobson A, Fidler S. Learning to predict 3D objects with an interpolation-based differentiable renderer. In: In proceedings of the advances in neural information processing systems, Vol. 32. 2019, p. 9609–19.
[9] Liu S, Chen W, Li T, Li H. Soft rasterizer: Differentiable rendering for unsupervised single-view mesh reconstruction. 2019, arXiv preprint arXiv:1901.05567.
[10] Choy CB, Xu D, Gwak J, Chen K, Savarese S. 3D-R2N2: A unified approach for single and multi-view 3D object reconstruction. In: In proceedings of the European conference on computer vision. 2016, p. 628–44.
[11] Fan H, Su H, Guibas LJ. A point set generation network for 3D object reconstruction from a single image. In: In proceedings of the IEEE conference on computer vision and pattern recognition. 2017, p. 605–13.
[12] Wang N, Zhang Y, Li Z, Fu Y, Liu W, Jiang Y-G. Pixel2Mesh: Generating 3D mesh models from single RGB images. In: In proceedings of the European conference on computer vision. 2018, p. 52–67.
[13] Pontes JK, Kong C, Sridharan S, Lucey S, Eriksson A, Fookes C. Image2Mesh: A learning framework for single image 3D reconstruction. In: Asian conference on computer vision. 2018, p. 365–81.
[14] Jack D, Pontes JK, Sridharan S, Fookes C, Shirazi S, Maire F, Eriksson A. Learning free-form deformations for 3D object reconstruction. In: Asian conference on computer vision. 2018, p. 317–33.
[15] Ge L, Ren Z, Li Y, Xue Z, Wang Y, Cai J, Yuan J. 3D hand shape and pose estimation from a single RGB image. In: In proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019, p. 10833–42.
[16] Wang Y, Zhong Z, Hua J. DeepOrganNet: on-the-fly reconstruction and visualization of 3D/4D lung models from single-view projections by deep deformation network. IEEE transactions on visualization and computer graphics 2019;26(1):960–70.
[17] Lin Y, Wang Y, Li Y-F, Wang Z, Gao Y, Khan L. Single view point cloud generation via unified 3D prototype. In: In proceedings of the AAAI conference on artificial intelligence, Vol. 35. 2021, p. 2064–72.
[18] Park JJ, Florence P, Straub J, Newcombe R, Lovegrove S. DeepSDF: Learning continuous signed distance functions for shape representation. In: In proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019, p. 165–74.
[19] Mescheder L, Oechsle M, Niemeyer M, Nowozin S, Geiger A. Occupancy networks: Learning 3D reconstruction in function space. In: In proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019, p. 4460–70.
[20] Niemeyer M, Mescheder L, Oechsle M, Geiger A. Differentiable volumetric rendering: Learning implicit 3D representations without 3D supervision. In: In proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020, p. 3504–15.
[21] Yariv L, Kasten Y, Moran D, Galun M, Atzmon M, Ronen B, Lipman Y. Multiview neural surface reconstruction by disentangling geometry and appearance. In: In proceedings of the advances in neural information processing systems, Vol. 33. 2020.
[22] Oechsle M, Peng S, Geiger A. UNISURF: Unifying neural implicit surfaces and radiance fields for multi-view reconstruction. In: In proceedings of the IEEE/CVF international conference on computer vision. 2021, p. 5589–99.
[23] Wang P, Liu L, Liu Y, Theobalt C, Komura T, Wang W. NeuS: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. In: In proceedings of the advances in neural information processing systems. 2021.
[24] Long X, Lin C, Wang P, Komura T, Wang W. SparseNeuS: Fast generalizable neural surface reconstruction from sparse views. In: In proceedings of the European conference on computer vision. 2022.
[25] Andrew A. Multiple view geometry in computer vision. Kybernetes 2001.
[26] Furukawa Y, Ponce J. Accurate, dense, and robust multiview stereopsis. IEEE Trans Pattern Anal Mach Intell 2009;32(8):1362–76.
[27] Schönberger JL, Zheng E, Frahm J-M, Pollefeys M. Pixelwise view selection for unstructured multi-view stereo. In: In proceedings of the European conference on computer vision. 2016, p. 501–18.
[28] Campbell ND, Vogiatzis G, Hernández C, Cipolla R. Using multiple hypotheses to improve depth-maps for multi-view stereo. In: In proceedings of the European conference on computer vision. 2008, p. 766–79.
[29] Triggs B, McLauchlan PF, Hartley RI, Fitzgibbon AW. Bundle adjustment - a modern synthesis. In: International workshop on vision algorithms. 1999, p. 298–372.
[30] Bleyer M, Rhemann C, Rother C. PatchMatch stereo-stereo matching with slanted support windows. In: BMVC, Vol. 11. 2011, p. 1–11.
[31] Merrell P, Akbarzadeh A, Wang L, Mordohai P, Frahm J-M, Yang R, Nistér D, Pollefeys M. Real-time visibility-based fusion of depth maps. In: In proceedings of the international conference on computer vision. 2007, p. 1–8.
[32] Zach C, Pock T, Bischof H. A globally optimal algorithm for robust TV-L1 range image integration. In: In proceedings of the international conference on computer vision. 2007, p. 1–8.
[33] Kazhdan M, Bolitho M, Hoppe H. Poisson surface reconstruction. In: In proceedings of the fourth eurographics symposium on geometry processing, Vol. 7. 2006.
[34] Kazhdan M, Hoppe H. Screened poisson surface reconstruction. ACM Trans Graph (ToG) 2013;32(3):1–13.
[35] De Bonet JS, Viola P. Poxels: Probabilistic voxelized volume reconstruction. In: In proceedings of the IEEE international conference on computer vision. 1999, p. 418–25.
[36] Broadhurst A, Drummond TW, Cipolla R. A probabilistic framework for space carving. In: In proceedings of the IEEE international conference on computer vision, Vol. 1. 2001, p. 388–93.
[37] Sitzmann V, Zollhöfer M, Wetzstein G. Scene representation networks: Continuous 3d-structure-aware neural scene representations. In: In proceedings of the advances in neural information processing systems. 2019.
[38] Agrawal M, Davis LS. A probabilistic framework for surface reconstruction from multiple images. In: In proceedings of the IEEE computer society conference on computer vision and pattern recognition, Vol. 2. 2001, p. II.
[39] Kutulakos KN, Seitz SM. A theory of shape by space carving. Int J Comput Vis 2000;38(3):199–218.
[40] Marr D, Poggio T. Cooperative computation of stereo disparity. Science 1976;194(4262):283–7.
[41] Tulsiani S, Zhou T, Efros AA, Malik J. Multi-view supervision for single-view reconstruction via differentiable ray consistency. In: In proceedings of the IEEE conference on computer vision and pattern recognition. 2017, p. 2626–34.
[42] Atzmon M, Haim N, Yariv L, Israelov O, Maron H, Lipman Y. Controlling neural level sets. In: In proceedings of the advances in neural information processing systems. 2019.
[43] Genova K, Cole F, Vlasic D, Sarna A, Freeman WT, Funkhouser T. Learning shape templates with structured implicit functions. In: In proceedings of the IEEE/CVF international conference on computer vision. 2019, p. 7154–64.
[44] Oechsle M, Mescheder L, Niemeyer M, Strauss T, Geiger A. Texture fields: Learning texture representations in function space. In: In proceedings of the IEEE/CVF international conference on computer vision. 2019, p. 4531–40.
[45] Michalkiewicz M, Pontes JK, Jack D, Baktashmotlagh M, Eriksson A. Implicit surface representations as layers in neural networks. In: In proceedings of the IEEE/CVF international conference on computer vision. 2019, p. 4743–52.
[46] Niemeyer M, Mescheder L, Oechsle M, Geiger A. Occupancy flow: 4D reconstruction by learning particle dynamics. In: Proceedings of the IEEE/CVF international conference on computer vision. 2019, p. 5379–89.
[47] Liu L, Gu J, Lin KZ, Chua T-S, Theobalt C. Neural sparse voxel fields. In: In proceedings of the advances in neural information processing systems. 2020.
[48] Liu S, Zhang Y, Peng S, Shi B, Pollefeys M, Cui Z. DIST: Rendering deep implicit signed distance function with differentiable sphere tracing. In: In proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020, p. 2019–28.
[49] Xiao Y, Zhu H, Yang H, Diao Z, Lu X, Cao X. Detailed facial geometry recovery from multi-view images by learning an implicit function. In: In proceedings of the AAAI conference on artificial intelligence. 2022.
[50] Mildenhall B, Srinivasan PP, Tancik M, Barron JT, Ramamoorthi R, Ng R. NeRF: Representing scenes as neural radiance fields for view synthesis. In: In proceedings of the European conference on computer vision. 2020.
[51] Boss M, Braun R, Jampani V, Barron JT, Liu C, Lensch H. NeRD: Neural reflectance decomposition from image collections. In: In proceedings of the IEEE/CVF international conference on computer vision. 2021, p. 12684–94.
[52] Martin-Brualla R, Radwan N, Sajjadi MS, Barron JT, Dosovitskiy A, Duckworth D. NeRF in the wild: Neural radiance fields for unconstrained photo collections. In: In proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2021, p. 7210–9.
[53] Neff T, Stadlbauer P, Parger M, Kurz A, Alla Chaitanya CR, Kaplanyan A, Steinberger M. DONeRF: Towards real-time rendering of neural radiance fields using depth oracle networks. 2021, arXiv e-prints, arXiv–2103.
[54] Niemeyer M, Geiger A. GIRAFFE: Representing scenes as compositional generative neural feature fields. In: In proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2021, p. 11453–64.
[55] Peng S, Zhang Y, Xu Y, Wang Q, Shuai Q, Bao H, Zhou X. Neural body: Implicit neural representations with structured latent codes for novel view synthesis of dynamic humans. In: In proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2021, p. 9054–63.
[56] Pumarola A, Corona E, Pons-Moll G, Moreno-Noguer F. D-NeRF: Neural radiance fields for dynamic scenes. In: In proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2021, p. 10318–27.
[57] Schwarz K, Liao Y, Niemeyer M, Geiger A. GRAF: Generative radiance fields for 3D-aware image synthesis. 2020, arXiv preprint arXiv:2007.02442.

[58] Srinivasan PP, Deng B, Zhang X, Tancik M, Mildenhall B, Barron JT. NeRV: Neural reflectance and visibility fields for relighting and view synthesis. In: In proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2021, p. 7495–504.
[59] Yu A, Ye V, Tancik M, Kanazawa A. PixelNeRF: Neural radiance fields from one or few images. In: In proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2021, p. 4578–87.
[60] Deng K, Liu A, Zhu J-Y, Ramanan D. Depth-supervised NeRF: Fewer views and faster training for free. In: In proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2022, p. 12882–91.
[61] Roessle B, Barron JT, Mildenhall B, Srinivasan PP, Nießner M. Dense depth priors for neural radiance fields from sparse input views. In: In proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2022, p. 12892–901.
[62] Jensen R, Dahl A, Vogiatzis G, Tola E, Aanæs H. Large scale multi-view stereopsis evaluation. In: In proceedings of the IEEE conference on computer vision and pattern recognition. 2014, p. 406–13.
[63] Geiger A, Lenz P, Stiller C, Urtasun R. Vision meets robotics: The KITTI dataset. Int J Robot Res 2013;32(11):1231–7.
[64] Wang Z, Bovik AC, Sheikh HR, Simoncelli EP. Image quality assessment: from error visibility to structural similarity. IEEE Trans Image Process 2004;13(4):600–12.
[65] Zhang R, Isola P, Efros AA, Shechtman E, Wang O. The unreasonable effectiveness of deep features as a perceptual metric. In: In proceedings of the IEEE conference on computer vision and pattern recognition. 2018, p. 586–95.