

# Coordinate Quantized Neural Implicit Representations for Multi-view Reconstruction

Sijia Jiang, Jing Hua, Zhizhong Han

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

sijiajiang@wayne.edu, jinghua@wayne.edu, h312h@wayne.edu

## Abstract

In recent years, huge progress has been made on learning neural implicit representations from multi-view images for 3D reconstruction. As an additional input complementing coordinates, using sinusoidal functions as positional encodings plays a key role in revealing high frequency details with coordinate-based neural networks. However, high frequency positional encodings make the optimization unstable, which results in noisy reconstructions and artifacts in empty space. To resolve this issue in a general sense, we introduce to learn neural implicit representations with quantized coordinates, which reduces the uncertainty and ambiguity in the field during optimization. Instead of continuous coordinates, we discretize continuous coordinates into discrete coordinates using nearest interpolation among quantized coordinates which are obtained by discretizing the field in an extremely high resolution. We use discrete coordinates and their positional encodings to learn implicit functions through volume rendering. This significantly reduces the variations in the sample space, and triggers more multi-view consistency constraints on intersections of rays from different views, which enables to infer implicit function in a more effective way. Our quantized coordinates do not bring any computational burden, and can seamlessly work upon the latest methods. Our evaluations under the widely used benchmarks show our superiority over the state-of-the-art. Our code is available at <https://github.com/MachinePerceptionLab/CQ-NIR>.

## 1. Introduction

Learning implicit representations from multi-view images is a challenge in reconstructing 3D geometry in a scene. The latest methods learn implicit representations using coordinate-based neural networks to infer signed distance or occupancy fields [43, 67, 66, 13, 63, 68, 64, 61, 62, 16, 25, 50] through volume rendering. By shooting rays across the fields, we render RGB values at a pixel

by integrating colors and geometry at 3D queries sampled along a ray through volume rendering. The images rendered from neural implicit functions are compared with the ground truth images, which measures errors to improve the neural implicit functions.

Learning high fidelity implicit representations requires to use positional encodings [39, 43, 63, 64] as a complement to coordinates, which remedies the incapability of coordinate-based neural networks in modeling high frequency details. Positional encodings are vectors formed by sinusoidal functions of coordinates with both low and high frequencies [55, 59], where the frequency band is shown as the key factor to capture details in different scenes. However, higher frequency turns out to bring noises, which results in artifacts on surfaces and in empty spaces. To stabilize the optimization with high frequency positional encodings, some methods [18, 64, 45] learn soft masks to gradually expose high frequency components over training iterations. However, this masking strategy relies on training iterations and numbers of frequency components, which is tedious to tune in a general sense.

To resolve this issue, we propose to use quantized coordinates to learn neural implicit representations from multi-view images. Instead of continuous coordinates and positional encodings of continuous coordinates in previous methods [39, 63, 68, 64, 61, 62, 16, 71], we use discrete coordinates and positional encodings of discrete coordinates as the input of coordinate-based neural networks, where we discretize the field in an extremely high resolutions. Our insight here is to decrease the uncertainty and ambiguity in the field during optimization. We achieve this by introducing discrete coordinates with two reasons. On the one hand, we enable networks to merely observe a finite set of discrete coordinates rather than infinite continuous variations, which simplifies the optimization by significantly reducing variations in the sample space. On the other hand, a discrete coordinate covers an area rather than a point, hence rays from different views are more easily to have overlapped samples with each other. This triggers more multi-view consistency constraints to take effect at these intersections, which leads

to more effective inference. Our quantized coordinates do not bring any extra computational burden, inconsistency on borders of neighboring coordinates, and provide a general strategy which can be used upon different methods. We evaluate our improvements over the latest methods under multiple benchmarks. Our contributions are listed below.

- i) We introduce quantized coordinates to learn neural implicit functions from multi-view images. By discretizing a field in an extremely high resolution, we introduce efficient ways of using discrete coordinates, which does not bring extra computational burden and inconsistency on borders of neighboring coordinates.
- ii) We report analysis on how discrete coordinates decrease the uncertainty and ambiguity in the field by reducing the variations in the sample space and triggering more multi-view consistency constraints to infer implicit functions in a more effective way.
- iii) Our discrete coordinates can seamlessly work upon the latest methods. We justify our effectiveness by showing significant improvements over the state-of-the-art results under the widely used benchmarks.

## 2. Related Work

**3D Reconstruction from Multiple Images.** Reconstructing 3D shapes from multiple images has been extensively studied in 3D computer vision [53, 54, 39, 14, 43, 63, 68, 64, 61, 62, 16, 8, 49]. Given multiple RGB images, classic multi-view stereo (MVS) [53, 54] methods employ multi-view consistency to estimate depth information. They rely on matching key points on different views, which is limited by large viewpoint variations and complex illumination. With multiple silhouette images, we can reconstruct 3D shapes as voxel grids using space carving [23]. The disadvantages of these methods include the inability of revealing concave structures and low resolutions in voxel grids.

Recent methods [65] employ neural networks to implement the MVS framework. During training, they learn priors using depth supervision or multi-view consistency in an unsupervised way, and then, generalize the priors to predict depth images for unseen cases through a forward pass.

These methods reconstructed 3D shapes as point clouds or voxel grids, both of which are discrete. While neural implicit representations for 3D reconstruction represent surfaces as the level set which is continuous.

**Neural Implicit Representations.** Neural implicit representations have shown prominent performance in representing 3D geometry [38, 44, 37, 9, 21, 5, 22, 47]. We can learn neural implicit representations using coordinate-based neural networks from 3D supervision [20, 4, 56, 33, 58, 26, 60], point clouds [72, 29, 15, 1, 70, 2, 5, 41, 17, 6, 30, 7, 72, 35, 32, 31, 24], or multi-view images [39, 14, 43, 63, 68, 64, 61, 62, 16, 36]. Since 3D supervision and point clouds expose more explicit geometry clues than multiple images,

methods learning from these kinds of supervision do not employ positional encodings as input. Hence, our discrete coordinates are mainly evaluated upon the methods using multi-view images as supervision.

With differentiable rendering techniques, we are enabled to evaluate the correctness of neural implicit representations using errors between rendered images and ground truth images. With surface rendering [21], DVR [42] and IDR [67] infer the radiance on surfaces. IDR also models view direction as a condition to reconstruct high frequency details. Since these methods focus on surfaces, they require masks to filter out the background.

NeRF [39] and its variations [45, 40, 51, 52, 48, 27, 34] use volume rendering to simultaneously model geometry and color. These methods were proposed for novel view synthesis, and render images without masks. Using volume rendering, unisurf [43] and NeuS [63] revise the rendering procedure to render occupancy and signed distance fields with colors, which infers accurate implicit functions. Following methods improve accuracy of implicit functions using additional priors or losses including depth [68, 3, 73], normals [68, 62, 16], and multi-view consistency [14].

## 3. Preliminary

**Neural Radiance Fields.** NeRF [39] represents scenes by jointly modeling volume densities and colors using a neural network. Starting a pixel, we shoot a ray, and integrates densities and colors at samples along the ray into RGB values at the pixel through volume rendering. At a 3D sample  $\mathbf{q} \in \mathbb{R}^3$ , the neural network predicts the density  $\sigma(\mathbf{q}) \in \mathbb{R}^+$  and color  $c(\mathbf{q}, \mathbf{d}) \in \mathbb{R}^3$ , where  $\mathbf{d}$  indicates the ray direction passing  $\mathbf{q}$  which enables to model view-dependent effects such as reflections.

To sample  $I$  queries  $\{\mathbf{q}_i\}$  along a ray with a direction  $\mathbf{d}$ , NeRF parameterizes the ray using the distance  $t$  to the camera center  $\mathbf{o}$ ,  $\mathbf{q}_i = \mathbf{o} + t\mathbf{d}$ . The rendered color along the ray is obtained by volume rendering below,

$$C = \sum_{i=1}^I T_i (1 - \exp(-\sigma(\mathbf{q}_i)\delta_i)) c(\mathbf{q}_i, \mathbf{d}), \quad (1)$$

where  $T_i = \exp(-\sum_{j < i} \sigma(\mathbf{q}_j)\delta_j)$  is the accumulated transmittance along the ray and  $\delta_i$  is the Euclidean distance between  $\mathbf{q}_{i+1}$  and  $\mathbf{q}_i$ . The network can be trained by minimizing the error between rendered images and ground truth images through the differentiable volume rendering.

To better fit data containing high frequency variations, positional encoding is introduced to map coordinates  $\mathbf{q}_i$  into a higher dimensional space using sinusoidal functions with a frequency band. Formally, the encoding function  $\gamma(\mathbf{q}_i)$  is defined below,

$$(\sin(\omega_1 \mathbf{q}_i), \cos(\omega_1 \mathbf{q}_i), \dots, \sin(\omega_L \mathbf{q}_i), \cos(\omega_L \mathbf{q}_i)), \quad (2)$$

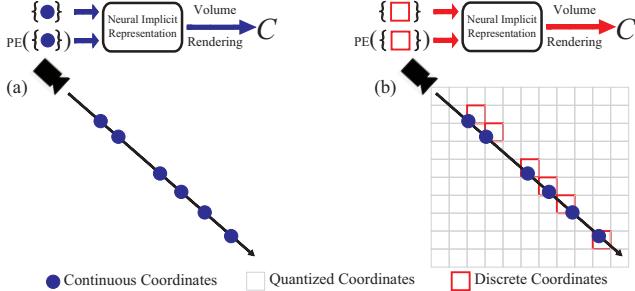


Figure 1. Illustration of our method. We discretize continuous coordinates in (a) into discrete coordinates using quantized coordinates in an extremely high resolution in (b). We use these discrete coordinates and their positional encoding to learn neural implicit representations... $\omega_L$  is a band containing  $L$  frequencies, and  $\omega_L = 2^{L-1}\pi$ . The encoding function  $\gamma$  is applied to each element in the coordinate vector  $q_i$  and the vector indicating view direction  $d$ .

**Neural Implicit Representations.** Based on NeRF, the latest methods learn neural implicit representations  $f_\theta(q_i)$ , such as occupancy fields [37] and signed distance fields [44], along with a color value  $c_\theta(q_i, d)$ , through volume rendering. These methods reformulate the volume rendering equation in Eq. 1 to replace densities into occupancy labels or signed distances, where a function  $V$  is defined to map  $f_\theta$  into alpha. The function  $V$  is the key to enable to learn 3D implicit representations from 2D supervision without masks. These methods optimize neural networks parameterized by  $\theta$  by minimizing the squared error below,

$$\min_{\theta} \left| \left| C_{GT} - \sum_i^I V(\{f_\theta(q_{j \leq i})\}) C_\theta(q_i, d) \right| \right|_2^2, \quad (3)$$

where  $C_{GT}$  is the ground truth color at the pixel emitting the ray.

Moreover, these methods improve the sampling strategy to infer implicit representations in a more efficient way. Specifically, they first use ray marching or secant method to find the intersection between the ray and the scene, and then sample more queries around the intersection to do volume rendering, where intersections are estimated surface points. More advanced techniques for smoother implicit fields include using normals of surface points as the input of the color network [43, 67, 66, 13, 63], adding constraints on normals of neighboring points [43], and using sparse depth from MVS as priors [14].

## 4. Method

**Issues of Continuous Coordinates.** To render RGB values along a ray, current methods sample queries  $q_i$  along the ray according to some specific sampling strategy. As illustrated in Fig. 1 (a), these queries are used as probes to sense the continuous field for volume rendering. They are associated

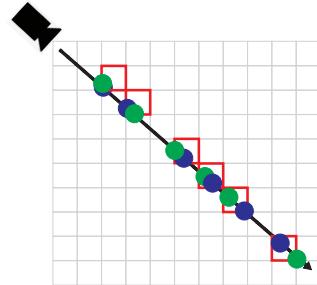


Figure 2. Different colors indicate different iterations (shown as nodes in two colors) of the same ray. Along the same ray, continuous coordinates sampled in different iterations are different, while their discrete coordinates may be the same set.

with continuous coordinates which are further manipulated by sinusoidal functions with a frequency band as a positional encoding  $\gamma(q_i)$ . Continuous coordinates have two issues.

For one thing, continuous coordinates produce a huge variations in the sample space. Along the same ray shown in Fig. 2, the queries sampled in one iteration (blue dots) do not overlap with the ones sampled in another iterations (green dots) since the field is optimized through instantly tuning parameters of the network. The variations are extended to be even larger with high dimensional positional encodings  $\gamma$  as an additional input. This makes implicit functions keep observing different samples as input during training, which is an obstacle that make neural networks to struggle to infer uncertainties and ambiguities in the field.

For another, continuous coordinates are not effective to impose multi-view consistency constraints on inferring implicit functions. The essence of using multi-view consistency to infer occupancy or signed distances is to involve intersections of rays from different views in volume rendering. However, queries are sampled on rays from different views separately without considering consistency, which may make points sampled on both rays do not overlap at the intersection due to randomness in sampling. As illustrated in Fig. 4 (a), two rays from two different views are supposed to intersect on a surface point, while the points sampled for volume rendering along the two rays (blue dots in one ray, green dots in another ray) do not overlap at the intersection. This means that the intersection will not get involved in volume rendering along both rays, resulting in no multi-view consistency constraints to be imposed on the intersection for inferring implicit function value. This is also another obstacle for inferring uncertainties and ambiguities in the field.

**Quantized Coordinates.** To resolve the issues of continuous coordinates, we introduce quantized coordinates to learn implicit functions from multi-view images. As shown in Fig. 1 (b), we first obtain quantized coordinates (centers of squares) by discretizing the field in an extremely high

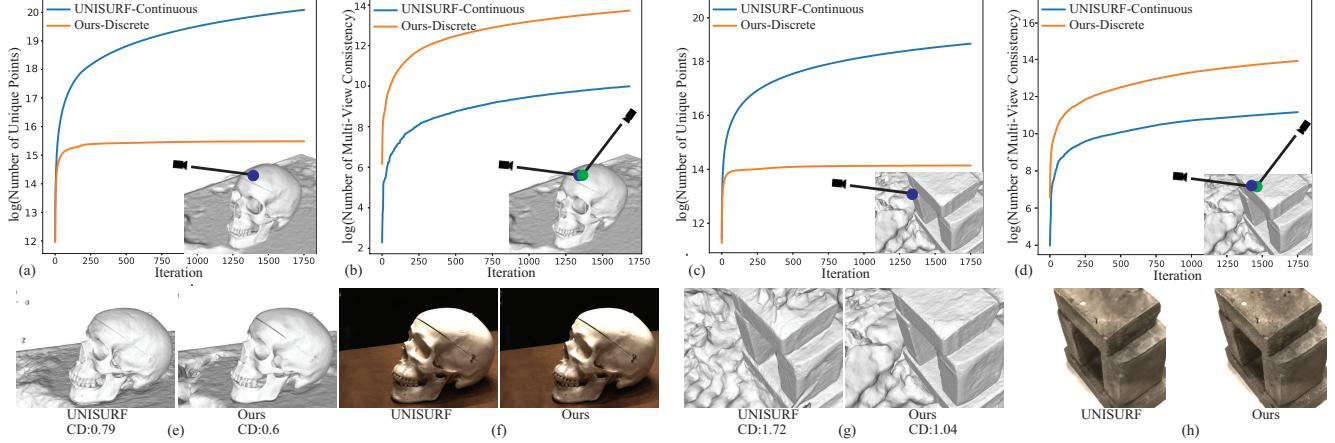


Figure 3. Statistics comparison of continuous coordinates and discrete coordinates in terms of the the number of unique coordinates that the network has observed and the number of overlapped samples along rays from different views.

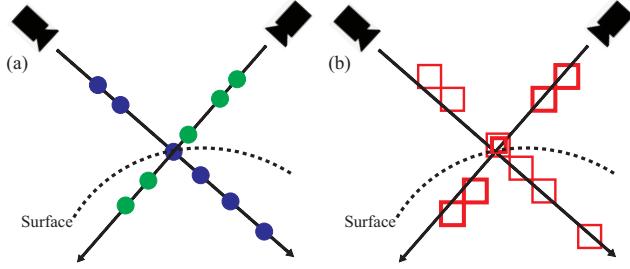


Figure 4. Discrete coordinates in (b) make it easier to involve the same location in volume rendering along the rays from different views than continuous coordinates in (a).

resolution, and then leverage these quantized coordinates to discretize continuous coordinates (blue dots) into discrete ones (centers of red squares). Specifically, we voxelize the field into a voxel grid with a resolution of  $R$ , and use the center of each voxel as quantized coordinates  $\mathbf{e}_j \in \mathbb{R}^3$ , where  $j \in [1, J]$  and  $J = R^3$ . We denote the set of quantized coordinates as  $E = \{\mathbf{e}_j | j \in [1, R^3]\}$ , and use nearest interpolation in  $E$  to discretize each query  $\mathbf{q}_i$  into a discrete coordinate  $\tilde{\mathbf{q}}_i$ , as formulated by,

$$\tilde{\mathbf{q}}_i = \mathbf{e}_k, \quad \text{where } k = \operatorname{argmin}_j \|\mathbf{q}_i - \mathbf{e}_j\|_2. \quad (4)$$

We use discrete coordinates  $\tilde{\mathbf{q}}_i$  of queries and their corresponding positional encodings  $\gamma(\tilde{\mathbf{q}}_i)$  as the input of the network. We reformulate Eq. 2 to obtain  $\gamma(\tilde{\mathbf{q}}_i)$  as,

$$(\sin(\omega_1 \tilde{\mathbf{q}}_i), \cos(\omega_1 \tilde{\mathbf{q}}_i), \dots, \sin(\omega_L \tilde{\mathbf{q}}_i), \cos(\omega_L \tilde{\mathbf{q}}_i), \quad (5)$$

where  $\{\omega_1, \dots, \omega_L\}$  is also a band containing  $L$  frequencies,

and  $\omega_L = 2^{L-1}\pi$ . Accordingly, with discrete coordinates  $\tilde{\mathbf{q}}_i$ , we reformulate volume rendering as,

$$C = \sum_{i=1}^I T_i (1 - \exp(-\sigma(\tilde{\mathbf{q}}_i)\delta_i)) c(\tilde{\mathbf{q}}_i, \mathbf{d}), \quad (6)$$

where  $T_i = \exp(-\sum_{j < i} \sigma(\tilde{\mathbf{q}}_j)\delta_j)$  is the accumulated transmittance along the ray and  $\delta_i$  is still the Euclidean distance between continuous coordinates  $\mathbf{q}_{j+1}$  and  $\mathbf{q}_j$ . In addition, we remove the duplicated discrete coordinates along a ray, and use the unique discrete coordinates in volume rendering in Eq. 6, which achieves more accurate approximation. We learn neural implicit functions with discrete coordinates by minimizing the rendering error,

$$\min_{\theta} \|C_{GT} - \sum_i^I V(\{f_{\theta}(\tilde{\mathbf{q}}_{j \leq i})\}) C_{\theta}(\tilde{\mathbf{q}}_i, \mathbf{d})\|_2^2. \quad (7)$$

**Why Discrete Coordinates Work.** With discrete coordinates, we can significantly reduce the variations in the sample space. Obviously, the network observes a finite set of the coordinates and also their corresponding positional encodings, rather than infinite variations with continuous coordinates especially with high frequencies positional encodings. As illustrated in Fig. 2, for points sampled in different iterations (blue and green dots), their discrete coordinates are the same (indicated by the centers of the red boxes).

We statistically justify our claim in reconstructing a 3D scene from multi-view images in Fig. 3 (a). We use UNISURF [43] as a baseline which samples continuous coordinates to learn an occupancy field via volume rendering. We discrete these continuous coordinates into discrete coordinates using nearest interpolation over  $R^3$  quantized coordinates, where  $R = 51200$ . During optimization, we monitor a fixed set of 1024 rays in each iteration, we focus on the

rays that hit the surface, and record the continuous coordinates and their discrete coordinates sampled on these rays. We count the number of unique continuous coordinates and the number of unique discrete coordinates that the network has observed respectively. We accumulate these numbers over training iterations respectively, and show them using a logarithmic function as the two lines in Fig. 3 (a). The comparison shows that discrete coordinates overlap a lot in different iterations, so the number of unique discrete coordinates increases very slowly, which expands much smaller variations in the sample space than continuous coordinates. We repeat this experiment in another scene, and we observe the similar statistics in Fig. 3 (c).

Moreover, the queries sampled on rays from different views have higher probability of co-occurrence in the same voxel, which makes them share the same discrete coordinate. As illustrated in Fig. 4 (a), although continuous coordinates of samples along two rays (blue and green dots) do not have an overlap at the intersection of the two rays on the surface, their corresponding discrete coordinates (centers of red boxes and centers of red boxes in bold) share the same intersection in Fig. 4 (b). This leads to involve the shared discrete coordinate in the volume rendering along both of the rays, which imposes multi-view consistency constraints at the shared discrete coordinate in a more effective way. More importantly, discrete coordinates facilitate rays from different views more easily intersect at discrete coordinates, which triggers more multi-view consistency constraints.

We statistically justify our claim using the same setting as Fig. 3 (a), and count the number of two rays that have an overlapped sampling on the surface in Fig. 3 (b). We still use  $51200^3$  quantized coordinates to discretize continuous coordinates of samples on rays, and monitor a fixed set of 1024 rays during optimization. For each ray that hits the surface, we project the hitting point to a neighboring view, and use the projection trajectory as another ray. Then, we sample points along these two rays separately using the sampling strategy in UNISURF, and check whether the two sets of sampled points have an overlap at the intersection. If both of the two rays have a sample on the surface, and their distance is smaller than a small threshold (1/16 of a voxel size), we regard these two rays involve the same sampled point in volume rendering, which indicates that a multi-view constraint take effect one time. Similarly, we check whether the two sampled points on the surface have the same discrete coordinates. We accumulate the times of multi-view constraint taking effect over iterations with continuous coordinates or discrete coordinates respectively, and show them using a logarithmic function as the two lines in Fig. 3 (b). The comparison shows that discrete coordinates triggers much more multi-view constraints than continuous coordinates. We repeat this experiment in another scene, and we observe the similar statistics in Fig. 3 (d). Although

neural network can generalize around continuous coordinates, imposing multi-view constraints on the same location through volume rendering can effectively infer 3D geometry with higher accuracy.

These benefits from discrete coordinates are vital to stabilize the optimization by reducing the uncertainty and ambiguity in the field, which achieves to reveal more accurate geometry and more smoother surfaces as the visual comparison in Fig. 3 (e) and (g). Although we discretize the field, the extremely high resolution does not produce artifacts or sawtooth effect in geometry or rendered images in Fig. 3 (f) and (h).

**Border Consistency.** Our quantized coordinates do not bring the disadvantages of voxelizing a field to neural implicit functions, although our voxelize the field in an extremely high resolution.

The disadvantages of voxelization include cubic computational complexity and inconsistency on borders of neighboring quantized coordinates. Different from feature grids [46, 73], which hold learnable features at vertices of grids in memory, we calculate quantized coordinates using a function of the field range and resolution, which does not bring any storage burden. This is also the key to enable us to quantize coordinates in an extremely high resolution. In addition, we implement the nearest interpolation in Eq. 4 by getting a continuous coordinate divided by the voxel interval, which avoids the computational nearest search. To achieve border consistency, the methods of learning local implicit functions [20, 4] use trilinear interpolation to interpolate features [46, 73] or implicit function values [20] from the nearest 8 voxel vertices. In contrast, our extremely high resolution leads to very small interval between neighboring quantized coordinates, which almost brings no inconsistency in implicit functions values or degenerate the rendering, as illustrated in Fig. 3 (e) and (g).

**Resolutions.** Although we claim quantized coordinates in an extremely high resolution benefit the learning of neural implicit representations from multi-view images, we note that continuous coordinates are actually quantized coordinates in an infinite resolution. Hence, a too high resolution does not help improve the inference. We will explore the effect of resolutions in experiments.

## 5. Experiments

We evaluate our method in 3D reconstruction from multi-view images for shapes with background and large scale scenes. We use quantized coordinate to learn either signed distance fields or occupancy fields with different baselines, and then run the marching cubes algorithm [28] to extract the zero level set as a surface. Note that we also use discrete coordinates to produce discrete signed distance field for the marching cubes.

## 5.1. Evaluations for Shapes

**Dataset and Metrics.** We evaluate our method in reconstructing 3D shapes without masks using multi-view images from the DTU dataset [19]. Following previous methods [43, 67, 66, 13, 63, 68, 64], we report our performance on the widely used 15 scans. For each scan, a scene is represented by 49 to 64 images with different shape appearances.

We use Chamfer distance to evaluate the accuracy of reconstructed surfaces, where we randomly sample points on the reconstructed surfaces, and compare them to the ground truth. Following previous methods [43, 67, 66, 13, 63, 68, 64, 10], we clean the reconstructed meshes using the respective masks. We use the official evaluation code released by the DTU dataset to measure our accuracy.

**Baselines.** To evaluate our method in learning both signed distance field and occupancy field, we use UNISURF [43], NeuS [63], Geo-Neus [14] and NeuralWarp [12] as baselines which are the state-of-the-art methods for learning implicit functions from multi-view images. All these methods do not use priors. Moreover, we do not evaluate our method in novel view synthesis, since the shape and radiance ambiguity [69] makes geometry no need to be represented as a surface, which is hard to have overlapped samples along different rays.

**Details.** We discretize a field into  $R^3$  voxels in an extremely high resolution, and regard the center of each voxel as a quantized coordinate. For the range of a field, UNISURF, NeuS, and NeuralWarp normalize a scene into a cube with a range of  $[-4, 4]$ ,  $[-2.5, 2.5]$ , and  $[-5.5, 5.5]$ , respectively. To evaluate our methods with different resolutions, we evaluate our results with two resolution settings which keeps each quantized coordinate covering a area with a similar size, i.e.,  $R = 51200$  and  $R = 25600$  for UNISURF and NeuS,  $R = 70400$  and  $R = 51200$  for NeuralWarp. We report these results in our supplementary materials, and list summarized results in the main text.

We use the official code released by UNISURF, NeuS, and NeuralWarp to produce our results with discrete coordinates. Moreover, we use the corresponding discrete coordinates to calculate positional encodings as in Eq. 5. For the normals required for color prediction or loss calculation in these methods, we also use the normals at the discrete coordinates. For the warping in NeuralWarp, we still use the continuous coordinates to get precise color from other views.

**Comparison.** We report numerical evaluations in DTU in Table. 1. We improve the performance of our baselines including UNISURF, NeuS, and NeuralWarp. Specifically, our results in all scenes outperform UNISURF. Except our comparable result in scene 97, we also achieve better performance than NeuralWarp in other scenes. Using NeuS as a baseline, we achieve a comparable result in scene 97, and get better results in other scenes except scene 83. The

reason is that there may be wrong parameter settings in the code, which makes us not manage to reproduce a 1.01 or similar result in scene 83 using NeuS. As for Geo-Neus, we can not reproduce the results reported in the original papers, hence, we train it and ours using the same data for fair comparison. Our results with Geo-Neus are the best among the results of all other state-of-the-art methods. We further provide visual comparisons in Fig. 5. Our advantages lie in the smooth surfaces with geometry details. Our methods can leverage more multi-view consistency to infer the implicit functions on the surface.

## 5.2. Evaluations for Scenes

**Dataset and Metrics.** We evaluate our performance in reconstructing scenes from multi-view images from ScanNet [11] and Replica [57]. We follow MonoSDF [68] to conduct evaluations using the same cases from these dataset. We also use the same metrics including Chamfer distance, the F-Score with a threshold of 5cm, and normal consistency to measure the error between the reconstructed surface and the ground truth surface.

**Baselines.** We use MonoSDF [68] as the baseline to evaluate our performance for scenes. It is the latest method for learning neural signed distance functions with depth and normal priors on images.

**Details.** MonoSDF normalizes a scene into a cube with a range of  $[-3.5, 3.5]$ , we use a resolution  $R = 51200$  to produce our results. We use the official code released by MonoSDF to produce our results with discrete coordinates. We use discrete coordinates to calculate positional encodings and also calculate normals at discrete coordinates.

**Comparisons.** We report our numerical comparisons in ScanNet in Tab. 2 which shows the average results over several scenes. We can see that we achieve much better results than our baseline MonoSDF, especially in terms of the metrics for surface smoothness. We provide the visual comparisons in Fig. 6 where we reconstruct more complete and more accurate surfaces than the other methods.

We further report our results in Replica. Our numerical and visual comparisons are shown in Tab. 3. We see that quantized coordinates can significantly reduce the variations in the sample space, and trigger more multi-view consistency at intersections of rays, which leads to more accurate, more completed, and smoother surfaces.

## 5.3. Analysis

We provide statistical analysis for our improvements over baselines. With quantized coordinates, we enable to decrease the variations in the sample space and trigger more multi-view consistency by involving the same discrete coordinate in volume rendering along rays from different view. This significantly leaves much less uncertainty and ambiguity in the field, which stabilizes the optimization. We repeat

| Scan                | 24          | 37          | 40          | 55          | 63          | 65          | 69          | 83          | 97          | 105         | 106         | 110         | 114         | 118         | 122         | Mean        |
|---------------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
| NeRF [39]           | 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        |
| VolSDF [66]         | 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        |
| HF-NeuS [64]        | 0.76        | 1.32        | 0.70        | 0.39        | 1.06        | 0.63        | 0.63        | 1.15        | 1.12        | 0.80        | 0.52        | 1.22        | 0.33        | 0.49        | 0.50        | 0.77        |
| MonoSDF [68]        | 0.66        | 0.88        | 0.43        | 0.40        | 0.87        | 0.78        | 0.81        | 1.23        | 1.18        | 0.66        | 0.66        | 0.96        | 0.41        | 0.57        | 0.51        | 0.73        |
| UNISURF [43]        | 1.32        | 1.36        | 1.72        | 0.44        | 1.35        | 0.79        | 0.8         | 1.49        | 1.37        | 0.89        | 0.59        | 1.47        | 0.46        | 0.59        | 0.62        | 1.02        |
| Ours(UNISURF)       | <b>0.85</b> | <b>0.95</b> | <b>1.00</b> | <b>0.38</b> | <b>1.25</b> | <b>0.59</b> | <b>0.69</b> | <b>1.36</b> | <b>1.19</b> | <b>0.71</b> | <b>0.52</b> | <b>1.15</b> | <b>0.42</b> | <b>0.48</b> | <b>0.50</b> | <b>0.80</b> |
| NeuS [63]           | 1.37        | 1.21        | 0.73        | 0.40        | 1.20        | 0.70        | 0.72        | <b>1.01</b> | <b>1.16</b> | 0.82        | 0.66        | 1.69        | 0.39        | 0.49        | 0.51        | 0.87        |
| Ours(NeuS)          | <b>0.71</b> | <b>0.90</b> | <b>0.68</b> | <b>0.38</b> | <b>1.0</b>  | <b>0.60</b> | <b>0.58</b> | 1.40        | 1.17        | <b>0.78</b> | <b>0.52</b> | <b>1.07</b> | <b>0.32</b> | <b>0.43</b> | <b>0.45</b> | <b>0.73</b> |
| NeuralWarp [12]     | <b>0.49</b> | 0.71        | 0.38        | 0.38        | 0.79        | 0.81        | 0.82        | 1.20        | <b>1.06</b> | .68         | 0.66        | 0.74        | 0.41        | 0.63        | 0.51        | 0.68        |
| Ours(NeuralWarp)    | <b>0.49</b> | <b>0.68</b> | <b>0.37</b> | <b>0.36</b> | <b>0.73</b> | <b>0.76</b> | <b>0.77</b> | <b>1.17</b> | 1.10        | <b>0.67</b> | <b>0.62</b> | <b>0.65</b> | <b>0.36</b> | <b>0.57</b> | <b>0.49</b> | <b>0.65</b> |
| Geo-NeuS [14]       | 0.46        | 0.85        | 0.38        | 0.43        | <b>0.89</b> | <b>0.50</b> | 0.50        | 1.26        | 0.89        | 0.66        | 0.52        | 0.82        | <b>0.31</b> | 0.43        | 0.46        | 0.62        |
| Ours(Geo-NeuS [14]) | <b>0.42</b> | <b>0.83</b> | <b>0.38</b> | <b>0.37</b> | 0.90        | 0.53        | <b>0.49</b> | <b>1.25</b> | <b>0.88</b> | <b>0.63</b> | <b>0.50</b> | <b>0.78</b> | <b>0.31</b> | <b>0.41</b> | <b>0.43</b> | <b>0.60</b> |

Table 1. Numerical comparisons with the latest methods in DTU dataset.

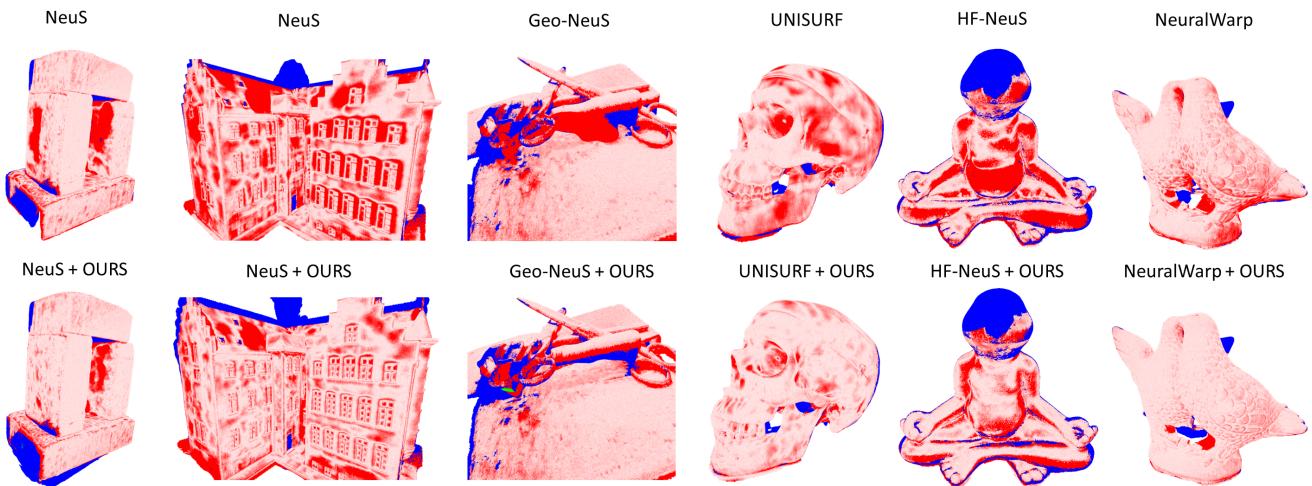


Figure 5. Visual comparison on DTU. Error maps highlight our improvements (white to red) over different baselines.

|                | Chamfer-L1↓  | Precision↑   | Recall↑      | F-score↑     |
|----------------|--------------|--------------|--------------|--------------|
| COLMAP [54]    | 0.141        | 0.711        | 0.441        | 0.537        |
| UNISURF [43]   | 0.359        | 0.212        | 0.362        | 0.267        |
| NeuS [63]      | 0.194        | 0.313        | 0.275        | 0.291        |
| VolSDF [66]    | 0.267        | 0.321        | 0.394        | 0.346        |
| Manhattan [16] | 0.070        | 0.621        | 0.586        | 0.602        |
| NeuRIS [62]    | 0.050        | 0.717        | 0.669        | 0.692        |
| MonoSDF [68]   | 0.042        | 0.799        | 0.681        | 0.733        |
| Ours           | <b>0.039</b> | <b>0.794</b> | <b>0.750</b> | <b>0.770</b> |

Table 2. Numerical comparisons in ScanNet.

|              | Normal C.↑   | CD-L1↓      | F-score↑     |
|--------------|--------------|-------------|--------------|
| MonoSDF [68] | 92.11        | 2.94        | 86.18        |
| Ours         | <b>93.86</b> | <b>2.76</b> | <b>90.16</b> |

Table 3. Numerical comparison with MonoSDF in Replica.

the same procedures as in Fig. 3 to count the number of unique coordinates that the network has seen and the number of multi-view consistency that takes effect in the first

1750 iterations using our method and UNISURF. We use the log function to scale the value, and report the ratio that UNISURF is over us in each scene in DTU in Tab. 4. As we can see, UNISURF has to observe 86.2 times more unique coordinates in average than us to infer neural implicit functions, however, can merely use 0.029 the number of multi-view constraints on the intersection of rays from different views of ours. Although neural networks can generalize values at continuous coordinates to the neighboring area, this brings uncertainty and ambiguity in the field, which may cause conflict effect in optimization that results in noisy surfaces and artifacts in empty space.

#### 5.4. Ablation Studies

We justify some key modules in our method based on UNISURF in a subset of the DTU dataset. We use Chamfer distance to evaluate the performance.

**Resolutions.** We explore the effect of resolu-

| Scan              | 24    | 37    | 40    | 55     | 63    | 65    | 69    | 83    | 97    | 105   | 106   | 110   | 114   | 118   | 122   | Mean  |
|-------------------|-------|-------|-------|--------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| Unique Ratio      | 140.0 | 50.0  | 163.5 | 80.8   | 57.8  | 99.6  | 48.2  | 59.4  | 107.1 | 89.1  | 86.4  | 43.5  | 136.4 | 75.6  | 55.3  | 86.2  |
| Consistency Ratio | 0.036 | 0.016 | 0.063 | 0.0234 | 0.012 | 0.024 | 0.021 | 0.014 | 0.027 | 0.024 | 0.053 | 0.022 | 0.041 | 0.039 | 0.022 | 0.029 |

Table 4. Statistical analysis for our improvements over baselines.

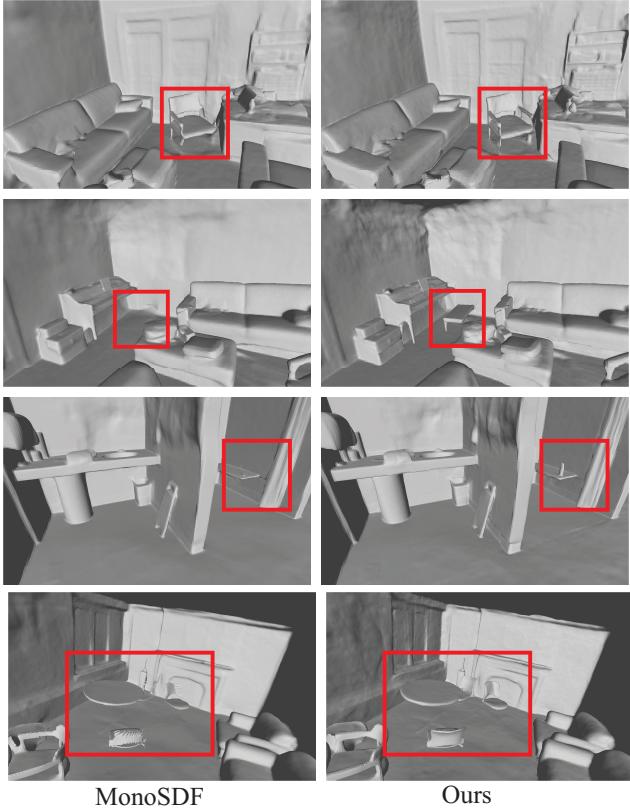


Figure 6. Visualization of improvements over MonoSDF in ScanNet. More visual comparisons in ScanNet can be found in our supplementary materials.

tions by learning neural implicit representations with quantized coordinates in different resolutions in Tab. 5. We use different resolution candidates  $\{1024, 25600, 38400, 51200, 76800, 102400\}$  to reconstruct the same scene. Compared to the continuous coordinates which can be regarded as infinity high resolution, as shown by “ $\infty$ ”, we achieve the best performance in  $R = 51200$ . The comparison shows that low resolution does not help infer accurate implicit representations, while the results with a too high resolution approach to the results with continuous coordinates. We visualize the effect of resolution in Fig. 7. With low resolution like 1024, we observe severe border inconsistency on the reconstructions and large error in the error map. While the error goes higher if we use a too high resolution. This is because much fewer points sampled along two rays with an intersection can overlap at the same discrete locations. Therefore, a too

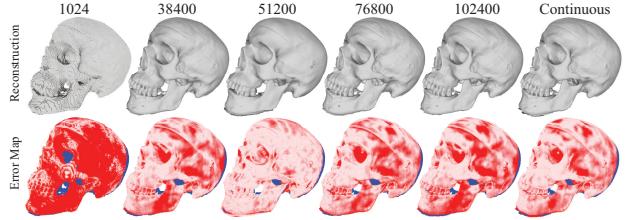


Figure 7. Visual comparison of quantized coordinate resolution in terms of reconstruction and error maps.

high resolution may degenerate the result.

| Resolution | 1024 | 25600 | 38400 | 51200       | 76800 | 102400 | $\infty$ |
|------------|------|-------|-------|-------------|-------|--------|----------|
| Chamfer ↓  | 1.67 | 0.70  | 0.64  | <b>0.59</b> | 0.62  | 0.69   | 0.79     |

Table 5. Effect of resolution for quantized coordinates.

**Border Consistency.** Since our quantized coordinates are defined in extremely high resolution, we achieve pretty good consistency on the border of neighboring quantized coordinates even we use the nearest interpolation to discretize continuous coordinates. We justify our nearest interpolation by comparing with the trilinear interpolation. For a query, we use the 8 nearest quantized coordinates to produce 8 occupancy labels and features, which are further used to predict 8 occupancy labels and colors for trilinear interpolation. The result of “Trilinear” in Tab. 6 shows that performing trilinear interpolation in extremely high resolution does not make the optimization converge well, and also brings 7 times more computation.

**Reconstruction with Marching Cubes.** We explore the effect of discrete coordinates on extracting surfaces with the marching cubes. With a implicit function learned with quantized coordinates, we can use either discrete coordinates or continuous coordinates to reconstruct meshes. The result of “MarchingCubes” Table 6 indicates that there is almost no difference between using discrete or continuous coordinates to extract meshes using marching cubes.

**Discrete Alternatives.** Besides using discrete coordinates and their corresponding positional encoding at the same time, we explore different discrete alternatives, such as using discrete coordinates with positional encodings of continuous coordinates or using continuous coordinates with positional encodings of discrete coordinates. The comparison in Tab. 6 shows that using discrete coordinates and their corresponding positional encodings achieves the best.

**Stability with Higher Frequency.** One advantage of quan-

| Trilinear | Discrete PE | Discrete Coordinates | MarchingCubes | Ours  |
|-----------|-------------|----------------------|---------------|-------|
| 0.72      | 0.75        | 0.74                 | 0.586         | 0.592 |

Table 6. Effect of trilinear interpolation and discrete alternatives.

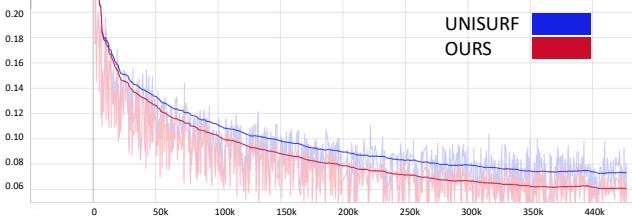


Figure 8. Speed up the convergence.

tized coordinates is to stabilize the optimization with high frequency positional encoding. We conduct experiments to compare with SAPE [18] using HF-NeuS. The comparisons in Tab. 7 show that our method, working along with HF-NeuS, can further stabilize with positional encoding with higher frequency. In contrast, HF-NeuS as well as NeuS by themselves are sensitive to the frequency and drastically degenerates its performance on some objects of DTU, as shown in Fig. 10. We found our method can also outperform SAPE in stabilizing optimization with higher frequency.

Moreover, we visualize the signed distance variance change with higher frequencies. We increase the frequency in positional encoding by adding either two high frequency  $[2^{14}, 2^{15}]$  or four high frequencies  $[2^{14}, \dots, 2^{17}]$ . With an interval of 50 iterations, we record the signed distances predicted by HF-NeuS and ours at the fixed 100 locations that are randomly sampled on the GT surface. Fig. 9 shows the variance of singed distances over 5000 iterations at each location. The comparisons show that our method produces lower signed distance variance than HF-NeuS on the sampled locations, which indicates that our method stabilizes the learning of signed distances with extremely high frequency positional encoding during training.

**Runtime Comparisons.** Runtime comparisons in Tab. 8 show that our quantized coordinates almost do not bring extra time cost. While quantized coordinates indeed lower the loss, which makes the optimization converge faster, as the comparison with UNISURF in Fig. 8.

| Scan                | 69          | 83          | 97          | 110         | Mean        |
|---------------------|-------------|-------------|-------------|-------------|-------------|
| NeuS [63]           | <b>0.57</b> | 1.48        | <b>1.09</b> | 1.2         | 1.09        |
| HF-NeuS [64]        | 0.70        | 1.41        | 1.29        | 1.58        | 1.25        |
| HF-NeuS [64] + OURS | 0.59        | <b>1.35</b> | 1.13        | <b>1.12</b> | <b>1.05</b> |

Table 7. Effect of stabilizing optimization with high frequencies.

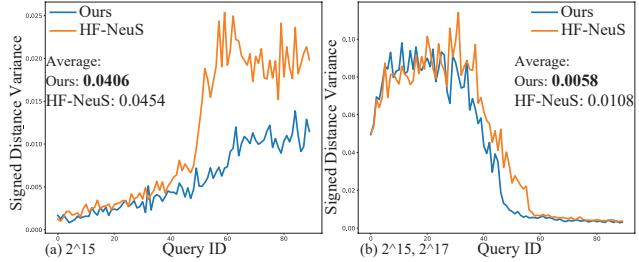


Figure 9. Signed Distance Variance.



Figure 10. Visualization of Stability with High Frequency.

|          | UNISURF | NeuS    | HF-NeuS | NeuralWarp | Geo-NeuS |
|----------|---------|---------|---------|------------|----------|
| baseline | 656.221 | 104.726 | 339.816 | 114.312    | 159.237  |
| Ours     | 656.992 | 104.591 | 340.214 | 114.411    | 158.819  |

Table 8. Comparison of 1000 iters run-time over baseline.

## 6. Conclusion

We introduce to learn neural implicit functions with quantized coordinates to decrease the uncertainty and ambiguity in the field during the optimization for multi-view 3D reconstruction. We transform continuous coordinates into discrete ones using nearest interpolation over the quantized coordinates. Our method significantly stabilizes the optimization and reveal more geometry details with high frequency positional encodings. We successively achieve this by reducing the variations in the sample space and triggering more multi-view consistency constraints to take effect in a more effective way. Our quantized coordinates are defined in extremely high resolution, which however does not bring any extra computational burden or inconsistency on borders of neighboring coordinates. Our experimental results show that we achieve the-state-of-the-art, and justify our ability of improving the accuracy of neural implicit functions learned by different methods in a general way.

## 7. Acknowledgements

This work was supported by NSF 61972353, IIS-1816511, OAC-1910469, and OAC-2311245 and Richard Barber interdisciplinary Research Award.

## References

- [1] Matan Atzmon and Yaron Lipman. SAL: Sign agnostic learning of shapes from raw data. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2020. 2
- [2] Matan Atzmon and yaron Lipman. SALD: sign agnostic learning with derivatives. In *International Conference on Learning Representations*, 2021. 2
- [3] Dejan Azinović, Ricardo Martin-Brualla, Dan B Goldman, Matthias Nießner, and Justus Thies. Neural rgb-d surface reconstruction. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 6290–6301, 2022. 2
- [4] Rohan Chabra, Jan Eric Lenssen, Eddy Ilg, Tanner Schmidt, Julian Straub, Steven Lovegrove, and Richard A. Newcombe. Deep local shapes: Learning local SDF priors for detailed 3D reconstruction. In *European Conference on Computer Vision*, volume 12374, pages 608–625, 2020. 2, 5
- [5] Chao Chen, Yu-Shen Liu, and Zhizhong Han. Latent partition implicit with surface codes for 3d representation. In *European Conference on Computer Vision*, 2022. 2
- [6] Chao Chen, Yu-Shen Liu, and Zhizhong Han. Gridpull: Towards scalability in learning implicit representations from 3d point clouds. In *IEEE International Conference on Computer Vision*, 2023. 2
- [7] Chao Chen, Yu-Shen Liu, and Zhizhong Han. Unsupervised inference of signed distance functions from single sparse point clouds without learning priors. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2023. 2
- [8] Rui Chen, Yongwei Chen, Ningxin Jiao, and Kui Jia. Fantasia3d: Disentangling geometry and appearance for high-quality text-to-3d content creation. *arXiv preprint arXiv:2303.13873*, 2023. 2
- [9] Zhiqin Chen and Hao Zhang. Learning implicit fields for generative shape modeling. *IEEE Conference on Computer Vision and Pattern Recognition*, 2019. 2
- [10] Gene Chou, Ilya Chugunov, and Felix Heide. Gensdf: Two-stage learning of generalizable signed distance functions. In *Neural Information Processing Systems*, 2022. 6
- [11] Angela Dai, Matthias Nießner, Michael Zollöfer, Shahram Izadi, and Christian Theobalt. Bundlefusion: Real-time globally consistent 3d reconstruction using on-the-fly surface re-integration. *ACM Transactions on Graphics*, 2017. 6
- [12] François Darmon, Bénédicte Bascl, Jean-Clément Devaux, Pascal Monasse, and Mathieu Aubry. Improving neural implicit surfaces geometry with patch warping. In *CVPR*, pages 6250–6259. IEEE, 2022. 6, 7
- [13] Qiancheng Fu, Qingshan Xu, Yew-Soon Ong, and Wenbing Tao. Geo-Neus: Geometry-consistent neural implicit surfaces learning for multi-view reconstruction. 2022. 1, 3, 6
- [14] Qiancheng Fu, Qingshan Xu, Yew-Soon Ong, and Wenbing Tao. Geo-Neus: Geometry-consistent neural implicit surfaces learning for multi-view reconstruction. In *Advances in Neural Information Processing Systems*, 2022. 2, 3, 6, 7
- [15] Amos Gropp, Lior Yariv, Niv Haim, Matan Atzmon, and Yaron Lipman. Implicit geometric regularization for learning shapes. In *International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 3789–3799, 2020. 2
- [16] Haoyu Guo, Sida Peng, Haotong Lin, Qianqian Wang, Guofeng Zhang, Hujun Bao, and Xiaowei Zhou. Neural 3d scene reconstruction with the manhattan-world assumption. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2022. 1, 2, 7
- [17] Anchit Gupta, Wenhan Xiong, Yixin Nie, Ian Jones, and Barlas Oğuz. 3dgen: Triplane latent diffusion for textured mesh generation, 2023. 2
- [18] Amir Hertz, Or Perel, Raja Giryes, Olga Sorkine-Hornung, and Daniel Cohen-Or. Sape: Spatially-adaptive progressive encoding for neural optimization. In *Thirty-Fifth Conference on Neural Information Processing Systems*, 2021. 1, 9
- [19] Rasmus Jensen, Anders Dahl, George Vogiatzis, Engil Tola, and Henrik Aanæs. Large scale multi-view stereopsis evaluation. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 406–413, 2014. 6
- [20] Chiyu Jiang, Avneesh Sud, Ameesh Makadia, Jingwei Huang, Matthias Nießner, and Thomas Funkhouser. Local implicit grid representations for 3D scenes. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2020. 2, 5
- [21] Yue Jiang, Dantong Ji, Zhizhong Han, and Matthias Zwicker. SDFDiff: Differentiable rendering of signed distance fields for 3D shape optimization. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2020. 2
- [22] Heewoo Jun and Alex Nichol. Shap-e: Generating conditional 3d implicit functions, 2023. 2
- [23] A. Laurentini. The visual hull concept for silhouette-based image understanding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(2):150–162, 1994. 2
- [24] Tianyang Li, Xin Wen, Yu-Shen Liu, Hua Su, and Zhizhong Han. Learning deep implicit functions for 3D shapes with dynamic code clouds. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 12830–12840, 2022. 2
- [25] Zhaoshuo Li, Thomas Müller, Alex Evans, Russell H Taylor, Mathias Unberath, Ming-Yu Liu, and Chen-Hsuan Lin. Neuralangelo: High-fidelity neural surface reconstruction. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 1
- [26] Shi-Lin Liu, Hao-Xiang Guo, Hao Pan, Pengshuai Wang, Xin Tong, and Yang Liu. Deep implicit moving least-squares functions for 3D reconstruction. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2021. 2
- [27] Yu-Tao Liu, Li Wang, Jie yang, Weikai Chen, Xiaoxu Meng, Bo Yang, and Lin Gao. Neudf: Learning neural unsigned distance fields with volume rendering, 2023. 2
- [28] William E. Lorensen and Harvey E. Cline. Marching cubes: A high resolution 3D surface construction algorithm. *Computer Graphics*, 21(4):163–169, 1987. 5
- [29] Baorui Ma, Zhizhong Han, Yu-Shen Liu, and Matthias Zwicker. Neural-pull: Learning signed distance functions from point clouds by learning to pull space onto surfaces. In *International Conference on Machine Learning*, 2021. 2
- [30] Baorui Ma, Yu-Shen Liu, and Zhizhong Han. Learning signed distance functions from noisy 3d point clouds via noise to noise mapping, 2023. 2

- [31] Baorui Ma, Yu-Shen Liu, Matthias Zwicker, and Zhizhong Han. Reconstructing surfaces for sparse point clouds with on-surface priors. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2022. 2
- [32] Baorui Ma, Yu-Shen Liu, Matthias Zwicker, and Zhizhong Han. Surface reconstruction from point clouds by learning predictive context priors. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2022. 2
- [33] Julien N. P. Martel, David B. Lindell, Connor Z. Lin, Eric R. Chan, Marco Monteiro, and Gordon Wetzstein. ACORN: adaptive coordinate networks for neural scene representation. *CoRR*, abs/2105.02788, 2021. 2
- [34] Luke Melas-Kyriazi, Christian Rupprecht, Iro Laina, and Andrea Vedaldi. Realfusion: 360deg reconstruction of any object from a single image, 2023. 2
- [35] Luke Melas-Kyriazi, Christian Rupprecht, and Andrea Vedaldi.  $pc^2$ : Projection-conditioned point cloud diffusion for single-image 3d reconstruction, 2023. 2
- [36] Xiaoxu Meng, Weikai Chen, and Bo Yang. Neat: Learning neural implicit surfaces with arbitrary topologies from multi-view images, 2023. 2
- [37] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3D reconstruction in function space. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2019. 2, 3
- [38] Mateusz Michalkiewicz, Jhony K. Pontes, Dominic Jack, Mahsa Baktashmotagh, and Anders P. Eriksson. Deep level sets: Implicit surface representations for 3D shape inference. *CoRR*, abs/1901.06802, 2019. 2
- [39] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. NeRF: Representing scenes as neural radiance fields for view synthesis. In *European Conference on Computer Vision*, 2020. 1, 2, 7
- [40] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *arXiv:2201.05989*, 2022. 2
- [41] Alex Nichol, Heewoo Jun, Prafulla Dhariwal, Pamela Mishkin, and Mark Chen. Point-e: A system for generating 3d point clouds from complex prompts, 2022. 2
- [42] Michael Niemeyer, Lars Mescheder, Michael Oechsle, and Andreas Geiger. Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2020. 2
- [43] Michael Oechsle, Songyou Peng, and Andreas Geiger. UNISURF: Unifying neural implicit surfaces and radiance fields for multi-view reconstruction. In *International Conference on Computer Vision*, 2021. 1, 2, 3, 4, 6, 7
- [44] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. DeepSDF: Learning continuous signed distance functions for shape representation. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2019. 2, 3
- [45] Keunhong Park, Utkarsh Sinha, Jonathan T. Barron, Sofien Bouaziz, Dan B Goldman, Steven M. Seitz, and Ricardo Martin-Brualla. Nerfies: Deformable neural radiance fields. *IEEE International Conference on Computer Vision*, 2021. 1, 2
- [46] Songyou Peng, Michael Niemeyer, Lars M. Mescheder, Marc Pollefeys, and Andreas Geiger. Convolutional occupancy networks. In *European Conference on Computer Vision*, volume 12348, pages 523–540, 2020. 5
- [47] Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion, 2022. 2
- [48] Amit Raj, Srinivas Kaza, Ben Poole, Michael Niemeyer, Nataniel Ruiz, Ben Mildenhall, Shiran Zada, Kfir Aberman, Michael Rubinstein, Jonathan Barron, Yuanzhen Li, and Varun Jampani. Dreambooth3d: Subject-driven text-to-3d generation, 2023. 2
- [49] Christian Reiser, Richard Szeliski, Dor Verbin, Pratul P. Srinivasan, Ben Mildenhall, Andreas Geiger, Jonathan T. Barron, and Peter Hedman. Merf: Memory-efficient radiance fields for real-time view synthesis in unbounded scenes, 2023. 2
- [50] Radu Alexandru Rosu and Sven Behnke. Permutosdf: Fast multi-view reconstruction with implicit surfaces using permutohedral lattices. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 1
- [51] Darius Rückert, Linus Franke, and Marc Stamminger. Adop: Approximate differentiable one-pixel point rendering. *arXiv:2110.06635*, 2021. 2
- [52] Sara Fridovich-Keil and Alex Yu, Matthew Tancik, Qinhong Chen, Benjamin Recht, and Angjoo Kanazawa. Plenoxels: Radiance fields without neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2022. 2
- [53] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016. 2
- [54] Johannes Lutz Schönberger, Enliang Zheng, Marc Pollefeys, and Jan-Michael Frahm. Pixelwise view selection for unstructured multi-view stereo. In *European Conference on Computer Vision*, 2016. 2, 7
- [55] Vincent Sitzmann, Julien N.P. Martel, Alexander W. Bergman, David B. Lindell, and Gordon Wetzstein. Implicit neural representations with periodic activation functions. In *Advances in Neural Information Processing Systems*, 2020. 1
- [56] Lars Mescheder, Marc Pollefeys, Andreas Geiger, Songyou Peng, Michael Niemeyer. Convolutional occupancy networks. In *European Conference on Computer Vision*, 2020. 2
- [57] Julian Straub, Thomas Whelan, Lingni Ma, Yufan Chen, Erik Wijmans, Simon Green, Jakob J. Engel, Raul Mur-Artal, Carl Ren, Shobhit Verma, Anton Clarkson, Mingfei Yan, Brian Budge, Yajie Yan, Xiaqing Pan, June Yon, Yuyang Zou, Kimberly Leon, Nigel Carter, Jesus Briales, Tyler Gillingham, Elias Mueggler, Luis Pesqueira, Manolis Savva, Dhruv Batra, Hauke M. Strasdat, Renzo De Nardi, Michael Goesele, Steven Lovegrove, and Richard A. Newcombe. The replica dataset: A digital replica of indoor spaces. *CoRR*, abs/1906.05797, 2019. 6
- [58] Towaki Takikawa, Joey Litalien, Kangxue Yin, Karsten Kreis, Charles Loop, Derek Nowrouzezahrai, Alec Jacobson,

- Morgan McGuire, and Sanja Fidler. Neural geometric level of detail: Real-time rendering with implicit 3D shapes. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2021. 2
- [59] Matthew Tancik, Pratul P. Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan T. Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. *NeurIPS*, 2020. 1
- [60] Jiapeng Tang, Jiabao Lei, Dan Xu, Feiyi Ma, Kui Jia, and Lei Zhang. SA-ConvONet: Sign-agnostic optimization of convolutional occupancy networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021. 2
- [61] Delio Vicini, Sébastien Speierer, and Wenzel Jakob. Differentiable signed distance function rendering. *ACM Transactions on Graphics*, 41(4):125:1–125:18, 2022. 1, 2
- [62] Jiepeng Wang, Peng Wang, Xiaoxiao Long, Christian Theobalt, Taku Komura, Lingjie Liu, and Wenping Wang. NeuRIS: Neural reconstruction of indoor scenes using normal priors. In *European Conference on Computer Vision*, 2022. 1, 2, 7
- [63] 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 *Advances in Neural Information Processing Systems*, pages 27171–27183, 2021. 1, 2, 3, 6, 7, 9
- [64] Yiqun Wang, Ivan Skorokhodov, and Peter Wonka. HF-NeuS: Improved surface reconstruction using high-frequency details. 2022. 1, 2, 6, 7, 9
- [65] Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. Mvsnet: Depth inference for unstructured multi-view stereo. *European Conference on Computer Vision*, 2018. 2
- [66] Lior Yariv, Jiatao Gu, Yoni Kasten, and Yaron Lipman. Volume rendering of neural implicit surfaces. In *Advances in Neural Information Processing Systems*, 2021. 1, 3, 6, 7
- [67] Lior Yariv, Yoni Kasten, Dror Moran, Meirav Galun, Matan Atzmon, Basri Ronen, and Yaron Lipman. Multiview neural surface reconstruction by disentangling geometry and appearance. *Advances in Neural Information Processing Systems*, 33, 2020. 1, 2, 3, 6
- [68] Zehao Yu, Songyou Peng, Michael Niemeyer, Torsten Sattler, and Andreas Geiger. MonoSDF: Exploring monocular geometric cues for neural implicit surface reconstruction. *ArXiv*, abs/2022.00665, 2022. 1, 2, 6, 7
- [69] Kai Zhang, Gernot Riegler, Noah Snavely, and Vladlen Koltun. NERF++: Analyzing and improving neural radiance fields. <https://arxiv.org/abs/2010.07492>, 2020. 6
- [70] Wenbin Zhao, Jiabao Lei, Yuxin Wen, Jianguo Zhang, and Kui Jia. Sign-agnostic implicit learning of surface self-similarities for shape modeling and reconstruction from raw point clouds. *CoRR*, abs/2012.07498, 2020. 2
- [71] Junsheng Zhou, Baorui Ma, Shujuan Li, Yu-Shen Liu, and Zhizhong Han. Learning a more continuous zero level set in unsigned distance fields through level set projection, 2023. 1
- [72] Junsheng Zhou, Baorui Ma, Yu-Shen Liu, Yi Fang, and Zhizhong Han. Learning consistency-aware unsigned distance functions progressively from raw point clouds. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. 2
- [73] Zihan Zhu, Songyou Peng, Viktor Larsson, Weiwei Xu, Hujun Bao, Zhaopeng Cui, Martin R. Oswald, and Marc Pollefeys. Nice-slam: Neural implicit scalable encoding for slam. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2022. 2, 5